

Міністерство освіти та науки України
Національна металургійна академія України

Теорія та методика
навчання математики,
фізики, інформатики

*Збірник наукових праць
Випуск V*

Том 3

Кривий Ріг
Видавничий відділ НМетАУ
2005

Теорія та методика навчання математики, фізики, інформатики: Збірник наукових праць. Випуск V: В 3-х томах. – Кривий Ріг: Видавничий відділ НМетАУ, 2005. – Т. 3: Теорія та методика навчання інформатики. – 319 с.

Збірник містить статті з різних аспектів дидактики інформатики і проблем її викладання в вузі та школі. Значну увагу приділено проблемам розвитку методичних систем навчання інформатики і питанням захисту інформації та інформаційної безпеки у шкільній та вузівській практиці.

Для студентів вищих навчальних закладів, аспірантів, наукових та педагогічних працівників.

Редакційна колегія:

В.М. Соловійов, доктор фізико-математичних наук, професор

Є.Я. Глушко, доктор фізико-математичних наук, професор

О.І. Олейніков, доктор фізико-математичних наук, професор

М.І. Жалдак, доктор педагогічних наук, професор

П.С. Атаманчук, доктор педагогічних наук, професор

В.І. Клочко, доктор педагогічних наук, професор

Ю.О. Дорошенко, доктор технічних наук, професор

О.Д. Учитель, доктор технічних наук, професор

І.О. Теплицький, відповідальний редактор

С.О. Семеріков, відповідальний секретар

Рецензенти:

Г.Ю. Маклаков – д-р техн. наук, професор кафедри кібернетики та обчислювальної техніки Севастопольського національного технічного університету, науковий керівник лабораторії біокібернетики, дійсний член Міжнародної академії біоенерготехнологій

А.Ю. Ків – д-р фіз.-мат. наук, професор, завідувач кафедри теоретичної фізики Південноукраїнського державного педагогічного університету (м. Одеса)

ISBN 966-537-619-3

ГЕОІНФОРМАЦІЙНІ СИСТЕМИ У СУЧАСНІЙ ПІДГОТОВЦІ СПЕЦІАЛІСТІВ З КОМП'ЮТЕРНИХ НАУК

Н.М. Аушева^а, А.Л. Гурін^б

м. Київ, Національний технічний університет України
“Київський політехнічний інститут”

^а aana@svitonline.com

^б artem@aprodos.ntu-kpi.kiev.ua

Постановка проблеми. На сучасному етапі зростає значення різноманітних автоматизованих систем. Ці системи складаються з програмних та апаратних засобів, які автоматизують роботу користувача, допомагають проводити розрахунки, зберігати велику кількість інформації та виконувати її сумісну обробку.

Сучасні геоінформаційні системи (ГІС) являють собою новий тип інтегрованих інформаційних систем і становлять нову основу автоматизованих систем керування. ГІС використовуються у транспорті, навігації, геології, географії, військовій справі, топографії, економіці. Практично це визначає ГІС як багатоцільові, багатоаспектні системи.

В світовій практиці розвиток геоінформаційних технологій, які забезпечують збір, обробку, зберігання і розподіл різних геоінформаційних ресурсів (даних), проходить дуже бурхливо і ця тенденція збережеться у найближчому десятиріччі. Володіння сучасними геоінформаційними технологіями і ресурсами створює передумови для успішного розвитку будь-якої держави. Тому необхідно розуміти важливість та актуальність впровадження в вищу освіту новітніх наукових розробок у галузі геоінформаційних технологій.

Аналіз останніх досліджень. На сьогоднішній день існує багато джерел інформації з різних аспектів застосування геоінформаційних технологій.

Розкриття основних понять та методів просторового аналізу географічних інформаційних систем можна одержати з літератури [1–3], де показано основні принципи побудови аналітичного процесу, реалізації просторових методів зберігання інформації та оформлення результатів.

Набули активного розвитку курси дистанційного навчання, створюються електронні підручники. Багато матеріалів студенти можуть отримати і в мережі Internet, в якій розміщені матеріали наукових конференцій, учбово-практичних семінарів, матеріали геоінформаційних розробок організацій [4–6].

Але слід зазначити, що основні нові матеріали знаходяться у зарубіжній літературі, а також на зарубіжних вебсайтах, що мають закритий доступ, або неповні данні, тому на сьогодні існує певна проблема забезпечення дисципліни літературою.

Формування цілей статті (постановка завдання). Головним об'єктом

діяльності спеціалістів комп'ютерного напрямку підготовки є створення та підтримка функціонування комп'ютерних систем у різноманітних сферах діяльності людини, які в багатьох випадках пов'язані з геоінформаційними даними: побудова карт зон забруднення навколишнього середовища; відображення місцезнаходження техногенних об'єктів; ведення кадастрового реєстру земельного фонду; адміністративне управління територіями; моніторинг стану гідросфери; відслідковування якості ґрунтів; ведення демографічної, економічної та політичної ситуації; розробка карт населених пунктів, транспортних систем і т. ін.

Метою цієї статті є висвітлення актуальності вивчення геоінформаційних технологій спеціалістів з комп'ютерних наук та визначення структурно-логічної схеми дисципліни "Геоінформаційні системи".

Основна частина. Геоінформаційна система – це автоматизована інформаційна система, яка призначена для обробки просторово-часових даних, основою інтеграції яких є географічна інформація. Згідно з цим, ГІС розглядається як набір підсистем, що охоплюють комплексні процеси збору, зберігання, попередньої обробки, представлення, аналізу даних, генерування отриманих результатів, що мають наступний перелік: підсистеми обробки даних; підсистеми аналізу даних; підсистеми використання інформації; підсистеми управління.

ГІС можуть використовуватись як системи управління, що призначені для забезпечення прийняття рішень по оптимальному керуванню міським господарством, землями і ресурсами, по керуванню транспортом і роздрібною торгівлею, використанню озер або інших просторових об'єктів. Геоінформаційні системи включають цілий набір технологій (передусім технології збору інформації) таких систем, як географічні інформаційні системи, системи картографічної інформації, автоматизовані системи картографування, автоматизовані фотограмметричні системи, земельні інформаційні системи, автоматизовані кадастрові системи і т.ін. Якщо говорити про автоматизовані інформаційні системи, то ГІС об'єднують ряд технологій типу автоматизованих систем наукових досліджень (АСНД), систем автоматизованого проектування (САПР), автоматизованих довідково-інформаційних систем (АДІС).

Сучасні ГІС працюють з базами даних і характеризуються широким набором даних, що збираються за допомогою різних методів і технологій. Велике значення на сьогоднішній день набули експертні систем. До основних задач, що потребують застосування експертних систем, відносяться обробка відеозображень, перетворення растрових зображень в векторні графічні моделі, обробка картографічної інформації, аналіз моделей, одержання рішень на базі геоінформації.

Основу процесів обробки складає цифрове моделювання, що дозволяє виконувати векторно-топологічне моделювання, побудову цифрових моделей місцевості, буферізацію об'єктів, аналіз мереж. В ГІС широко викорис-

товується велика кількість методів і процесів моделювання. В цілому моделювання і побудова моделей базується на відомих принципах і підходах, які використовуються в інших автоматизованих системах. Потрібно уточнити, що в будь-якому випадку основою інтеграції технологій в ГІС є технологія САПР.

Таким чином, в ГІС принципово вирішуються всі задачі, які виконувались раніше в автоматизованих системах управління, але на більш високому рівні інтеграції і об'єднанні даних. Це дає змогу дивитись на ГІС, як на новий сучасний варіант автоматизованих систем управління, що використовують більшу кількість даних і більшу кількість методів аналізу й прийняття рішень, причому в першу чергу використовуються методи просторового аналізу.

З точки зору програмування, модулі і додатки являють собою єдину середу користувача інструментальних ГІС. Як правило, до ядра ГІС підключаються тематично орієнтовані модулі, доповнені додатками для керування модулями даних, виконання розрахунків, аналізу і проектування, побудови цифрових моделей, обробки растрових зображень, організації інтерфейсів. Зростає значення модулів для тривимірного (3D) проектування, генерації планів, автоматизованого документування проектів і вибору оптимальних варіантів. При цьому реалізована можливість підключення модулів, розроблених користувачем.

Спеціалістам з комп'ютерних наук необхідно навчитись розробляти нові додатки і доповнювати можливості ГІС при вирішенні конкретної практичної задачі.

В зв'язку з вищесказаним, метою і завданням дисципліни "Геоінформаційні системи" є ознайомлення з принципами картографії, сучасними методами просторового аналізу та оволодіння базовим інструментарієм відповідного програмного забезпечення, а також надання вмінь по розробці нових додатків, що доповнюють можливості учбової ГІС. В даному випадку як навчальна система використовується MapInfo Professional, що на сьогодні має універсальні можливості ГІС широкого призначення і дозволяє вирішувати фахівцям широкий клас завдань. Система має наступні можливості: методи аналізу даних в реляційній базі даних, пошук географічних об'єктів, методи тематичного зафарбування карт, методи створення та редагування легенд, підтримка широкого набору форматів даних, доступ до віддалених баз даних і т.ін.

Дисципліна має практичну направленість щодо оволодіння навичками роботи з сучасною ГІС та створення програмного забезпечення, яке розширює її можливості аналізу та обробки просторової інформації. Для цього необхідно надати змогу оволодіти мовою програмування MapBasic і виконати комплекс лабораторних робіт.

Для вивчення матеріалу з дисципліни "Геоінформаційні системи" пропонується розглянути наступні розділи та теми:

Розділ 1. Загальні відомості про геоінформаційні системи.

Тема 1.1. Теоретичні основи побудови геоінформаційних систем.

Тема 1.2. Моделі організації даних та види просторового аналізу в ГІС.

Розділ 2. Методологія вирішення геоінформаційних задач на базі MapInfo.

Тема 2.1. Середовище MapInfo.

Тема 2.2. Технологія створення векторних карт в середовищі MapInfo.

Тема 2.3. Робота з даними в ГІС.

Розділ 3. MapBasic.

Тема 3.1. Інтегроване середовище MapBasic.

Тема 3.2. Розробка програмного забезпечення для ГІС.

Тема 3.3. Механізм доступу до віддалених баз даних.

Тема 3.4. Інтегрована картографія.

Розділ 4. Методологія проектування нових геоінформаційних систем.

Тема 4.1. Менеджмент ГІС проекту.

В структурно-логічній схемі навчання геоінформаційні системи необхідно розмістити після освоєння студентом знань з основних фундаментальних дисциплін, а саме на 3 або 4 курсі.

Для вивчення першого розділу рекомендуємо використовувати традиційні підручники з географії, картографії, топографії. Одночасно другий розділ є принципово новий і потребує ознайомлення з елементами багатьох різнопланових дисциплін: математичних (топология, теорія графів), обробки інформації (структуризація інформації, моделювання даних, побудова поверхонь). Освоєння третього розділу вимагає наполегливої роботи безпосередньо з програмним продуктом, тому студентам необхідно вже прослухати дисципліни “Основи програмування та алгоритмічні мови” та “Об’єктно-орієнтоване програмування”. Четвертий розділ потребує знань економічних дисциплін (економічне планування, теорія економічних ризиків і т.ін.)

Для досконалого вивчення матеріалу з дисципліни “Геоінформаційні системи” обсяг навчального навантаження може бути розрахований в залежності від спеціальності з якої навчається студент. Рекомендується виділяти від 108 до 216 годин (аудиторні години та самостійна робота).

Лабораторні роботи проводяться з метою одержання практичних навичок користування геоінформаційними системами та створення прикладних програм, які дозволяють автоматизувати роботу з геоданими і являються важливими елементами геоінформаційних технологій.

Для прикладу розглянемо декілька лабораторних робіт.

Лабораторна робота №1. Побудова топологічного опису полігонів та поверхонь.

Мета роботи: Одержати практичні навички в створенні та редагуванні геоінформаційних баз даних.

Завдання: намалювати декілька топологічних (картографічних) об’єктів, структурно описати їх топологічні відношення, визначити струк-

тури бази даних для опису об'єктів і створити записи в базі даних, які описують топологічні відношення.

Лабораторна робота №2. Просторовий аналіз, побудова TIN поверхонь.

Мета роботи: Освоїти методи просторового аналізу та навчитися моделювати поверхні.

Завдання: намалювати декілька однотипних топологічних (картографічних) об'єктів; провести оверлейний аналіз та створити нові сутності, провести аналіз суміжності; побудувати кілька полігонів та транспортних вузлів і виконати аналіз сполученості, провести класифікацію та рекласифікацію; побудувати точки та лінії поверхні; провести TIN моделювання.

Лабораторна робота №3. Побудова та використання тематичних карт.

Мета роботи: Оволодіти методами тематичного картографування та освоїти технологію їх створення.

Завдання: побудувати тематичні карти, створити легенди, освоїти метод ранжування даних по діапазонах, метод створення діаграм, метод точкової щільності, метод різної величини символів, метод індивідуальних значень.

Висновки. Таким чином, геоінформаційні системи є важливою складовою підготовки фахівців з напрямку “Комп’ютерні науки”. В роботі визначена тематика та наведені приклади лабораторних робіт. Подальші публікації за проблематикою даної роботи будуть пов’язані з особливостями викладання дисципліни “Геоінформаційні системи” для різних спеціальностей.

Література:

1. Цветков В.Я. Геоинформационные системы и технологии. – М.: Финансы и статистика, 1998. – 288 с.
2. Mapinfo Professional (русское издание). Руководство пользователя. – Mapinfo Corporation Troy, New York. Эсти-Мап, 1999. – 539 с.
3. Митчелл Энди. Руководство по ГИС анализу. Часть 1: Пространственные модели и взаимосвязи. – К.: ЗАО ЕСОММ Со, Стилос, 2000. – 198 с.
4. <http://www.geomatica.kiev.ua>
5. <http://www.mylend.org.ua>
6. <http://www.geology.univer.kharkov.ua>

ОБЗОР CASE-СРЕДСТВ ДЛЯ ПОСТРОЕНИЯ ДИАГРАММ UML

А.В. Бабич

г. Полтава, Полтавский государственный педагогический университет

им. В.Г. Короленко

alexander.babich@rambler.ru

В настоящее время унифицированный язык моделирования (UML – Unified Modeling Language) является одним из наиболее популярных инструментов в сфере разработки программного обеспечения (ПО). Это воистину «самая модная» технология на данный момент. В чем же причина такой популярности? Дело в том, что UML является визуальным языком моделирования, который позволяет системным архитекторам представлять свое видение системы в стандартной и легкой для понимания форме. Также UML предоставляет эффективный механизм совместного использования проектных решений и взаимодействия разработчиков во время командной работы над проектом.

Данная работа не ставит своей целью рассказать об основных конструкциях языка (хотя сейчас наблюдается явный дефицит такой литературы на русском, а тем более, на украинском языках). Возможно, мы еще вернемся к этой теме в последующих публикациях, а пока отошлем читателя к стандарту языка [1]. Цель данной работы – дать краткий обзор и сравнительную характеристику CASE-средств, которые могут быть использованы для построения диаграмм UML как при работе над реальными проектами, так и в учебных целях. Ведь предметы, связанные с проектированием ПО, читаются во многих вузах Украины и СНГ, и именно потому такой обзор мог бы быть востребованным.

Эта публикация написана на основе учебных материалов, разработанных автором во время его участия в проекте «Виртуоз» [2] – программе семестровой индустриально-академической стажировки для молодых IT-преподавателей из стран СНГ, проводимой корпорацией Intel совместно с ННГУ при поддержке IBM, Microsoft, Borland, Kaspersky Lab с августа по декабрь 2004 года. При сравнении рассмотренных программных продуктов автор опирался на свои собственные субъективные впечатления от работы с ними, поэтому его оценки не претендуют на истину в последней инстанции.

Как уже говорилось, UML – отличное средство моделирования, но строить диаграммы на бумаге – не всегда удобно, хотя бы по причине сложностей с редактированием, распространением и т.д. Поэтому используются CASE-средства – программы специального вида. CASE-средства помогут разработчикам строить профессионально выглядящие диаграммы, даже если кто-то из них не в состоянии провести прямую линию!

CASE-средства [3] (от Computer Aided Software/System Engineering) позволяют проектировать любые системы на компьютере. Необходимый эле-

мент системного и структурно-функционального анализа, CASE-средства применимы практически во всех сферах деятельности, позволяя моделировать бизнес-процессы, базы данных, компоненты программного обеспечения, деятельность и структуру организаций. Результат применения CASE-средств – оптимизация систем, снижение расходов, повышение эффективности, снижение вероятности ошибок.

В данной работе мы рассмотрим некоторые наиболее достойные внимания, с точки зрения автора, CASE-средства для построения UML-диаграмм. Причем постараемся рассмотреть и признанных лидеров рынка, и его «аутсайдеров», и коммерческих «монстров», и «легкие» программы с открытым исходным кодом. А именно, речь пойдет о таких программных продуктах:

- IBM Rational Rose
- Borland Together
- Microsoft Visio
- SparxSystems Enterprise Architect
- Gentleware Poseidon
- SmartDraw
- Dia

Далее мы постараемся дать краткую характеристику каждого продукта.

IBM Rational Rose [4] – современное и мощное средство анализа, визуального моделирования и разработки программных систем. На данный момент является, по сути, стандартом де-факто среди программ подобного рода. Rose имеет весь необходимый набор визуальных средств проектирования, помогает решить проблемы с кодогенерацией на определенном языке программирования, осуществляет прямое и обратное проектирование. Такой арсенал позволит не только проектировать новую систему, но и доработать старую, произведя процесс обратного проектирования. Для того, чтобы наиболее полно покрыть весь сегмент рынка средств проектирования и разработки, компания IBM выпускает несколько версий своего продукта. Каждая из них может решать как строго определенный круг задач, так и весь спектр проблем проектирования и разработки. Rational Rose неоднократно признавалось различными изданиями лучшим средством проектирования.

Borland Together ControlCenter [5] – это интегрированная платформа разработки, позволяющая упростить и ускорить анализ, дизайн, разработку и развертывание комплексных корпоративных приложений. Комбинируя эти возможности в одном интегрированном решении с поддержкой UML, Together ControlCenter помогает командно строить высококачественные системы быстро и эффективно. Продукт действительно очень мощный и интересный. И еще один, важный для нас факт: существует Borland Together Designer Community Edition – бесплатная среда разработки для создания диаграмм UML 2.0.

Microsoft Office Visio Professional 2003 [6] – это программа для по-

строения диаграмм, которая поможет разработчикам ПО и программным архитекторам эффективнее документировать, проектировать, обмениваться идеями. Тесная интеграция с Microsoft Office позволяет Visio Professional 2003 легко сделать информацию, содержащуюся в диаграммах повторно используемой и распространяемой. С точки зрения автора продукт еще весьма «сыроват» и подходит лишь для домашнего применения.

Enterprise Architect (EA) [7] – мощное средство для описания, документирования и построения Вашего программного проекта. Как уверяют разработчики (Sparx Systems), Enterprise Architect – это программа для UML-моделирования и проектирования нового поколения. Вот фраза из их рекламных материалов:

WELCOME to the next generation in UML modeling and design software! At Sparx Systems, we realize that because you want to remain competitive, you need to be **productive**. You need to have your **whole team** perfectly equipped with the **very latest** trouble-free UML modeling software. In other words, you need the most reliable, capable and progressive business modeling software, that won't break the budget.

Enterprise Architect существует в вариантах для Windows и Linux и является действительно неплохим средством для UML-моделирования, с возможностью многопользовательской работы и дружественным интерфейсом. С EA отлично интегрируется другой продукт Sparx Systems – Zicom Mentor. Zicom Mentor – это простая и понятная утилита, представляющая собой словарь/ассистент по UML, который отлично интегрируется с EA. Zicom Mentor ответит на Ваши вопросы, поможет получить и проверить Ваши знания, начать новый проект. Zicom Mentor включает интерактивные курсы, электронные книги и тесты, множество справочной информации по UML.

Poseidon for UML [8] – это популярное CASE-средство для UML-моделирования. Poseidon берет свое начало из открытого проекта ArgoUML и на данный момент уже является признанным профессионалами пакетом. Сейчас это быстро развивающееся сообщество пользователей, которые используют Poseidon для проектирования серьезных приложений. Poseidon for UML существует в нескольких редакциях, чтобы покрыть потребности разных категорий пользователей. Особой похвалы заслуживает сайт приложения, который содержит много полезной информации по Poseidon, UML, ООАП вообще, тесты и курсы по UML, плагины, ООАП-гlossарий и многое другое. Poseidon известен своим потрясающим удобством (usability), имеет бесплатную версию и является, с точки зрения автора, лучшей программой такого рода.

SmartDraw [9] – это простая и дружественная альтернатива MS Visio. Субъективно, возможности SmartDraw на порядок превышают возможности MS Visio. Недостатками этого пакета является его цена и архаичный интерфейс в стиле Win'95. Преимущество над конкурентами – беспрецедентная поддержка пользователей.

Dia [10] – это gtk+ базированная программа создания диаграмм, распространяемая под GPL-лицензией. Dia создавалась по подобию коммерческой программы Visio. Она может быть использована для рисования многих видов диаграмм. Dia уже в состоянии, пригодном для использования, но продукт все время развивается. А главное – Dia абсолютно бесплатна!

Подводя итоги, автор попытался сравнить рассмотренные приложения по нескольким критериям и представил результаты в виде таблицы.

Наименование	Возможности визуализации	Простота	Удобство интерфейса	Кратко...
IBM Rational Rose	10	0	1	стандарт де-факто
Borland Together	10	2	2	удобное интегрированное решение
Microsoft Visio	8	3	2	отличная интеграция с MS Office
Enterprise Architect	10	4	9	мощное и удобное средство
Poseidon	9	3	10	удобство использования
Smart Draw	8	10	5	простота освоения
Dia	5	10	5	минимум необходимой функциональности

Естественно, рассмотренные здесь пакеты – очень малая часть всего доступного в Интернете ПО для визуального моделирования с помощью UML. Список другого ПО для создания UML-диаграмм можно найти, например, на http://www.objectsbydesign.com/tools/umltools_byCompany.html. На <http://www.uml.org/#Links-UML2Tools> есть ссылки на другие каталоги подобных программ. А на <http://www.gskinner.com/gmodeler/app/run.html> Вы найдете уникальное в своем роде *онлайнное средство UML-проектирования*.

Литература:

1. <http://www.omg.org/cgi-bin/apps/doc?formal/03-03-01.pdf>
2. <http://www.intel.ru>
3. <http://interface.ru>
4. <http://www-306.ibm.com/software/rational/>
5. <http://www.borland.com/together/>
6. <http://office.microsoft.com/en-us/FX010857981033.aspx>
7. <http://www.sparxsystems.com.au/>
8. <http://www.gentleware.com/>
9. <http://www.smartdraw.com/>
10. <http://www.gnome.org/projects/dia/home.html>

ИСПОЛЬЗОВАНИЕ БРАНДМАУЭРОВ ДЛЯ ЗАЩИТЫ LINUX-СЕРВЕРОВ

А.Н. Бакал

г. Киев, Национальный педагогический университет им. М.П. Драгоманова
vd34@ukr.net

Весь трафик в сети состоит из пакетов. Каждый пакет состоит из двух частей: заголовка и тела. В заголовке пакета находятся данные об источнике, типе пакета, и другая информация, характерная для пакетов определённых типов. Тело пакета содержит ту информацию, которую мы хотим передать [1].

Брандмауэр (firewall) – системный элемент, обеспечивающий защиту сети от несанкционированного доступа [1].

По сути, брандмауэр представляет собой фильтр пакетов: он просматривает заголовок каждого проходящего через него пакета, а потом решает, что делать со всем пакетом. Фильтрация пакетов встроена в ядро ОС Linux.

В новых дистрибутивах Linux (ядро 2.4.x и старше) стандартным брандмауэром, выполняющим данные функции, является IPTables.

Обычно IPTables (IPChains – в дистрибутивах с версией ядра 2.2.x) используется на шлюзах, соединяющих две сети, например, локальную сеть и Internet. При этом мы имеем право разрешить передавать или принимать какие-либо пакеты. Это позволяет обеспечить (в сочетании с другими методами) желаемый уровень безопасности.

Применение IPTables позволяет обезопасить сервер от таких видов атак:

- фрагментация пакетов;
- IP-спуфинг.

Для обеспечения поддержки брандмауэра IPTables необходимо перекомпилировать ядро. Технология настройки и компиляции ядра описана в [2]. При этом необходимо включить опции ядра: IP:firewalling, IP:firewall packet, IP:always defragment, IP:masquerading и IP-Forwarding. Более подробную информацию о настройке ядра Linux для поддержки IPTables можно найти в [3].

Рассмотрим кратко принцип работы IPTables. Ответим на вопрос: как пакеты проходят через фильтр? При запуске ядру доступны три firewall-цепочки:

- INPUT (входящие пакеты);
- OUTPUT (исходящие пакеты);
- FORWARD (перенаправление пакетов).

Цепочка (firewall-цепочка) – это набор правил вида «заголовок пакета: действие».

Все цепочки объединяются в таблицы (отсюда и происходит название

брандмауэра). Главной таблицей, в которой хранятся правила обработки всех стандартных типов трафика, является таблица `filter`. К стандартным таблицам относятся также таблицы `nat` (используется для построения NAT-преобразователей) и `mangle` (определяет типы преобразования некоторых пакетов). Все команды, которыми задают или изменяют правила фильтрации пакетов, соответствующим образом автоматически изменяют или дополняют таблицу `filter`.

При поступлении входящего пакета принимается решение: принять или уничтожить пакет. Ядро принимает решение, поочерёдно просматривая все правила цепочки. Если заголовок пакета соответствует заголовку, указанному в правиле, выполняется заданное в правиле действие. Если нужное правило отсутствует, то пакет обрабатывается согласно правилу, установленному по умолчанию для всей цепочки. Правило, используемое для цепочки по умолчанию, называется правилом `policy` и в целях безопасности должно содержать значение `DROP` – уничтожить пакет.

Сделать это можно, используя команду `iptables` с параметром `-P`:

```
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
```

Таким образом задаются правила по умолчанию для трёх стандартных цепочек, содержащихся в таблице `filter`.

В качестве действия также можно указывать:

- `ACCEPT` – предписывает пропускать все пакеты;
- `REJECT` – также (как и `DROP`) предписывает отвергать все пакеты, но при этом источнику сообщения будет отправлено соответствующее уведомление.

Помимо использования параметра `-P`, над цепочками возможны и другие действия (параметры команды `iptables`):

- `-N` – создать новую цепочку;
- `-A` – добавить новое правило и др. [3].

Итак, рассмотрим теперь практический пример конфигурирования брандмауэра `IPTables` для настройки шлюза.

Перечислим функции, которые должен выполнять шлюз:

1. Поддержка связи с провайдером.
2. Маршрутизация IP-пакетов между локальной сетью и сетью Internet для выхода пользователей локальной сети в Internet.
3. Обеспечение IP-сервиса.
4. Защита локальной сети от несанкционированного доступа из Internet.

Конфигурирование шлюза в ОС Linux состоит из следующих этапов:

1. Настройка ядра.
2. Настройка сети.
3. Конфигурирование `IPTables`.
4. Настройка DNS.

5. Настройка Squid.

В данном примере будет использоваться для сетевых интерфейса – eth0, идущий к провайдеру, и eth1 – во внутренней сети. Назначим интерфейсу eth0 IP-адрес 111.111.111.111, а eth1 – 192.168.1.1.

Теперь создадим цепочку, через которую пойдет весь трафик от провайдера:

```
iptables -N provider
iptables -A INPUT -i eth0 -j provider
```

Запретим ip-spoofing:

```
iptables -A provider -s 192.168.1.1/16 -j DROP
iptables -A provider -s 127.0.0.1/8 -i -j DROP
```

Запретим telnet и ssh снаружи:

```
iptables -A provider -p tcp --destination-port 23 -j REJECT
iptables -A provider -p tcp --destination-port 22 -j REJECT
```

Запретим samba наружу:

```
iptables -A provider -p tcp --destination-port 137 -j REJECT
iptables -A provider -p udp --destination-port 137 -j REJECT
```

То же самое сделаем для портов 138, 139, 129. Подробно о настройке samba можно прочитать в Samba-HOWTO.

Создадим цепочку для подсчета трафика:

```
iptables -N traffic
iptables -I INPUT -i eth0 -s ! 123.123.123.0/24 -p all -j traffic
iptables -A traffic -d 123.123.123.123
```

Для того, чтобы правила были постоянными (при перезагрузке PC правила IPTables теряются), воспользуемся скриптами iptables-save и iptables-restore:

```
# iptables-save > /etc/iptables.rules
```

Далее создадим скрипт управления пакетной фильтрацией:

```
# ! /bin/sh
# Если правил нет, то ничего не делать
[ -f /etc/iptables.rules ] || exit 0
case "$1" in
start)
echo -n "Включение пакетной фильтрации: "
/sbin/iptables-restore < /etc/iptables.rules || exit 1
echo 1 > /proc/sys/net/ipv4/ip-forward
echo "." ;;
stop)
echo -n "Отключение пакетной фильтрации: "
echo 0 > /proc/sys/net/ipv4/ip-forward
/sbin/iptables -X
/sbin/iptables -F
/sbin/iptables -P INPUT ACCEPT
/sbin/iptables -P OUTPUT ACCEPT
/sbin/iptables -P FORWARD ACCEPT
echo "Использование: /etc/init.d/packetfilter {start | stop}"
exit 1 ;;
```

```
esac
exit 0
```

И, наконец, этот скрипт необходимо добавить в сценарии загрузки системы.

Функциональность скрипта и настроек брандмауэра проверена автором в дистрибутиве Mandrake Linux 10.

Литература:

1. Шиндер Дебра Л. Основы компьютерных сетей. –М.: Cisco Press.
2. Ward Brian. Linux Kernel-HOWTO. V.0.80
(<http://www.math.psu.edu/ward/>)
3. Колисниченко Д.Н. Linux-сервер своими руками. – 2-е изд., перераб. и доп. – СПб.: Наука и техника, 2004.

ОРГАНІЗАЦІЙНО-ПЕДАГОГІЧНІ УМОВИ ФОРМУВАННЯ ІНФОРМАЦІЙНОЇ КОМПЕТЕНТНОСТІ

Н.В. Баловсяк

м. Чернівці, Чернівецький торговельно-економічний інститут
Київського національного торговельно-економічного університету
nadia123@mail.ru, nadia@chtei.cv.ua

Використання інформаційних технологій в освіті та в професійній діяльності спричиняє виникнення нових підходів до результату освіти, що стосується оволодіння роботою з комп'ютерними, інформаційними та телекомунікаційними технологіями. Метою статті є визначення змісту і структури інформаційної компетентності як результату освіти у загальному контексті професійної підготовки фахівця.

Коли ведеться мова про роботу з інформацією та інформаційними технологіями, в літературі використовується декілька понять.

Поряд з поняттям “інформаційна компетентність” часто використовуються такі поняття, як “комп'ютерна компетентність”, “комп'ютерна грамотність”, “технологічна грамотність”, “інформаційна грамотність”, “інформаційно-технологічна компетентність”, “інформаційна культура”. При цьому слід відмітити різний підхід авторів до трактування цих понять.

Розглянемо, що означають ці поняття.

В.А. Дарлінгер визначає **комп'ютерну компетентність** як рівневу освіту, що характеризує професійну підготовку фахівця до використання інформаційно-комунікаційних технологій на теоретичному, практичному і творчому рівні [4].

Комп'ютерна компетентність і комп'ютерна грамотність передбачають лише формування навичок роботи з комп'ютерною технікою, а не з інформаційними технологіями взагалі [3]. Формування у студентів комп'ютерної грамотності переслідує тільки одну мету – прагматичну, але при цьому упускається інша – загальноосвітня, яка полягає в освоєнні студентами фундаментальних понять сучасної інформатики.

В дослідженнях американської асоціації бібліотекарів для позначення знань, умінь і навичок студентів при роботі з інформацією, інформаційними та комп'ютерними технологіями використовують поняття **комп'ютерна грамотність, технологічна грамотність та інформаційна грамотність**.

Комп'ютерна грамотність – це механічне знання основ роботи певних апаратних і програмних засобів.

Технологічна грамотність – це розуміння основних концепцій технологій та їх застосування.

Американська асоціація бібліотекарів визначає **інформаційну грамотність** як набір здібностей, який є у індивідуумів, і передбачає уміння визнати необхідність інформації, уміння її оцінити і ефективно використати [1].

Тобто інформаційна грамотність – це навички роботи з інформацією у різних формах її представлення.

Відмінність між комп'ютерною та технологічною грамотністю полягає в тому, що комп'ютерна грамотність в основному акцентується на вмісті, аналізі і інформаційному пошуку, зв'язку і оцінці, технологічна грамотність передбачає глибоке розуміння технологій як взагалі так і технологій роботи з конкретними програмними продуктами. В цьому випадку інформаційна грамотність являє більш широку область компетентності [1].

Іноді також зустрічаються поняття “**технологічна грамотність**” і “**інформаційно-технологічна компетентність**”.

П.В. Беспалов використовує поняття **інформаційно-технологічна компетентність** для позначення взаємодії фахівця з інформацією та інформаційними технологіями.

Інформаційно-технологічна компетентність (ІТК) розглядається як інтегральне особистісне утворення, що характеризує зрілу особистість людини сучасного інформаційного суспільства й таке, що охоплює три основні підструктури особистості:

- мотивацію (до засвоєння і застосування комп'ютерних інформаційних технологій);
- здатності (до сприйняття, розумової переробки й обміну з іншими людьми комп'ютерних інформаційних технологій);
- досвід (знання про комп'ютерні інформаційні технології й уміння їх застосовувати) [2].

Отже, підсумовуючи викладене вище, визначимо інформаційну компетентність як сукупність компетенцій, пов'язаних з роботою з інформацією у всіх її формах та представленнях, які дозволяють ефективно користуватись інформаційними технологіями різного виду, як в традиційній друкованій формі, так і комп'ютерними та телекомунікаційними, працювати з інформацією у різних її формах і представленнях як у повсякденному житті так і в професійній діяльності.

Всі умови розвитку інформаційної компетентності студентів об'єднаємо в наступні групи:

1. Організаційно-педагогічні умови
2. Змістовні умови
3. Технологічні умови
4. Ціннісно-мотиваційні умови

Технологічні умови дозволяють сформувати навчальне середовище, в якому комп'ютерні та інформаційні технології та програмне забезпечення є невід'ємною частиною організації та функціонування навчального процесу. Застосування комп'ютерів у всьому навчальному процесі, особливо щодо вивчення тих технологій, які напряму не пов'язані із вивченням комп'ютерних технологій та програмного забезпечення є шляхом до формування компетенцій застосовності, особливо що стосується застосування

комп'ютера до розв'язання повсякденних задач. Технологічні умови забезпечують можливість цього застосування, і означають достатню кількість робочих місць за комп'ютером для студентів, вільний доступ у будь-який час до комп'ютерів, наявність необхідного програмного забезпечення.

Ціннісно-мотиваційні умови забезпечать формування необхідних психологічних якостей, за наявності яких можливе формування інших складових інформаційної компетентності. Ціннісно-мотиваційні умови направлені на формування ціннісної та потребово-мотиваційної сфери особистості.

Змістовні умови формування інформаційної компетентності визначають вимоги до змісту навчальних дисциплін в цілому, та до конкретних видів навчальної діяльності як то змісту навчальних задач, вправ, перевірочних завдань.

Організаційно-педагогічні умови формування інформаційної компетентності майбутнього економіста визначаються застосуванням таких форм і методів організації навчального процесу, які дозволять сформувати компетенції, що є елементами інформаційної компетентності майбутнього економіста, а в поєднанні із змістовними та технологічними умовами призведуть до формування об'єктивних характеристик особистості, які входять в модель інформаційної компетентності майбутнього економіста.

Організаційно-педагогічні умови сприяють:

- 1) формуванню компетентностей роботи з інформацією ;
- 2) формуванню здатностей застосування комп'ютерних та інформаційних технологій у розв'язанні різного роду задач (професійних та повсякденних).

Виходячи з цілей застосування організаційно-педагогічних умов у формуванні інформаційної компетентності майбутнього економіста до їх складу повинні входити наступні елементи:

- 1) умови формування інформаційної складової інформаційної компетентності майбутнього економіста;
- 2) формування операційного компоненту складової застосовності;
- 3) елементи загальної організації педагогічного процесу для забезпечення оволодіння визначеними в структурі інформаційної компетентності майбутнього економіста компетенціями;
- 4) особливості організації педагогічного процесу дисциплін, основним об'єктом та предметом вивчення яких є комп'ютер та програмне забезпечення.

Організаційно-педагогічні умови представимо у вигляді сукупності умов, одні з яких описують педагогічні технології, які застосовуватимуться в організації всього навчального процесу (*організаційні умови*), а інші являтимуть особливості організації конкретних видів навчальної діяльності (*педагогічні умови*).

До педагогічних умов формування відносяться застосування відповідних технологій навчання. Однією з технологій, яка є обов'язковим елемен-

том організації навчального процесу, є новітні технології навчання, описані в технологічних умовах формування інформаційної компетентності майбутнього економіста.

Ця технологія є базовою, вихідною, і решта технологій є похідними, такими що базуються і в повній мірі використовують нові інформаційні технології навчання.

Організаційні умови визначають загальні технології формування інформаційної компетентності майбутнього економіста в організації всього навчального процесу, які виражаються у застосуванні певних педагогічних технологій. Педагогічні умови розкривають особливості застосування цих педагогічних технологій, тобто організаційних умов для конкретних видів навчальної діяльності та форм організації навчального процесу.

Конкретні зміст, форми та методи застосування організаційних умов формування інформаційної компетентності описані в педагогічних умовах формування інформаційної компетентності майбутнього економіста.

Загальними педагогічними технологіями, що використовуються у формуванні інформаційної компетентності майбутнього економіста є наступні:

- технологія розвивального навчання;
- технологія проблемного навчання;
- проектна технологія.

I. Технологія розвивального навчання у вузі є однією з визначальних, вона дозволяє сформувати самостійну творчу особистість. Використання цієї технології дозволяє забезпечити індивідуальний підхід до кожного студента, особливо це важливо на молодших курсах за умов різної базової підготовки студента. Шляхами реалізації цієї технології є формулювання проблеми, для розв'язання якої розглядається тема, використання різнорівневих індивідуальних завдань для студентів, формування умінь застосовності.

II. Технологія проблемного навчання дозволяє сформувати цілий ряд як психологічних якостей особистості, так і здатностей та умінь, які є елементами інформаційної компетентності майбутнього економіста. Постановка проблеми дозволяє сформувати у студента навчальні (а у більш загальному вигляді та на старших курсах – наукові) проблеми, що призводить до формування інформаційної потреби та мотивації щодо вивчення певних розділів навчальної програми. Використання проблемного навчання призводить до формування знань та умінь не стільки як результату навчальної діяльності, скільки як набору дій для розв'язання певної проблеми – навчальної, наукової чи особистої, не пов'язаної із професійною діяльністю. Тобто використання технології проблемного навчання дозволяє сформувати компетенції застосовності.

III. Проектна технологія. Зміст та основні положення цієї технології застосовуються у вищій школі давно. Ця технологія, яка визначає таку організацію педагогічного процесу, яка дозволяє через проектну діяльність,

тобто діяльність по створенню цілісного об'єкту як результату навчальної діяльності, що вимагає набору певних знань та умінь, передбачає розв'язання цілого ряду задач. Подібними видами діяльності у вищій школі є курсові та дипломні проекти, в контексті формування інформаційної компетентності майбутнього економіста використання проектної технології можливе не лише при написанні курсових та дипломних робіт. Це дозволяє сформуванню інформаційну складову інформаційної компетентності майбутнього економіста.

Сукупне застосування всіх умов формування інформаційної компетентності майбутнього економіста дасть змогу отримати бажані результати.

Література:

1. Information Literacy Competency Standards for Higher Education (<http://www.ala.org/ala/acrl/acrlstandards/informationliteracycompetency.htm>)

2. Беспалов П.В. Акмеологический подход к формированию и развитию информационно-технологической компетентности государственных служащих // Материалы конференции «Информационные технологии в образовании-2003» (<http://ito.edu.ru/2003/II/3/II-3-2414.html>)

3. Голунова Л.В. Научно-теоретическое обоснование понятия "информационная грамотность" // Всероссийская научная конференция "Наука и образование". г. Белово, 12-13 апреля 2002 г. (http://belovo.kemsu.ru/conferens/conferens1/tezis/Sek5_1/26.html)

4. Далингер В.А. Компьютерная компетентность – основа профессионализма современного учителя математики // Материалы конференции «Информационные технологии в образовании-2003» (<http://ito.edu.ru/2003/II/3/II-3-1788.html>)

АВТОМАТИЗОВАНИЙ КОНТРОЛЬ ЗНАТЬ В СИСТЕМІ УПРАВЛІННЯ НАВЧАЛЬНО-ПІЗНАВАЛЬНОЮ ДІЯЛЬНІСТЮ СТУДЕНТІВ

В.Ю. Баранов

м. Одеса, Одеська національна академія зв'язку імені О.С. Попова

bwr@ukr.net

Педагогічне тестування в нинішній час стає однією з найбільш актуальних інформаційних технологій освіти. Про це, зокрема, свідчать дослідження українських та російських вчених В.С. Аванесова, В.П. Беспалько, І.Є. Булах, С.Г. Зайцева, В.І. Карлашука, А.А. Киверялга, Г.У. Матушанського, Ю.І. Машбиця, Н.М. Розенберга, Н.Ф. Тализіної, Г.В. Фролової, а також іноземних вчених Г. Вітцлака, П. Клайна, Е. Стоунса, Р. О'Лірі, Дж.П. Семпсона-мол., Д. Стівенса. В умовах різкого прискорення темпів розвитку суспільства сучасний фахівець зобов'язаний постійно підвищувати кваліфікацію, засвоювати нові знання. Сприяє цьому процесу використання інструментів вимірювання навчальних досягнень. Кваліфіковано розроблена програма автоматизованого контролю знань здатна водночас забезпечити реалізацію основних функцій контролю та звільнити викладача від великого обсягу рутинної праці, і, таким чином, зберегти його час для творчої роботи з студентами. Не менш важливим є те, що використання засобу автоматизованого контролю дозволить студенту самостійно виявляти недоліки у структурі своїх знань і працювати додатково для ліквідації недоліків. У випадку дистанційного навчання автоматизований контроль дозволить забезпечити зворотний зв'язок між викладачем та студентами.

Однією з тенденцій розвитку освіти в країнах Заходу є передача комп'ютеру функції перевірки знань студентів та учнів (про це свідчать, зокрема, роботи німецького вченого Г. Вітцлака, англійських П. Клайна, Р. О'Лірі, Д. Стівенса та американського Е. Стоунса). Зокрема, у США щорічно знання та здібності у всіх верствах населення перевіряються за допомогою 47 мільйонів тестових бланків, що випускаються 400 тестовими компаніями.

Завданням педагогічної науки є розробка методик (та – разом з фахівцями з інформатики – відповідних програмних засобів), які б гарантували педагогічну ефективність функціонування засобів автоматизованого контролю знань в системі управління навчально-пізнавальною діяльністю студентів.

Одною з основних проблем автоматизованого контролю знань студентів вже більш 40 років залишається «проблема вільних відповідей» [1]. Саме у вільній формі студенти звичайно відповідають на питання не тільки усних, але й багатьох письмових іспитів. І саме такі відповіді дають викладачу максимум інформації щодо знань студента. Крім того, цікаві питання заохо-

чують студентів до виконання завдання, активізують і стимулюють їхню розумову діяльність. Для того, щоб суттєво підвищити якість автоматизованого контролю знань та задовольнити основним психолого-педагогічним вимогам щодо нього, необхідно наблизити можливості комп'ютерного оцінювання відповідей студентів до можливостей викладача. Тобто потрібно зробити так, щоб комп'ютер задавав запитання студентові у формі, наближеній до тієї, у якій формулює свої усні чи письмові запитання досвідчений викладач; а головне – оцінював би відповіді, наведені також у майже довільній формі. Однак у переважній більшості програм комп'ютерного контролю знань основною формою відповіді є вибір з переліку кількох можливих відповідей на тестову задачу, серед яких є як правильні, так і неправильні відповіді.

Безумовно, існує певний клас задач, які доцільно подавати саме у такій формі меню. Але така форма запитань має серйозні психолого-педагогічні вади, про що зокрема свідчать дослідження В.П. Беспалько, П. Клайна, Ю.І. Машбиця, Н.М. Розенберга, А.О. Смирнова, Н.Ф. Талізіної та інших. До числа недоліків цієї форми в першу чергу слід віднести істотне обмеження самостійної діяльності студента. Якщо при звичайному опитуванні студентові потрібно самостійно формулювати свої думки, то при роботі з меню відповідей він лише вибирає з запропонованих готових варіантів.

Без відкритих завдань стає набагато важчим чи взагалі неможливо отримати адекватну оцінку рівня знань студента. Існуючі програми комп'ютерного контролю знань, як правило, пропонують студенту кілька десятків завдань, тоді як за умов усного чи письмового опитування викладачу потрібна суттєво менша їхня кількість. З цих причин форма запитань, яка ґрунтується лише на виборі, не може бути не тільки єдиною формою завдань з сеансу комп'ютерного тестування, але й основною. Отже, необхідно вивчати способи розширення можливостей автоматизованого контролю знань, за рахунок використання нетрадиційних методів оцінки відповідей.

Розглянемо принципи оцінювання відповідей, які застосовано у системі комп'ютерного контролю знань CONTROL, розробленій на кафедрі інформаційних технологій ОНАЗ ім. О.С. Попова [2], [3]. З цієї метою наведемо визначення ряду понять з теорії комп'ютерного контролю знань, які сформульовано виходячи з оригінального підходу, застосованого при розробці системи CONTROL Г.В. Піддубним та Л.Л. Леоненко [4]. Зокрема, систему комп'ютерного контролю будемо називати універсальною, якщо вона не передбачає обмежень ні типів тестових задач, ні форми подання відповідей на ці задачі. До відповідей різного типу в універсальній системі контролю застосовуються різні методи, або правила ідентифікації (правила висновку). Універсальна система передбачає розширення своїх можливостей щодо оцінки відповіді за рахунок приєднання нових правил висновку. Основні типи правил висновку, що найчастіше вживаються для ідентифікації відповідей,

поділено на три великі класи: 1) правила порівняння чисел, числових множин та послідовностей, 2) правила порівняння формул та 3) правила порівняння текстів.

Кожне правило висновку, що надає оцінку відповіді студента, повинно базуватися на певній формальній теорії подібності висловлювань відповідної структури. В системі CONTROL для забезпечення механізму роботи різних правил порівняння текстів використовується теорія подібності текстів розроблена Г.В. Піддубним [5]. Серйозною перевагою правил порівняння текстів, основаних на теорії подібності, є можливість допускати модифікації, скорочення і незначні помилки в словах текстів, що порівнюються. Однак теорія подібності мало пристосована до оцінки структури фрази. Фактично вона виходить з припущення, що якщо «вільна» відповідь містить всі необхідні (або ж всі «важливі») слова, вона може вважатися правильною незалежно від порядку слів. Часто це припущення є прийнятним, однак також часто правильність відповіді залежить від її структури, а не тільки лексичного складу. Використання розробленого алгоритму ідентифікації на основі подібності текстів можна зробити більш ефективним, якщо врахувати специфіку ідентифікації відповідей неелементарної структури. Для оцінки відповіді, що складається з частин, які комп'ютер здатний розрізнити, кожен з частин можна оцінити окремо, а загальну оцінку отримати в результаті наступного об'єднання часткових оцінок. Алгоритми пошуку в відповіді її складових частин істотно залежать від характеру останніх. Розширення множини форм відповідей, які розпізнаються автоматизованою системою контролю знань, можливе за рахунок включення різних типів відповідей складної структури, зокрема семантично та синтаксично різномірних. Методи оцінювання відповідей відповідної структури, реалізовані у нових, розроблених автором, правилах висновку [6], [7].

Система CONTROL, яка використовує раніше розроблені, а також запропоновані автором правила висновку, здатна розрізнити правильні і неправильні відповіді, зокрема, на наступні питання з курсу «Інформатика»:

Чому не можна використовувати оперативний запам'ятовуючий пристрій для тривалого зберігання програм і даних?

Змінні яких базових типів мови PASCAL можна використовувати як параметри циклу FOR?

Що таке вказівник (POINTER)? Дайте визначення цього поняття.

Яку задачу виконує наступний фрагмент програми?

```
...  
var S : string; k : byte;  
begin  
S:=Edit1.Text;  
k:=Length(S);  
while (k>0) and (S[1]=' ') do  
begin DELETE(S, 1, 1); k:=k-1 end;
```

...

Відповідаючи системі CONTROL на подібні питання, студент може особисто формулювати свої думки. Викладач має широкі можливості настройки допустимих меж «варіацій» відповіді. Зрозуміло, викладач не перераховує всілякі варіації відповіді, а використовує надані системою засоби «охоплення» їх єдиним «описувачем допустимої відповіді».

У 2002–2003 рр. проводився педагогічний експеримент, у якому взяли участь усі студенти 1-го курсу ОНАЗ ім. О.С. Попова (515 студентів з 22 груп). Враховуючи отримані у ході проведення експерименту дані [8], можемо стверджувати, що нові правила висновку можуть знайти широку сферу застосування у системах комп'ютерного контролю, побудованих за принципами, подібними до тих, що реалізовано у системі CONTROL. Крім того, результати експерименту дозволяють зробити висновок, що використання комп'ютерного контролю за допомогою системи CONTROL має певні суттєві переваги відносно «паперової технології» перевірки знань. Комп'ютерна технологія не потребує перевірки виконаних завдань викладачем, не вимагає великої кількості друківаних матеріалів, а також надає студентам можливість майже необмеженого за часом тренування.

Отримані результати експерименту дозволяють рекомендувати викладачам вищих навчальних закладів використовувати систему комп'ютерного контролю CONTROL в якості: 1) зручного і корисного тренажера, придатного для закріплення і повторення вивченого матеріалу; 2) ефективного засобу автоматизованого підсумкового та поточного контролю знань. Застосування системи CONTROL у зазначених функціях повинно сприяти більш ефективному використанню робочого часу викладачів, який буде витрачатися більшою мірою не на перевірку виконаних студентами завдань, а на додаткове консультування студентів з складних питань курсу.

Література:

1. Баранов В.Ю. Формы ответов при автоматизированном контроле знаний студентов // Научный вестник ПДПУ ім. К. Д. Ушинського. – Одеса: ПДПУ, 2004. – № 7. – С. 17–24.
2. www.control.hotmail.ru.
3. Баранов В.Ю., Леоненко Л.Л. Компьютерное тестирование знаний: об алгоритмах оценки «свободных» ответов // Информационные технологии и информационная безопасность в науке, технике и образовании «Инфортех-2004»: Материалы межд. науч.-практ. конф. – Севастополь-Киев: НТО РЭС Украины, 2004. – С. 163-171.
4. Леоненко Л.Л., Піддубний Г.В. Комп'ютерна технологій тестування знань (автоматизована система оцінювання знань «Control»): Рукопис посібника для викладачів середніх та вищих учбових закладів. – Одеса, 1999. – 186 с.

5. Леоненко Л.Л., Поддубный Г.В. Теория подобия конечных последовательностей и ее приложение к распознаванию образов // Автоматика и телемеханика. – М., 1996. – №8. – С. 119–131.

6. Баранов В. Ю. Методи оцінювання семантично різномірної відповіді у системі комп'ютерного тестування знань // Комп'ютерно-орієнтовані системи навчання: Зб. наук. праць. – К.: НПУ ім. М. П. Драгоманова, 2002. – Вип. 5. – С. 97–110.

7. Баранов В. Ю. Оценка синтаксически разнородного ответа в системе компьютерного тестирования знаний // Науковий вісник ПДПУ ім. К.Д. Ушинського. – Одеса: ПДПУ, 2001. – Вип. 10-11. – С. 29–34.

8. Баранов В. Ю. Комп'ютерне тестування з інформатики: підсумки педагогічного експерименту в Одеській національній академії зв'язку // Теорія та методика навчання математики, фізики, інформатики: Зб. наук. праць. Вип. 4: В 3-х томах. – Кривий Ріг: НМетАУ, 2004. – Т. 3: Теорія та методика навчання інформатики. – С. 6-12.

НАВЧАЛЬНІ ДОСЛІДЖЕННЯ ПРИ ВИВЧЕННІ МЕТОДІВ ОБЧИСЛЮВАЛЬНОЇ МАТЕМАТИКИ

Л.І. Білоусова, Т.В. Белявцева, О.Г. Колгатін, Л.С. Колгатіна
м. Харків, Харківський національний педагогічний університет
ім. Г.С. Сковороди
Lib215@list.ru

Постановка проблеми. У підготовці майбутніх фахівців в області математики курс чисельних методів відіграє значну роль, оскільки при його вивченні студенти опановують способи і засоби розв'язування тих математичних задач, що виникають на практиці і непідвласні строгим методам чистої математики.

Курс чисельних методів можна розглядати як своєрідний “місток” між логічно вивіреними математичними теоріями і реальністю. Аналізуючи чисельні методи, легко помітити, що вони часто являють собою прямий наслідок з теорем чистої математики, їхню проекцію на практичні задачі. Серед них є методи настільки прості й очевидні, що їх можна вивести не з теоретичних посилок, а попросту спираючись на здоровий глузд чи геометричну інтерпретацію задачі. Однак, є і такі методи, що вражають уяву оригінальністю і своєрідністю ідеї, нестандартністю підходу до розв'язування задачі.

Постановка курсу чисельних методів являє собою досить складну проблему. Це зумовлено низкою факторів, з яких наведемо основні.

Теоретична частина курсу досить важка для сприйняття студентами, оскільки обґрунтування чисельного методу, з одного боку, вимагає широкого залучення апарату чистої математики з різних її областей; з іншого боку, математична основа чисельних методів ґрунтується на оцінках, що не завжди виглядають досить переконливими. Більш того, багато з них студент повинен прийняти на віру, тому що їхнє послідовне виведення виходить за межі навчального курсу і найчастіше навіть не наводиться в підручниках.

Усе сказане вище ускладнюється ще і тією обставиною, що поряд з теоретично встановленими нормами застосування того чи іншого методу існують і практичні правила – “неписані закони”, що не мають строгого обґрунтування, але якими проте зручно і доцільно керуватися на практиці. Згідно з цими правилами встановлюється реальна сфера дії чисельного методу, що звичайно виходить за рамки тієї, котра визначена теорією; умови застосовності методу одержують конкретизацію з врахуванням реальних технічних можливостей, а для контролю обчислювального процесу й оцінювання досягнутої точності рішення задачі пропонуються досить прості прийоми і співвідношення.

Використання практичних правил дозволяє додати процедурі застосування чисельного методу технологічність. Разом з тим, недоведеність практичних правил залишає деякий сумнів у їхній правомірності, усунути який

дозволяє лише досвід багаторазового контрольованого застосування чисельного методу – той самий досвід, що і породив ці правила.

Слід зазначити також, що світ чисельних методів надзвичайно різноманітний, кожен з них має свою специфіку, свою область ефективного застосування, тому основною задачею обчислювача є правильний вибір методу, найбільш придатного для розв'язування поставленої конкретної задачі, вміле сполучення різних методів на різних етапах її розв'язування, для чого вимагаються не тільки і не стільки теоретичні знання в галузі чисельних методів, скільки інтуїція, що здобувається в міру нагромадження знову ж такі особистого досвіду застосування цих методів.

Таким чином, курс чисельних методів, у силу свого явно вираженого практичного характеру, з необхідністю має спиратися на лабораторний практикум, якість постановки якого значною мірою визначає результати навчання за курсом у цілому.

Метою даної роботи є висвітлення цілей, способу і результатів реалізації навчально-дослідницького лабораторного практикуму з чисельних методів.

У стандартній постановці лабораторний практикум з чисельних методів зводиться до виконання розрахунків, необхідних для розв'язування задачі за відомим алгоритмом. Використання засобів обчислювальної техніки дозволяє цю роботу полегшити або автоматизувати, однак, у будь-якому випадку, коли це використання здійснюється на рівнях, що не виходять за рамки виконання обчислень або програмування, діяльність студента зводиться до відтворення алгоритму методу і кропіткої роботи з числами, що фактично призводить до заміщення змістовної задачі рутинною роботою.

У такому режимі за час, що відводиться на вивчення курсу, вдається лише випробувати окремі методи на прикладі розв'язування якої-небудь однієї задачі. У такому усіченому і, можна сказати, збитковому виді курс чисельних методів утрачає свою привабливість і внутрішню красу і, цілком природно, виявляється нудним і нецікавим для студентів.

Наше глибоке переконання полягає в тому, що істотних змін у постановці курсу чисельних методів і, як наслідок, у математичній підготовці студентів, можна досягти лише перетворенням лабораторного практикуму на цикл навчальних досліджень. При цьому дуже істотними є дві обставини: навчальні дослідження не вкрапляються окремими епізодами в тканину практикуму, а складають сутність кожної лабораторної роботи; використання обчислювальної техніки здійснюється на рівні середовища підтримки професійної математичної діяльності.

Перша обставина змушує переглянути весь курс, надавши лекціям характеру тематичних оглядів, а практикуму – систематичності, що є необхідною умовою для поетапного розвитку, поглиблення й ускладнення навчальних досліджень студентів з опорою на набутий досвід такої діяльності та дослідницькі уміння і навички, які формуються.

Необхідно відзначити, що епізодичне використання навчальних досліджень у лабораторному практикумі за принципом "час від часу" недоцільно. Практика показала, що в такому випадку студенти не усвідомлюють суті запропонованих їм завдань, а недостатній рівень дослідницьких умінь привносить у їхню діяльність елементи хаотичності і безсистемності. В решті більш привабливою формою проведення практикуму для більшості студентів виявляється звична робота за інструкціями.

Що стосується другої обставини, то орієнтація вузівського навчального процесу на використання сучасного професійного комп'ютерного інструментарію, а не на навчальні пакети, представляється найбільш доцільною. Така орієнтація, з одного боку, сприяє формуванню в студентів стійких навичок використання комп'ютера в професійних цілях, з іншого боку – визначає досить високий рівень постановки навчальних досліджень, відразу відтинаючи рутинну роботу.

Професійні пакети підтримки математичної діяльності, що одержали широке поширення, не розраховані на застосування в навчанні. Вони забезпечують розв'язання широкого кола стандартних математичних задач, залишаючи схованими від користувача використані для розв'язання методи. Разом з тим, такі пакети оснащені досить потужними і зручними вбудованими засобами, що дозволяють розширити функції пакета, у тому числі і такі, котрі пристосовують його для використання з метою навчання.

Для постановки навчально-дослідницьких робіт з курсу чисельних методів нами був узятий за основу пакет MathCAD, засобами якого був розроблений комплект динамічних опорних конспектів (ДОК'ів), що підтримують виконання таких робіт із усіх тем курсу. Таким чином, фактично студенту була надана віртуальна лабораторія для проведення обчислювальних експериментів.

Вибір пакета MathCAD зумовлений тим, що він широко застосовується для розв'язування прикладних задач математики і разом з тим йому притаманні такі якості, що дозволяють використовувати його в навчанні: можливість створення динамічної екранної сторінки, вільне переміщення курсору по екрану, досить розвинена вбудована мова і т.д. Створення ДОК'а в середовищі MathCAD зводиться до розробки програми, що реалізує алгоритм відповідного чисельного методу, і інтерфейсу, зручного для введення даних задачі і відображення на екрані процесу і результатів роботи алгоритму. Математичні можливості пакета були використані для оцінювання якості отриманих результатів.

Кожен ДОК орієнтований на роботу з одним з чисельних методів і надає можливість багаторазових випробувань цього методу на різних задачах з виведенням на екран результатів у числовій і графічній формі. Проводячи навчальне дослідження, студент здійснює серію таких випробувань і на підставі спостереження за обчислювальним процесом, шляхом аналізу його характеристичних показників робить висновки.

Необхідно відзначити, що задачі, розв'язувані студентом у ході навчального дослідження, істотно відрізняються від тих, котрі складають суть традиційної лабораторної роботи. Так, наприклад, при дослідженні чисельних методів розв'язування рівнянь студенту пропонується встановити, який критерій варто обрати для оцінки близькості знайденого наближення до шуканого значення кореня рівняння – точність, з якою це наближення задовольняє рівняння, чи точність, з якою це наближення повторює попереднє. У кожному дослідженні студенту пропонується вирішити такі задачі: експериментально оцінити порядок і швидкість збіжності методу; виділити основні фактори, що впливають на ці характеристики; встановити область ефективного застосування методу.

При дослідженні, наприклад, інтерполяційних формул, де, на перший погляд, усе ясно – чим більше вузлів інтерполяції, тим вище ступінь полінома, точніше наближення, – студент має переконатися в тому, що далеко не завжди це й справді так. Для досягнення потрібної точності іноді доцільно змінити тактику: замість нарощування вузлів використовувати дроблення проміжку інтерполяції. Студенту пропонується побудувати найкраще можливе наближення функції на відрізку по заданій на ньому обмеженій кількості її значень. Як варто розпорядитися цими даними? Який спосіб інтерполяції дасть найбільш надійний результат? Вивчаючи питання про точність відновлення значення функції в проміжній точці таблиці за інтерполяційними формулами, студент експериментально встановлює правило для вибору тих табличних значень, на які варто спиратися для мінімізації похибки і т.д.

Для того, щоб діяльність студента була осмисленою, націленою і забезпечувала досягнення прогнозованого навчального ефекту, нами було розроблено методичну підтримку практикуму у виді планів-звітів з кожної лабораторної роботи.

Плани-звіти виконані за єдиною схемою і складаються з двох частин – інформативної й інструктивної. В інформативній частині повідомляється тема роботи, її ціль, програмне забезпечення роботи, наводиться характеристика вхідних і вихідних числових і графічних даних.

Інструктивна частина містить порядок виконання роботи, де позначені і зафіксовані її ключові моменти. Для орієнтації студента на виконання дослідження йому спочатку пропонується ланцюжок відповідним чином підібраних питань. Деякі з них адресовані до інтуїтивних уявлень студента про досліджуваний метод, інші – на те, щоб наштотхнути його на думку про можливу помилковість таких уявлень. У ході обмірковування запропонованих питань студент одержує можливість зорієнтуватися в проблемі, усвідомити її та вибудувати робочу гіпотезу дослідження.

Уся наступна – основна – робота студента спрямована на перевірку, уточнення, конкретизацію гіпотези. Ця робота виконується за запропонованим планом, що визначає окремі етапи дослідження, задачі, що

розв'язуються на кожному етапі, експериментальний матеріал, який потрібно отримати, форму його подання і т.д. У міру просування практикуму інструкції студенту все менш деталізуються, здобуваючи характер рекомендацій. Деякі експерименти він повинний продумати, поставити і здійснити самостійно.

Для виконання кожної з лабораторних робіт підібрані індивідуальні варіанти комплектів задач, на яких пропонується випробувати метод для отримання експериментального матеріалу, що відповідає меті роботи. При бажанні студент може доповнити ці комплекти задачами за власним вибором.

Завершальним етапом дослідження є підведення його підсумків. Це пропонується зробити у вигляді висновків, контури яких з більшим чи меншим ступенем виразності намічені в плані-звіті. Підказки допомагають студенту зафіксувати результати роботи, структурувати їх, дозволяють звернути увагу на ті моменти дослідження, що можуть залишитися непоміченими.

Виконання запланованого дослідження дає студенту досить глибоке розуміння властивостей і специфіки застосування досліджуваного методу, і це повинно знайти відображення в "творі на вільну тему": придумати таку практичну задачу, для якої найбільш ефективним інструментом рішення є саме досліджуваний метод.

Зазначимо, що плани-звіти надаються студентам як у друкованому виді, так і в електронній формі. Остання використовується паралельно з ДОК'ом під час проведення лабораторної роботи, що зручно для перенесення експериментальних даних з ДОК'а в заготовлені таблиці, для підготовки звітних матеріалів.

Висновки. Досвід впровадження описаного практикуму в навчальний процес на фізико-математичному факультеті Харківського національного педагогічного університету дозволяє зробити наступні висновки. Курс чисельних методів набув більшої значимості у формуванні математичної культури студентів, було істотно розширено коло апробованих методів і коло розглянутих задач. Навчальні дослідження, при наявності відповідного програмного і методичного забезпечення, а також при певній наполегливості викладача виявилися цілком посиленою і результативною формою навчальної роботи студентів.

РАЗРАБОТКА КОМПЬЮТЕРНЫХ ОБУЧАЮЩИХ ПРОГРАММ ПО ОС LINUX И ЕЕ ПРИЛОЖЕНИЯМ

Л.Р. Билялова

г. Симферополь, Крымский государственный инженерно-педагогический
университет

Biljalova@ukr.net

В настоящее время активно разрабатываются компьютерные инструментальные средства для ведения учебных курсов. Практически по всем направлениям учебных дисциплин создаются электронные учебники и самоучители. Усиление интереса к подобным источникам связано с появлением мультимедийных технологий, а также с развитием средств коммуникаций, сети Интернет.

Известно, что возможности применения компьютеров в учебном процессе весьма многообразны. Они могут служить для моделирования изучаемых явлений или систем, для реализации учебных игр, применяться для выполнения вычислений, для редактирования текстов, в качестве различного рода тренажеров, а также как инструмент автоматизации проектирования, программируемого управления экспериментами, как информационно-поисковая или экспертная система, наконец, как средство практического обучения самой компьютерной технике и программированию. Вместе с тем, особый интерес представляет использование компьютера в качестве дидактического инструмента, например, на уроках информатики. Использование обучающих программ является одним из важнейших методов использования вычислительной техники в образовании.

Выбирая стратегию преподавания информатики в школе, необходимо учитывать, что задача общеобразовательного курса – это в большой степени выработка определенного стиля мышления, формирование наиболее общих навыков, умений и представлений, нежели освоение тех или иных конкретных языков и технических средств программирования.

В последние годы школьный курс «Основы информатики и вычислительной техники» вышел на качественно новый этап своего развития. Во-первых, более-менее унифицировался набор школьной вычислительной техники. Во-вторых, создано и разрабатывается ориентированное на учебные цели программное обеспечение. Сегодня созданы обширные программные средства компьютерных информационных технологий, позволяющих работать с компьютером непрограммирующему пользователю. Поэтому минимальным уровнем компьютерной грамотности является овладение средствами компьютерных информационных технологий.

Проблемы использования электронных учебных средств и ресурсов Интернета в образовательных учреждениях являются актуальными не только в Украине или на пост советском пространстве, но и в развитых странах

Европы. Внедрение новых информационных технологий в образовании в рамках программы Фонда новых перспектив в Великобритании [6, с. 77] показало с одной стороны необходимость подобных инициатив, а с другой стороны – их эффективность. Практически все слушатели курсов, организованных этой программой [6, с. 82], отметили прогресс в освоении средств электронного обучения (e-learning).

Переучивание и доучивание работающих педагогов применению ИКТ в преподавании, безусловно, необходимо. Но, как показывают различные оценки, количество обученных педагогов существенно отличается от количества тех, кто самостоятельно и целенаправленно, со знанием требований к профессиональной деятельности в условиях информатизации образовательного пространства и своих возможностей способен применять ИКТ в процессе обучения, методической деятельности, собственного непрерывного профессионального педагогического образования [5, с. 91].

В связи с интенсивным развитием информационных и коммуникационных технологий (ИКТ) и активным внедрением их в образовательный процесс, необходимо, с одной стороны, повышение ИКТ-квалификации работающих учителей. С другой стороны, необходимо активное внедрение ИКТ-технологий в процесс обучения студентов, особенно студентов педагогических вузов. ИКТ-компетентность учителя становится важной составляющей его профессионализма. Среди компонентов профессиональной ИКТ-компетентности называется умение представлять образовательную информацию с использованием различных компьютерных средств, создание информационных компьютерных обучающих программ, разработка компьютерных тестов, разработка систем рейтинговой оценки знаний учащихся, создание учебных пособий и материалов на электронных носителях с использованием инструментальных средств, создание собственных Интернет-ресурсов учебного назначения [9, с. 99].

Программами для общеобразовательных учебных заведений по дисциплине «Информатика» [7, с. 185] предусмотрено, что в зависимости от типа компьютерной техники и программного обеспечения, учитель может самостоятельно выбирать методические пути решения образовательных задач курса. И в соответствии с выбранной методикой изучения курса учитель может выбирать подходящие учебные пособия и дидактическое обеспечение. Например, учитель может опираться на операционную систему (ОС) Windows или Linux, а также на приложения соответствующей операционной системы.

На территории Украины большое распространение получили операционные системы семейства Windows фирмы Microsoft, соответственно большинство программного обеспечения разработано под этот тип операционных систем. Но данная операционная система является коммерческим продуктом фирмы Microsoft, поэтому наблюдается тенденция использовать в учебном процессе альтернативный вид ОС – Linux, являющейся свободно

распространяемой ОС.

Linux – это UNIX-подобная операционная система, написанная в рамках модели открытого кода. Модель открытого кода подразумевает свободное распространение исходного кода системы и возможность ее модификации любым программистом [1, с. 1041]. В результате Linux сегодня – это полноценная графическая операционная система с различными оболочками: Motif, CDE, KDE, GNOME, GNUStep и т.д. Они совместили в себе современную функциональность, удобство использования и отличный дизайн с технологическими преимуществами операционной системы семейства Unix.

ОС Linux на сегодняшний день очень популярна. Этот программный продукт играет на рынке информационных технологий все более значимую роль ввиду острой борьбы с пиратским программным обеспечением. Поэтому настало время задуматься о переходе на свободно распространяемые программные среды, к числу которых относится и ОС Linux. В последнее время все чаще стали говорить о Linux, как об операционной системе, способной в ближайшем будущем если не вытеснить с рынка Microsoft Windows, то полноценно заменить ее на большинстве домашних персональных компьютеров, а также в системе общего среднего и высшего образования [1, с. 1043]. Сегодня Linux – полноценная ОС семейства UNIX. Практически все важнейшие программные пакеты адаптированы и для Linux. Уже есть версии ОС Linux, поддерживающие украинские шрифты.

Вместе с тем учебно-методической литературы о Linux и ее приложениях на русском языке крайне мало, а на украинском она практически отсутствует. Большая часть имеющейся в продаже литературы посвящена самой ОС Linux и рассчитана она в первую очередь на специалистов и опытных пользователей. Наряду с этим в последнее время стали появляться учебные пособия, рассчитанные на пользователя, который имеет опыт работы в среде Windows [4, с. 3]. Названный учебник позволяет приобрести базовые знания для работы пользователя в среде Linux и с ее приложениями.

Однако, методической литературы явно недостаточно, а электронных обучающих программ практически нет. В связи с этим возникла необходимость в разработке программных средств учебного назначения по обучению работе в ОС Linux, а также с ее приложениями.

В учебных учреждениях используется ОС Windows и соответственно пакет MS Office. Однако в последнее время появилась тенденция в учебных учреждениях в качестве ОС использовать Unix-системы. Это обусловлено рядом причин (условия их распространения, надежность, современность, мобильность, дружелюбность к пользователю, перспективность и т.д.). Эта тенденция имеет широкие перспективы для своего развития, однако на данный момент нет ни достаточного опыта в преподавании данных тем, ни достаточно хорошо разработанных методик преподавания. Однако, достаточное развитие получили учебные пособия [8] и методики обучения работе с аналогичными приложениями ОС Windows.

Среди приложений ОС Linux можно назвать приложения пакета офисных программ OpenOffice.

В составе его можно выделить несколько компонентов

- Writer – развитый текстовый процессор, сопоставимый по возможностям с текущими версиями MS Word или WordPerfect; позволяет работать как с обычными текстовыми документами в собственном формате, так и с Web-страницами в формате HTML; он же выполняет функции браузера;
- Calc – электронная таблица, функционально близкая к MS Excel;
- Draw – векторный графический редактор, который сопоставим примерно с CorelDraw 5–6 версий; обладает уникальной особенностью: позволяет, хотя и в ограниченном объеме, работать с настоящей трехмерной графикой в стандарте OpenGL;

Одним из обязательных этапов университетского педагогического образования студентов специальности «Информатика» КГИПУ является применение современных информационных технологий, инструментальных средств разработки приложений, сред программирования для разработки мультимедийных учебных пособий по различным дисциплинам, в том числе и при выполнении курсового и дипломного проектирования [2, с. 89].

В рамках научно-исследовательской и учебно-методической работы студентами и преподавателями кафедры информатики КГИПУ ведется разработка электронных учебников, методических пособий, мультимедийных курсов, тестирующих программ, которые используются в учебном процессе. Для преподавания информатики в школе студентами специальности «Информатика» в рамках дисциплины «Школьный курс информатики и методика преподавания» разрабатываются электронные учебно-методические материалы под ОС Linux.

Учебники и учебные пособия нового поколения должны иметь разветвленную структуру, для которой характерна блочно-модульная модель представления знаний [3, с. 87]. Кроме этого электронный учебник должен содержать не только информационный блок, но и блок текущего или итогового контроля знаний.

Разрабатываемые электронные учебные пособия содержат несколько обязательных блоков (рис. 1). Информационный блок может быть использован преподавателями для подготовки к уроку по соответствующей теме, а учениками для дополнительных занятий или выполнения домашнего задания. Блок организационных форм обучения предназначен для подготовки преподавателя к проведению лабораторных и практических занятий, а также содержит раздаточный материал для соответствующих занятий. Блок контроля знаний содержит тесты для проведения текущего и итогового контроля по пройденной теме, а также редактор тестов, позволяющий преподавателю изменять набор вопросов. Кроме того, имеется блок, содержащий конспекты уроков по заданной теме.

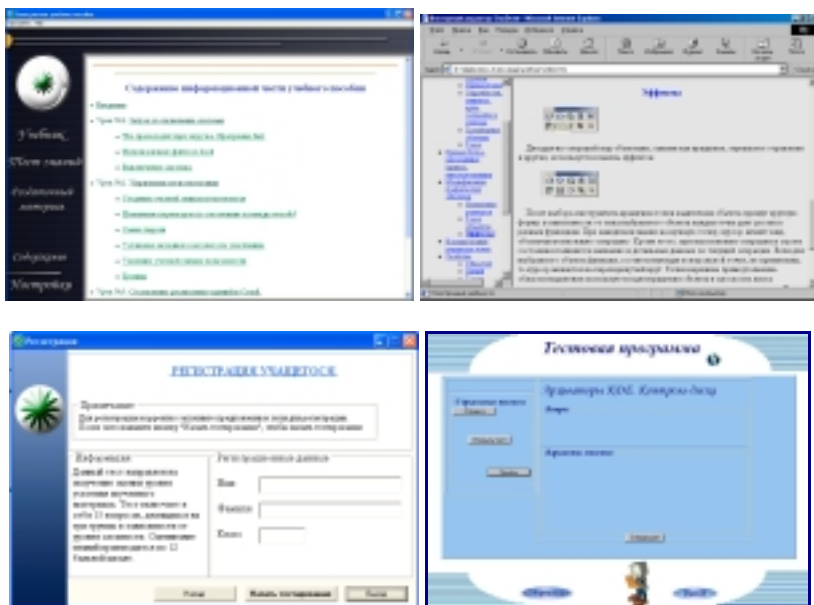


Рис. 1. Фрагменты окон обучающих программ

Часть из разрабатываемых программ проходит апробацию во время прохождения педагогической практики в общеобразовательных учреждениях Крыма. Созданные электронные учебно-методические пособия могут применяться в учебных заведениях в качестве дополнительного учебного средства для самостоятельной работы учащихся, для организации фронтальной работы в классе, для подготовки учителя к уроку. Электронные помощники призваны помочь учащемуся, способствуя более эффективному, наглядному процессу обучения, а также учителю, обеспечивая дифференцируемое изложение учебного материала и оперативный контроль знаний учащихся.

Литература:

1. Армстронг (мл.), Джеймс. Секреты Unix: 2-е изд.: Пер. с англ. – М.: Издательский дом «Вильямс», 2001. – 1072 с.
2. Билялова Л.Р., Литвинова Э.В. Использование современных информационных технологий в преподавании информатики в вузах. Ученые записки КГИПУ. Выпуск 3. – Симферополь: Доля, 2002. – С. 87-92.
3. Виштак О.В. Дидактические основы разработки педагогического сценария мультимедийного учебного пособия по информатике // Информатика и образование. – 2004. – №7. – С. 87-90.
4. Глушаков С.В., Сурядный А.С. Linux для дома и офиса: Учебный курс. – Харьков: Фолио, 2002. – 389 с.

5. Горбунова Л.Н., Семибратов А.М. Освоение информационных и коммуникационных технологий педагогами в контексте ориентации на профессионально-личностное развитие. // Информатика и образование. – 2004. – № 7. – С. 91-95.

6. Джон П. Кателл. Информационные и коммуникационные технологии для активного обучения. // Информатика и образование. – 2004. – №3. – С. 77-86.

7. Информатика. Програми для загальноосвітніх навчальних закладів. – Запоріжжя: Прем'єр, 2003. – 304 с.

8. Информатика: Навч. посібн. для 10-11 кл. середн. загально-освітн. шкіл / І.Т. Зарецька, Б.К. Колодяжний, А.М. Гуржій, О.Ю. Соколов. – Х.: Факт, 2001. – 498 с.

9. Лебедева М.Б., Шилова О.Н. Что такое ИКТ-компетентность студентов педагогического университета и как ее формировать. // Информатика и образование. – 2004. – № 3. – С. 95-100.

ЗАСТОСУВАННЯ ЕЛЕКТРОННОГО ПІДРУЧНИКА У НАВЧАЛЬНОМУ ПРОЦЕСІ

О.В. Бондар

м. Луганськ, Луганський національний педагогічний університет
ім. Т. Шевченка

В останні роки з розвитком комп'ютерної техніки і мережних технологій роль електронних видань навчального і словниково-довідкового характеру безупинно зростає. У зв'язку з актуальністю даного питання вченими, педагогами, фахівцями інформаційно-бібліотечної й іншої сфер ведуться розробки основних принципів, що характеризують сучасний електронний підручник, його підготовку, поширення, застосування і т. і. Велике значення приділяється і процесу створення електронних підручників (ЕП) [1].

Що ж таке «Електронний підручник» і в чому його відмінності від звичайного підручника? Звичайно електронний підручник являє собою комплект навчальних, контролюючих, моделюючих і інших програм, розташованих на магнітних носіях (твердому або гнучкому дисках) ПЕОМ, у яких відбито основний науковий зміст навчальної дисципліни. ЕП часто доповнює звичайний і особливо ефективний у тих випадках, коли він:

- забезпечує практично миттєвий зворотний зв'язок;
- допомагає швидко знайти необхідну інформацію (у тому числі контекстний пошук), пошук якої в звичайному підручнику утруднений;
- істотно заощаджує час при багаторазових звертаннях до гіпертекстових пояснень;
- поряд з коротким текстом – показує, розповідає, моделює і т.д. (саме тут виявляються можливості і переваги мультимедіа-технологій) дозволяє швидко, але в темпі найбільш придатному для конкретного індивідуума, перевірити знання з визначеного розділу.

До недоліків ЕП можна віднести погану фізіологічність дисплея як засобу сприйняття інформації (сприйняття з екрана текстової інформації набагато менш зручно й ефективно, чим читання книги) і більш високу вартість у порівнянні з книгою.

Педагогічний потенціал ЕП порівняно зі звичайним (паперовим) значно більший:

- *гіпертекстовість* – можливість перегляду навчального матеріалу за гіперпосиланнями;
- *мультимедійність* – можливість використання всіх засобів мультимедіа для більш ефективного подання навчального матеріалу (звук, графіка, мультиплікація, анімація, відео);
- *інтегрованість* – ЕП може включати не тільки навчальні матеріали, але й запитання, тести для самоконтролю та контролю, гіперпосилання на іншу довідкову та навчальну літературу;

– *конструктивність* – тільки інформаційні та комунікаційні технології (ІКТ) дозволяють будувати навчальний курс за принципами конструктивізму у навчанні, згідно з яким навчання реалізується через конструювання когнітивних (уявних) моделей через експерименти з реальністю або її комп'ютерними моделями;

– *інтерактивність* – можливість організувати навігацію (послідовність пред'явлення навчального матеріалу) підручника залежно від успішності, психофізіологічних або інших індивідуальних характеристик студента, тобто забезпечити ЕП засобами зворотного зв'язку. Це забезпечує керуваність процесу навчання (при цьому можливе керування як автоматичне, за допомогою самого ЕП, так і самостійне керування студентом послідовністю вивчення матеріалу).

Жодної з цих можливостей не можна реалізувати за допомогою традиційного, некомп'ютерного підручника.

Складання якісних ЕП, які у певній мірі відповідають обговореним вище принципам, є дуже трудомісткою справою. Тут можна виділити умовно чотири класи ЕП за рівнем їх опанування педагогічним потенціалом ІКТ.

1. **Базовий рівень** – електронний варіант звичайного підручника, який оздоблено системою гіперпосилань, електронним змістом та електронним індексним показником.

2. **Достатній рівень** – ЕП додатково до розвинутої системи гіперпосилань використовує мультимедійні засоби представлення матеріалу.

3. **Продвинутий рівень** – ЕП додатково до розвинутої системи гіперпосилань та мультимедійних засобів представлення навчального матеріалу має також систему комп'ютерних тестів для тематичного та підсумкового контролю успішності навчального процесу.

4. **Визначний рівень** – ЕП, додатково до якостей продвинутого рівня, інтегрований з фаховим пакетом для даної предметної області або спеціалізованим діяльнісним середовищем для комп'ютерного моделювання задач предметної галузі та дослідження їх на основі комп'ютерних експериментів.

5. **Предметно-дослідницький рівень** – ЕП, який побудований у діалоговій формі, де подання навчальному матеріалу передує обговорення та постановка задачі, особисто вагомої для студента, яка займає ключове місце в даному курсі, розв'язок якої будується в режимі діалогу "студент – ЕП", для формування гіпотез використовується комп'ютерне моделювання, виконується постійний моніторинг успішності на основі діалогів та тестів, який визначає порядок вивчення тем [2].

В.М. Гасов і А.М. Циганенко, говорячи про технології створення гіпертекстових видань, виділяють у ній наступні складові: зміст; оформлення; програмні засоби, що у взаємодії з апаратними засобами і дозволяють одержати повноцінний електронний підручник [3].

Засоби створення електронних підручників можна розділити на групи, наприклад, використовуючи комплексний критерій, що включає такі показ-

ники, як призначення і виконувани функції, вимоги до технічного забезпечення, особливості застосування. Відповідно до зазначеного критерію можлива наступна класифікація:

- традиційні алгоритмічні мови;
- інструментальні засоби загального призначення;
- засоби мультимедіа;
- гіпертекстові і гіпермедіа засоби.

При виборі засобів необхідна оцінка наявності:

- апаратних засобів визначеної конфігурації;
- сертифікованих програмних систем;
- фахівців необхідного рівня.

Крім того, необхідно враховувати призначення розроблюваного ЕП, необхідність подальшої модифікації й доповнення новими даними, обмеження на обсяг пам'яті й ін.

Завдяки технології, яка бурхливо розвивається, засоби мультимедіа і гіпермедіа стають досить дешевими, і їх можна встановлювати на більшість персональних комп'ютерів. Крім того, потужність і швидкодія апаратних засобів дозволяють використовувати вищезгадані засоби.

В даний час до підручників пред'являються наступні вимоги:

1. Інформація з обраного курсу має бути добре структурованою і представляти собою закінчені фрагменти курсу з обмеженим числом нових понять.

2. Кожен фрагмент, поряд з текстом, має представляти інформацію в аудіо- або відео вигляді ("живі лекції"). Обов'язковим елементом інтерфейсу для живих лекцій буде лінійка прокручування, що дозволяє повторити лекцію з будь-якого місця.

3. Текстова інформація може дублювати деяку частину живих лекцій.

4. На ілюстраціях, що представляють складні моделі або пристрої, повинна бути миттєва підказка, що з'являється або зникає синхронно з рухом курсору по окремих елементах ілюстрації (карти, плану, схеми, креслення зборки виробу, пульта керування об'єктом і т.д.).

5. Текстова частина повинна супроводжуватися численними перехресними посиланнями, що дозволяють скоротити час пошуку необхідної інформації, а також могутнім пошуковим центром. Перспективним елементом може бути підключення спеціалізованого тлумачного словника з даної предметної області.

6. Відеоінформація або анімації повинні супроводжувати розділи, які важко зрозуміти в звичайному викладі. У цьому випадку витрати часу для користувачів у п'ять-десять разів менше в порівнянні з традиційним підручником. Деякі явища взагалі неможливо описати людині, яка ніколи їх не бачила (водоспад, вогонь і т.д.). Відеокліпи дозволяють змінювати масштаб часу і демонструвати явища в прискореній, уповільненій або вибірковій зйомці.

7. Наявність аудіо інформації, що у багатьох випадках є основною і часом незамінною змістовною частиною підручника.

Серед режимів роботи ЕП можна виділити 3 основних:

- 1) навчання без перевірки;
- 2) навчання з перевіркою, при якому наприкінці кожної глави (параграфа) тому, якого навчають, пропонується відповісти на кілька питань, що дозволяють визначити ступінь засвоєння матеріалу;
- 3) тестовий контроль, призначений для підсумкового контролю знань з виставлянням оцінки.

Таким чином, можна сказати, що електронний посібник – одна з технологій, яка сприяє розвитку самостійної роботи і мислення студентів та підвищує рівень засвоєння ними матеріалу.

Література:

1. Вуль В.А. Электронные издания. – М. – СПб.: Изд-во “Петербургский институт печати”, 2001. – 308 с., ил.
2. Прокопенко І.Ф., Биков В.Ю., Раков С.А. До питання інформатизації вищих педагогічних навчальних закладів // Комп’ютер у школі та сім’ї. – 2002. – №4. – С. 8–13.
3. Гасов В.М., Цыганенко А.М. Методы и средства подготовки электронных изданий: Учебное пособие. – М.: МГУП, 2001. – 735 с.: ил.

ДИСЦИПЛІНА “ПРИКЛАДНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ” – ЛАНКА МІЖ ТЕОРІЄЮ ТА ПРАКТИКОЮ

О.М. Боско, Н.В. Гринь
м. Кривий Ріг, Інститут ділового адміністрування

Необхідність вивчення програмного забезпечення, яке має актуальність на даний час викликана, насамперед, тим, що вищі навчальні заклади покликані своєчасно реагувати на будь-які зміни, що відбуваються у науці та практичному попиту суспільства. Дисципліна “Прикладне програмне забезпечення” є саме тією ланкою, яка єднає науковий досвід із практичним застосуванням. Оскільки вона не має лекційного забезпечення, а лише курс лабораторних робіт, то викладач має унікальну можливість швидкої перебудови стосовно практичних потреб програмного забезпечення фахівців різних напрямів народного господарства. Крім того, ця дисципліна є необхідною для закріплення практичних навичок роботи студентів на ПЕОМ, оскільки програмне забезпечення розглядається з точки зору автоматизованого робочого місця, а саме з таким поняттям вони стикаються майже одразу, приступаючи до безпосередньої роботи за фахом.

Прикладом гнучкості дисципліни “Прикладне програмне забезпечення” є розроблена нами методика вивчення програми “БЕСТ-ЗВІТ” для студентів спеціальності “Облік та аудит”. Ця програма з’явилась на ринку програмного забезпечення в сфері бухгалтерії ще в 2000 р. За чотири роки вона стала найбільш поширеною для забезпечення АРМ підготовки звітних форм. Актуальність її в наш час визначається підтримкою таких державних установ, як податкова інспекція, де, згідно останнім вимогам, необхідно кожну звітну форму оформити як в друкованій формі, так і в електронному вигляді в базовому форматі, запропонованому “БЕСТ-ЗВІТ”. Отже, оскільки ця програма набула статусу офіційної, то необхідність її детального вивчення стала очевидною.

До цього року в рамках програми з дисципліни нами було передбачено лише три лабораторних роботи для поверхового ознайомлення з “БЕСТ-ЗВІТ”. Програма розглядалась за наступними етапами: введення на основі довідки програми та заповнення картки підприємства, ознайомлення з заповненням форми декларації з податку на прибуток підприємства та форми декларації з податку на додану вартість. В 2004 р. на останньому етапі вивчення дисципліни в рамках експерименту було значно розширено вивчення цієї програми. Не ознайомлення, а детальне вивчення програми було покладене в основу лабораторних завдань, які разом утворюють рішення комплексної задачі і послідовно наочно показує студентам якості “БЕСТ-ЗВІТ”. Розширення відбулось завдяки введення вивчення тієї частини програми, яка демонструє прозорість даних від попереднього звітнього періоду до наступного, а також вивчення квартальних звітних форм у соціальні фонди із заро-

бітної плати. Слід зазначити, що в цьому році ми роз'єднали програму третього семестру вивчення дисципліни “ППЗ” між студентами спеціальностей “Облік та аудит” і “Фінанси”.

В третьому семестрі “фінансисти” мали можливість більш детально ознайомитись з базами даних на прикладі бази даних Access. Методика викладання відрізнялась від загальноприйнятої системи тим, що особлива увага, в першу чергу, була звернена саме на прояв індивідуальної творчості студентів. Було запропоновано створити декілька баз даних з різною тематикою. Студенти розбились на творчі групи із двох чи трьох членів для створення конкретної бази, тему якої вибирали самостійно зі списку запропонованих. Кожен з членів групи займався розробкою своєї частини роботи, але наприкінці, збираючи свою групову роботу та оформлюючи звіт, вони повинні були передати один одному знання. Роль викладача на період розробки баз полягала в індивідуальних консультаціях. Контроль здійснювався на кінцевому семінарському занятті, де кожен творчий колектив захищав свою роботу та здійснювався зворотний зв'язок за допомогою питань з аудиторії.

Особливо треба відмітити результати анкетування студентів, яке відбулось наприкінці вивчення дисципліни. Анкетування проводилось у чотирьох групах студентів четвертого курсу спеціальності “Облік та аудит” і “Фінанси”. Відповіді на питанні анкети свідчать про те, що загалом студенти задоволені підбором практичних програм і вважають їх необхідними для своєї подальшої роботи. Анкетування проводилось на основі анонімності, лише треба було вказати свою групу для визначення року навчання, але масове зазначення своїх прізвищ та імен також свідчать про позитивне ставлення студентів до вивчення цієї дисципліни. Стовідсотково студенти вказали на необхідність введення такої дисципліни, як обов'язкової для вивчення. Також цікаво відмітити, що майже всі студенти висловили побажання розділення груп на підгрупи для можливості реального здійснення індивідуального підходу в навчанні.

РАЗРАБОТКА СИСТЕМЫ ПРЕПОДАВАНИЯ ОСНОВ ТЕЛЕМЕДИЦИНЫ В МЕДИЦИНСКОМ ВУЗЕ

А.В. Владимирский, Е.Т. Дорохова
г. Донецк, Донецкий государственный медицинский университет
им. М. Горького
avv@telemed.org.ua

Телемедицина, наряду с иными технологиями электронного здравоохранения, уже стала неотъемлемой частью системы мировой медицины. Активно телемедицинские системы развиваются и в Украине. Практическое использование телемедицины требует создания качественной системы обучения этим новым медицинским знаниям и умениям. Необходима специализированная эффективная додипломная подготовка врачей с использованием максимально возможного спектра информационных цифровых технологий [2, 6, 8–10]. В наших предыдущих публикациях мы уже рассматривали некоторые теоретические, методические и организационные аспекты преподавания телемедицины [1, 5–6]. В данной статье мы хотели бы представить концептуальную систему додипломного обучения студентов медицинских вузов по курсу «Телемедицина». Прежде всего, в соответствии с сегодняшними реалиями, такая система учитывает методологические принципы реформирования подготовки медицинских и фармацевтических кадров в Украине в контексте Болонского процесса, которые изложены в учебной программе (далее – Программа) с аналогичным названием, предложенной Министерством здравоохранения Украины [7]. В Программе отмечено, что Болонский процесс для медицинского и фармацевтического образования в Украине является не только стратегией выбора, но и реальной возможностью получения высшего образования на уровне европейских (мировых) стандартов для нынешнего и будущих поколений.

Разработанная нами концептуальная система додипломного обучения студентов медицинских вузов по курсу «Телемедицина» соответствует новой модели организации учебного процесса, приведенной в Программе. Вместе с тем система оригинальна благодаря тому, что в нее включены такие составляющие:

- шаблоны лекций и практических занятий по телемедицине и наборы тестовых контрольных заданий к ним;
- мультимедийные обучающие средства, автоматизированное рабочее место (АРМ) студента, электронные учебники.

Нами определены конечные цели изучения дисциплины [6], осуществлено структурирование учебного материала по содержанию и выражение содержательных модулей в зачетных кредитах, разработана и предложена шаблонная учебная программа по курсу «Телемедицина» для студентов медицинских вузов.

СОДЕРЖАНИЕ ПРОГРАММЫ

Модуль 1

Содержательный модуль 1

Основные тенденции развития телемедицины в мире

Тема 1. Определение, предмет и задачи телемедицины. История, классификация и строение телемедицинских систем. Основные телемедицинские процедуры.

Тема 2. Этико-деонтологические, юридические проблемы телемедицины.

Тема 3. Медицинская информация. Ее виды, стандарты и способы передачи. Способы защиты медицинской информации.

Тема 4. Телемедицинские ресурсы Интернет. Принципы построения телемедицинских сетей.

Содержательный модуль 2

Клинические аспекты телемедицины

Тема 5. Телеконсультирование, инструктаж, дистанционное манипулирование. Определение, задачи. Строение и функционирование телемедицинских систем.

Тема 6. Биорадиотелеметрия и мониторинг. Определение, задачи. Строение и функционирование систем. Тактико-медицинские системы. Домашняя телемедицина.

Тема 7. Частные виды телемедицины.

Содержательный модуль 3

Дистанционное обучение

Тема 8. Дистанционное обучение. Определение, задачи. Строение и функционирование систем для дистанционного обучения. Демонстрационная видеоконференция.

Модуль 2

Выполнение проекта: Подготовка медицинской информации для телемедицинских процедур и передача с помощью телекоммуникационных технологий.

Работа с электронной формой истории болезни. Подготовка данных для телеконсультирования. Подготовка и проведение демонстрационной видеоконференции. Организация асинхронного телеконсультирования. Проведение мониторинга и телеметрии. Инструктаж, дистанционное манипулирование. Дистанционное обучение.

Самостоятельная работа: Разработка Интернет-ресурса для телеконсультирования. Разработка фрагмента мультимедийной системы для дистанционного обучения. Разработка программного обеспечения для телемедицинских процедур. Научно-исследовательская работа по теме, определяемой преподавателем в рамках учебной программы.

В настоящее время проводится разработка методических пособий и совершенствование опубликованных ранее тестовых заданий [3].

По-нашему мнению, шаблон лекции по тому или иному вопросу телемедицины должен включать в себя следующие разделы:

- название;
- цели овладения учебным материалом;
- обязательный учебный материал (в рамках учебной программы);
- новейшую информацию о тенденциях развития данной проблемы и видение будущего;
- иллюстративный материал (презентации с элементами анимации, видеоролики, фотографии);
- краткий обзор основных ресурсов Интернет по данной проблеме с онлайн-демонстрацией 1-2 сайтов;
- дискуссию, ответы на вопросы.

В качестве примера приведем структуру вводной лекции курса «Телемедицина», предназначенной для студентов.

Шаблонная структура лекции курса "Телемедицина"

1. Название «Компьютерные технологии в современном здравоохранении – Телемедицина».

2. Общая цель – уметь использовать телемедицинские технологии для клинического применения основных телемедицинских процедур.

3. Обязательный учебный материал:

- определение, предмет, задачи телемедицины;
- строение и классификация телемедицинских систем;
- характеристика основных телемедицинских процедур;
- телеконсультирование (классификация, технологии, выбор методики, показания, особенности клинического использования);
- технологическая база телемедицины;
- клинические примеры.

4. Достижения телемедицины в Донецке и Донецкой области, клинические результаты, перспективы создания региональной телемедицинской сети.

5. Мультимедийная презентация, иллюстрации (таблицы, фотографии из собственной практики, фотографии из ресурсов Интернет, анимационные и видео ролики).

6. Обзор сайта «Телемедицина в Украине».

7. Дискуссия. Ответы на вопросы.

В процессе лекционного курса и проведения практических занятий необходимо использовать отдаленные и непосредственные мультимедийные обучающие средства (МОС). Наиболее рациональным считаем создание МОС на основе технологии гипертекстовых документов, снабженных графическими, видео, звуковыми и т.д. файлами. Необходимо предусмотреть логическое сопряжение и свободное соединение МОС с тематическими ресурсами Интернет.

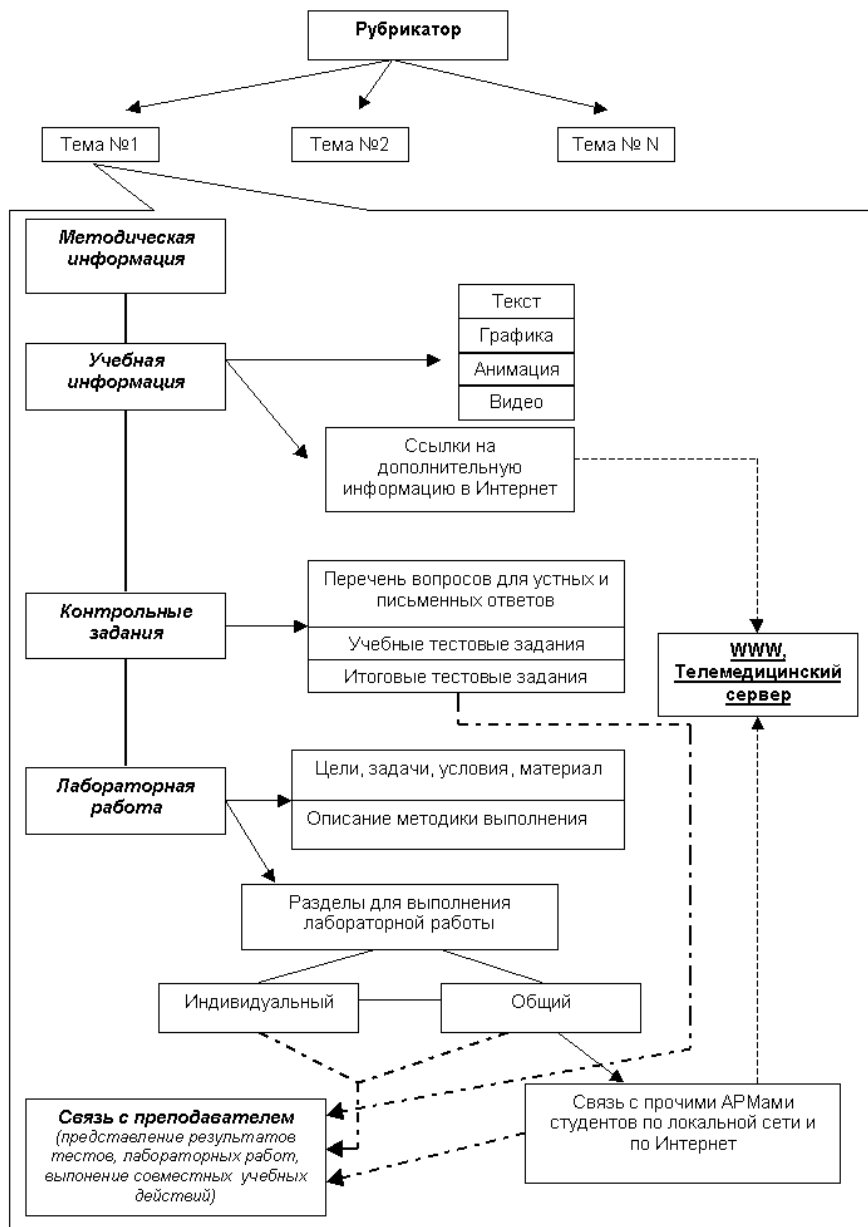


Рис. Принципиальная схема АРМа студента для изучения курса «Телемедицина»

Такие МОС позволят студенту ознакомиться с методической и учебной информацией по той или иной проблеме, самостоятельно изучить дополнительные веб-сайты, провести обучающее и итоговое тестирование, выполнить самостоятельную работу.

Полагаем, что наиболее эффективным было бы создание МОС в виде автоматизированного рабочего места (АРМ) студента по изучению курса «Телемедицина». Основные составляющие такого комплекса приведены на рисунке.

Практические занятия по телемедицине необходимо проводить в компьютерном классе, имеющем локальную сеть и подключение к Интернет. Кроме того, необходимо такое оборудование:

- цифровая фотокамера, негатоскоп, планшетный сканер, аудиомикрофон (для изучения методик подготовки медицинской информации к телемедицинским процедурам);
- веб-камера (для моделирования или проведения видеоконференции).

Методически важным элементом преподавания раздела «Телемедицина» считаем предварительную договоренность с другими вузами о совместном осуществлении некоторых фрагментов обучения: в процессе выполнения самостоятельной работы студенты из разных городов, стран должны иметь возможность обмениваться учебными случаями для телеконсультирования, проводить синхронные процедуры, моделировать сеансы дистанционного обучения и т.д.

В качестве шаблона для разработки средств дистанционного обучения основам телемедицины предполагаем использовать разработанную нами ранее систему TeleTrauma [4, 10].

Предложенная концептуальная система додипломного обучения студентов медицинских вузов по курсу «Телемедицина» частично реализована в Донецком государственном медицинском университете им. М. Горького. Продолжается ее совершенствование и практическая реализация.

Литература:

1. Владимирский А.В. Телемедицинские технологии на основе Интернет: телеконсультирование и дистанционное обучение // Украинский медицинский альманах. – 2003. – Т.7, №2. – С. 71-74.
2. Григорьев А.И., Логинов В.А., Буравков С.В., Буравкова Л.Б., Орлов О.И. Использование информационных подходов космической медицины в преподавании телемедицины. –<http://www.telemed.ru/rfrs/tmsymp/loginov.html>.
3. Казаков В.Н., Климовицкий В.Г., Владимирский А.В. Телемедицина. – Донецк: Типография ООО «Норд», 2002. – 100 с.
4. Климовицкий В.Г., Владимирский А.В., Рушай А.К., Худобин В.Ю. Дистанционное обучение в травматологии и ортопедии – собственные разработки // Травма. – 2003. – Т.4, №1. – С. 62-68.

5. Казаков В.М., Климовицький В.Г., Владзимирський А.В. Телетравматологія та телеортопедія – клінічні та освітні аспекти // Медична освіта. – 2002. – №2. – С.27–31.
6. Казаков В.М., Дорохова О.Т. Концептуальні підходи до викладання телемедицини // Український журнал телемедицини та медичної телематики – 2004. – Т.2, №1. – С. 4–7.
7. Передерій В.Г., Белан С.М., Булах І.Є., Гулько С.М. та ін. Методологічні принципи реформування підготовки медичних та фармацевтичних кадрів в Україні у контексті Болонського процесу// Навчальна програма. – Київ, 2004. – 10 с.
8. Хасаншин Ю.Р., Хасаншина Е.В., Шевелев В.М. Телемедицинские технологии в рамках медицинского университета // Український журнал телемедицини та медичної телематики – 2003. – Т.1, №1. – С. 96–105.
9. Яценко В.П., Яценко Е.В. Теоретические и прикладные аспекты телемедицины // Вестник Международного Соломонова университета. Медицинская инженерия. – 2001. – №5. – С. 26-33.
10. Magee JH. Validation of medical modeling & simulation training devices and systems. // Stud Health Technol Inform. – 2003;94:196-8.
11. TeleTrauma. Система для дистанционного обучения. – www.telemed.org.ua. Last check 15.11.2004.

МІЖПРЕДМЕТНІ ЗВ'ЯЗКИ У СТРУКТУРУВАННІ КУРСУ “ІНФОРМАТИКА І КОМП'ЮТЕРНА ТЕХНІКА”

Л.П. Воронець

м. Суми, Сумський національний аграрний університет
voronecl@mail.ru

Як свідчить аналіз нормативних документів про освіту [3, 4, 6, 12], на сьогодні одним з головних завдань її прогностичного розвитку є розробка концептуально-дидактичних і змістовно-операційних положень, нормативів, рекомендацій щодо формування змісту освіти, її навчальних курсів, предметів.

Під формуванням, зокрема курсу “Інформатика і комп'ютерна техніка”, ми будемо розуміти комплексне виконання двох взаємопов'язаних завдань: добір інформації до змісту курсу і його структурування у логічно закінчені блоки навчального матеріалу з метою реалізації принципу наступності і послідовності навчання, комплексного застосування різних форм проведення занять.

Проблема формування, зокрема структурування, навчальних курсів систематично привертає увагу науковців. При цьому психологи, на нашу думку, у більшій мірі вирішують проблему з позицій оптимізації процесу навчання, досягнення цілей, мети, виконання завдань відповідно до психологічного, фізіологічного розвитку суб'єктів діяльності з урахуванням закономірностей психічних процесів формування системи знань, умінь і навичок. Фактично психологи і медики розробляють та вказують шляхи, способи, прийоми тощо виконання, дотримання встановлених “нормативних вимог” для реалізації принципу “не нашкодь” у процесі виконання поставлених завдань, досягнення тамі і мети навчання з урахуванням можливостей оптимізувати, раціоналізувати процес пошуку правильних рішень. Дидакти вирішують проблему з позицій досягнення означених цілей, мети, розв'язання завдань навчання за умов обов'язкового виконання дидактичних принципів, вимог на різних рівнях засвоєння знань, умінь, навичок організації процесу навчання [8, 9, 10, 11 та інші]. У дидактичному аспекті можна виділити три основні напрямки її розв'язання:

- обмеження обсягів навчальної інформації, яку бажано включити у інваріантну частину програми;
- визначення форм проведення занять і їх навчального навантаження;
- оцінка якості навчання, проведення контролю знань і досягнень студентів.

Складність вирішення проблеми структуризації курсу “Інформатика та комп'ютерна техніка” окреслюється тим, що цей курс вивчається протягом усього терміну навчання у ВНЗО, тобто продовжується і при вивченні інших загальнонавчальних і спеціальних предметів навчальних планів спеціа-

льностей. Це дає підстави вважати, що одним із одним із важливих елементів, вимог до структуризації курсу повинна бути *інтегрованість* змістових і операційно-діяльнісних компонентів на рівні окремих логічно-закінчених блоків інформації курсу і його міжпредметних зв'язків з вивченням інших дисциплін.

Під між предметними зв'язками, згідно з визначенням академіка В.Н. Келбакіані [5], ми будемо розуміти таку конструкцію змісту, що належить двом та більше навчальним предметам, курсам і має наступні основні характеристики:

1. Смысловое співвідношення елементів змісту, що входять в склад 2-х і більше навчальних предметів.
2. Методичні прийоми навчання (а також форми навчального процесу), адекватні предметам, між якими встановлюється зв'язок.
3. Забезпечення спрямованості формування умінь та навичок комплексно-го використання знань при розв'язанні навчальних задач.

При цьому слід відмітити, що можливість урахування міжпредметних, внутрішньопредметних взаємозв'язків між окремими змістовими і операційно-діяльнісними характеристиками структурованих навчальних предметів, курсів, доведено вже давно, зокрема у дослідженнях відомого фізіолога І.П. Павлова. Він обґрунтовував високу ефективність і продуктивність усвідомлення, засвоєння знань, умінь, навичок з позицій необхідності формування стереотипності розумової діяльності на рівні другої сигнальної системи.

Доцільність інтегрованого структурування навчальних курсів на базі внутрішніх і міжпредметних зв'язків достатньо обґрунтовано пояснюється й іншими авторами робіт з позицій особливостей і закономірностей процесу запам'ятовування навчальної інформації у вигляді системи знань, умінь, навичок. Так, автори робіт [1, 7, 13] вважають, що прийоми смислового групування навчального матеріалу та виділення смислових опорних пунктів, смислового співвідношення того, що засвоюється, у зв'язку з чимось вже відомим, є одними з ефективних шляхів підвищення якості й міцності запам'ятовування, оперативного пошуку та відновлення. При цьому *зазначають*:

- 1) включення міжпредметних зв'язків в зміст навчального предмету є результатом відображення сучасних тенденцій науки, які виникли під впливом процесів інтеграції в науці;
- 2) цілісне, системне відображення одного об'єкту вивчення в різних навчальних предметах можливо через виявлення, планування та реалізацію міжпредметних зв'язків.

Визначимо, які інформаційні функції, призначення міжпредметних зв'язків повинні бути у процесі структурування змісту програм навчальних предметів, зокрема курсу "Інформатика і комп'ютерна техніка". Це можна зробити на основі вивчення і аналізу міжпредметних зв'язків. С.В. Баба-

джанян і В.М. Монахов [1] описують такі типи зв'язків:

- 1) понятійно-часовий зв'язок;
- 2) односторонній зв'язок між предметами;
- 3) додатковий зв'язок;
- 4) зв'язок між предметами за сутністю трактування понять та визначень.

О.Д. Шебалін [14] пропонує класифікацію міжпредметних зв'язків за ступенем їх узагальненості. Всі зв'язки автор розглядає як світоглядні, серед яких виділяє генеральні, внутрициклічні, локальні. Генеральні міжпредметні зв'язки сприяють формуванню цілісної наукової картини світу, внутрициклічні виробляють адекватне уявлення про загальні закономірності наук, мета локальних міжпредметних зв'язків – об'єднувати уявлення по загальних темах різних навчальних предметів.

Дослідники міжпредметних зв'язків не залишили без уваги і дидактичні цілі їх використання, що розкриваються через такі фактори:

1. Інтеграція знань, яка передбачає, що навчання йде через вивчення окремих предметів, кожен з яких описує тільки одну сторону дійсності, що забезпечує глибину пізнання. Однак це заважає цілісному баченню світу. Синтез знань, що полягає в показі одного й того ж факту чи явища з різних точок зору, у виявленні взаємозв'язків, існуючих між різнотипними знаннями, дозволяє сформувати більш повну та об'єктивну картину світу в цілому.
2. Утворення нового знання виходячи з того, що деякі знання теоретичного та прикладного характеру не можуть бути отримані засобами тільки одного навчального предмета, а через процедуру зв'язків інформації, що поступають від суміжних навчальних предметів.
3. Синтез кількісної та якісної сторін навчальної інформації, оснований на філософському законі переходу кількісних змін в якісні.

Узагальнюючи різні висловлювання, думки з приводу ролі внутрішньо-предметних і міжпредметних зв'язків при структуруванні курсу “Інформатика і комп'ютерна техніка”, на нашу думку, слід відзначити, що усі вони є справедливими і не викликають заперечень з позицій дидактики. Проблема визначається у тому, як на практиці зафіксувати у змісті курсу “Інформатика і комп'ютерна техніка” ці взаємозв'язки для забезпечення їх нормативного виконання. На нашу думку, першим етапом розв'язання даного завдання є побудова моделі структуризації курсу, на базі якої можна створити програми, де будуть чітко визначені *взаємозв'язки і взаємовпливи між*: метою, цілями і завданнями курсу “Інформатика і комп'ютерна техніка”; структурованими логічно-закінченими блоками навчального матеріалу курсу “Інформатика і комп'ютерна техніка”; змістом програмних навчальних предметів та курсу “Інформатика і комп'ютерна техніка”; ОКХ рівнів освіти бакалавра, спеціаліста, магістра і змістом курсу “Інформатика і комп'ютерна техніка”; динамікою розвитку пізнавальних можливостей суб'єктів навчання і змістом завдань курсу “Інформатика і комп'ютерна

техніка” та послідовністю їх вирішення у контексті інтегрованого системно-модульного його вивчення.

Література:

1. Бабаджанян С.В., Монахов В.М. Междпредметные связи естественно-математических дисциплин на факультативных занятиях // Советская педагогика. – 1980. – № 1. – С. 14-21.
2. Гершунский Б.С. Компьютеризация в сфере образования: проблемы и перспективы. – М.: Педагогика, 1987. – 264 с.
3. Державна національна програма “Освіта” /Україна ХХІ століття/. – К.: Веселка, 1994..
4. Збірник законодавчих та нормативних документів про освіту. –К.: Вип. 1, 1991. – 311 с.
5. Келбакиани В.Н. Междпредметная функция математики в подготовке будущих учителей. – Тбилиси: Изд-во Тбилис. ун-та, 1994. – 360 с.
6. Концепція інформатизації системи освіти України // Освіта. – 1992. – 11 листопада.
7. Лошкарева Н.А. О понятии и видах междпредметных связей // Сов. педагогика. – 1972. – № 6. – С. 48-56.
8. Максимова В.Н. Структура и принципы отбора содержания профильных образовательных программ // Профилирование школы: разработка учебных планов / Материалы междуна. семинара. – СПб.: Образование, 1996. – С. 83-108.
9. Матрусов И.С., Леднев В.С. Некоторые пути совершенствования общего среднего образования. – В кн.: Перспективы развития содержания общего среднего образования: (сборник научных трудов). – М.: НИИ сод. и мет. обуч., 1974. – Вып. I. – С. 3-28.
10. Мирошниченко А.А. Теория и технология конструирования профессионально ориентированных структур учебных элементов. Автореф. дис. ... д-ра пед. наук. – Глазов, 1999. – 32 с.
11. Сохор А.М. Логическая структура учебного материала. – М.: Педагогика, 1974. – 192 с.
12. Стратегія реформування освіти в Україні: Рекомендації з освітньої політики. – К.: К.І.С., 2003. – 296 с.
13. Федорова В.Н., Кирюшкин Д.М. Междпредметные связи на материале естественнонаучных дисциплин средней школы. – М.: Педагогика, 1973. – 152 с.
14. Шебалин О.Д. Подготовка учителей к реализации мировоззренческой функции школьного предмета. – М.: Изд-во Моск. гос. пед. ин-та, 1985. – 91 с.

ПІДГОТОВКА РОЗРОБНИКА ІНФОРМАЦІЙНИХ СИСТЕМ В НТУУ "КПІ"

О.О. Гагарін, В.І. Гайдаржи
м. Київ, Національний технічний університет України "КПІ"

Постановка проблеми. На теперішній час потреба в спеціалістах, що мають професійні знання та уміння зі створення та використання розподілених комп'ютерних систем інформаційно-аналітичного призначення, є актуальною у зв'язку з тим, що з активним впровадженням комп'ютерних технологій в процедури аналізу закономірностей функціонування та керування технологічними процесами в різних предметних областях постійно зростає потреба в системах такого класу.

Для досягнення цієї мети необхідно забезпечити послідовне вивчення майбутніми спеціалістами відповідного кола навчальних дисциплін. Саме питанням про структуру цього кола та його зміст присвячено дану статтю.

Аналіз останніх досліджень. На поточний час основним документом, що визначає структуру та зміст навчального процесу з підготовки бакалаврів в галузі розробки інформаційних систем, є галузеві стандарти (ОКХ та ОПП) з безперервної підготовки бакалаврів з напрямку "Комп'ютерні науки", розроблені вченими кафедри АУТС НТУУ "КПІ", та концепція безперервної освіти в галузі інформаційних систем, видана Міністерством освіти та науки.

Крім того, в НТУУ "КПІ" на кафедрі АПЕПС розроблені відповідні стандарти та концепція екологічної освіти в технічних університетах України, в яких розглядаються питання підготовки розробників інформаційних систем в галузі програмного забезпечення автоматизованих систем та комп'ютерного еколого-економічного моніторингу.

Наведені в даній статті матеріали є поглибленням та практичної реалізацією концептуальних підходів до розгляду конкретного змісту навчальних дисциплін з підготовки розробників складних розподілених інформаційних систем.

Метою статті є розгляд структури частини навчального плану та змісту дисциплін присвячених послідовній підготовці розробників інформаційних систем на прикладі підготовки спеціалістів за спеціальностями 7.080403 – "Програмне забезпечення автоматизованих системи" та 7.080407 – "Комп'ютерний еколого-економічний моніторинг".

Основна частина

Підготовка спеціалістів за спеціальністю "Програмне забезпечення автоматизованих системи" та "Комп'ютерний еколого-економічний моніторинг" здійснюється на бази підготовки бакалаврів з напрямку "Комп'ютерні науки" і передбачає за мету формування як на ступені підготовки бакалавра, так і на ступені підготовки спеціаліста та магістра знань умінь та навичок з

проектування складних інформаційних систем. Для досягнення цієї мети в межах навчального плану побудована наступна базова послідовність викладання дисциплін.

1. Організація баз даних
2. Основи автоматизованого проектування складних інформаційних об'єктів і систем
3. Розподілені системи обробки інформації
4. Проектування систем з розподіленими базами даних

Організація баз даних

Матеріал курсу є базовим при вивченні студентами дисциплін навчального плану, пов'язаних із створенням інформаційних систем. Отримані знання дозволять студентам використовувати методи інформаційного моделювання при вивченні інших інженерних дисциплін, виконанні курсових і дипломних робіт.

У структурно-логічній схемі навчання зазначена дисципліна розміщена у 5, 6 семестрах.

Загальна кількість навчального часу – 243 години. В тому числі: лекцій – 35, практичних – 18, лабораторних – 70 та СРС – 120.

Основною метою викладання дисципліни є надання майбутнім фахівцям основ науково – теоретичних знань та практичних навичок з модельних, мовних та програмно-технологічних засобів проектування, розробки та використання баз даних на прикладі СКБД FoxPro. В системі підготовки фахівця дисципліна займає особливе місце, оскільки вона формує найважливіші практичні вміння з проектування баз даних для інформаційних систем функціонування яких спирається на використання баз даних

Зміст навчального матеріалу дисципліни:

Розділ 1. Основні визначення

Тема 1.1 Інформаційні системи і системи обробки даних.

Розділ 2. Первинний аналіз предметної області

Тема 2.1 Логічне моделювання. Ієрархічна та мережна модель даних. Структури даних. Операції над даними. Обмеження цілісності.

Тема 2.2 Проектування концептуальної моделі предметної області

Тема 2.3 Концептуальне моделювання бази даних. Алгебра Кодда. Реляційна модель даних. Нормалізація відношень.

Розділ 3. СКБД Visual FoxPro

Тема 3.1 Поняття БД у FoxPro.

Тема 3.2 Технологічні можливості систем обробки даних. Створення та використання елементів бази даних (таблиць, індексів та представлень).

Тема 3.3 Навігація. Фільтрація. Сортування. Методи пошуку даних.

Тема 3.4 Сумісна обробка декількох таблиць. Зв'язки виду “один до одного” та “один до кількох”.

Розділ 4. Visual FoxPro. Мова програмування XBase

Тема 4.1 Створення додатків. Створення форм для введення та редагування даних в Visual FoxPro. Проектування меню.

Тема 4.2 Проектування звітів.

Тема 4.3 Налаштування середовища Visual FoxPro. Внутрішня та зовнішня конфігурації. Файл config.fr.

Тема 4.4 Управління проектом та створення додатку Project.

Розділ 5. Мова SQL

Тема 5.1 Обробка таблиць.

Тема 5.2 Робота з індексами.

Тема 5.3 Робота з віртуальними таблицями та курсорами.

Тема 5.4 Вбудований SQL. Поняття процедури, тригерів, транзакцій.

Програма курсу, крім лекцій, включає виконання циклу лабораторних робіт і курсові роботи. Метою лабораторних робіт є самостійна розробка баз даних і створення програм, які реалізують різні засоби створення, поновлення і використання баз даних. Метою курсових робіт є повна реалізація інформаційної системи з використанням процедурних та технологічних засобів СКБД.

В результаті вивчення дисципліни студент повинен:

- *набути теоретичних знань з реляційної алгебри Кодда та реляційної моделі даних, засвоїти поняття відношення та операцій над відношеннями;*
- *отримати практичні навички з проектування нормалізованих баз даних;*
- *вміти формувати по заданій функціональності в предметній області концептуальну схему відповідної бази даних;*
- *набути практичні навички зі створення реляційної БД та встановлення відношень між таблицями;*
- *вміти створювати програмні проекти для обробки інформації що міститься в базах даних з використанням спеціалізованих програмних засобів, таких як конструктори екранних форм, звітів, меню, запитів. Такі засоби, звичайно, входять у системи керування базами даних (СКБД).*

Основи автоматизованого проектування складних інформаційних об'єктів і систем

Дисципліна є складовою у підготовці спеціалістів напряму “Комп’ютерні науки”, тому що розробка прикладного програмного забезпечення для автоматизації адміністративної та виробничої діяльності являє собою систему, яка забезпечує обробку складної інформаційної моделі систем управління або виробництва. Інформаційна модель в більшості випадків є базою даних досить складної структури, яка містить опис зміст багатьох інформаційних об’єктів. Розробка такої складної системи проводиться із залученням систем автоматизованого проектування програмного забезпечення, що надають можливість виконувати необхідні операції щодо формування моделей інформаційної системи на стадії проектування системи та

виконання різноманітних операцій над базами даних для підтримки функціональності системи на стадії її експлуатації. Програми, розроблені з використанням САПР, значно підвищують якість розробки та зменшують час, необхідний для неї.

Зазначена дисципліна включена до циклу “Професійно-орієнтованих дисциплін за переліком програми”. У структурно-логічній схемі навчання зазначена дисципліна розміщена у 7 семестрі.

Загальна кількість навчального часу – 108 годин. В тому числі: лекцій – 18, лабораторних – 36 та СРС – 54.

Метою викладання дисципліни є:

1. Вивчення та практичне засвоєння засобів моделювання складних інформаційних систем засобами мови UML.

2. Формування умінь здійснювати розробку інформаційних систем на основі використання SQL-орієнтованих серверів баз даних засобами середовища візуального проектування Delphi.

Детально розглядаються питання моделювання та проектування інформаційних системи за допомогою графічної уніфікованої мови моделювання UML.

Велика увага приділяється також розгляду питання використання баз даних DBase та Paradox для створення інформаційних систем. Розглядаються властивості SQL-серверу InterBase щодо підтримки складних баз даних, особливо питання розробки збережених процедур та тригерів, а також механізми управління транзакціями.

Зміст навчального матеріалу дисципліни:

Розділ 1. Технологія проектування складних інформаційних систем

Тема 1.1 Процес проектування інформаційної системи.

Тема 1.2 Моделювання складної інформаційної системи засобами мови UML. Діаграми UML. Case-засоби UML-моделювання.

Розділ 2. Машина баз даних Borland Database Engine (BDE)

Тема 2.1. Основи використання машини баз даних в середовищі Delphi.

Розділ 3. Проектування Delphi-додатків для роботи з базами даних

Тема 3.1 Загальні засоби обробки баз даних засобами серверів баз даних.

Тема 3.2 Використання сервера баз даних InterBase.

Тема 3.3 Основні елементи даних в бази даних. Склад метаданих. Тригери та збережені процедури.

Тема 3.4 Транзакції. Управління транзакціями. Кешовані зміни.

В результаті вивчення дисципліни студент повинен:

1. Набути теоретичних знань з
 - основних принципів розробки прикладного програмного забезпечення, яке працює з базами даних в середовищі WINDOWS-орієнтованих операційних систем;
 - принципів побудови та використання інструментальних засобів мо-

делювання складних інформаційних систем;

- принципів та технологічних процедур звернення до баз даних, які є керованими різноманітними системами керування базами даних;
 - можливостей CASE-систем автоматизованого проектування програмного забезпечення (зокрема, системи діалогового проектування Delphi) відносно обробки інформації, що зберігається в базах даних;
 - особливостей обробки баз даних, які є керованими стандартними СКБД (наприклад, DBase та Paradox), та SQL-сервером баз даних InterBase.
2. Набути практичні навички з:
- моделювання складних інформаційних систем засобами мови UML;
 - виконання типових операцій щодо створення різних типів UML-діаграм, якв використовуються на етапах моделювання складних інформаційних систем;
 - розробки інформаційних систем із зверненням до баз даних, які є керованими різноманітними системами керування базами даних, за допомогою системи діалогового проектування Delphi;
 - детальної розробки інформаційні системи з обробки баз даних, які є керованими стандартними системами керування базами даних (до яких традиційно відносять системи керування базами даних DBase та Paradox), та SQL-сервером баз даних InterBase.

Розподілені системи обробки інформації

Дисципліна “Розподілені системи обробки інформації” є одною із складових у підготовці спеціалістів напряму “Комп’ютерні науки”. Це обумовлено тим, що розробка прикладного програмного забезпечення для створення корпоративних систем автоматизації адміністративної та виробничої діяльності великих підприємств потребує від спеціалістів знань сучасних технологій створення розподілених інформаційних систем. Така розробка не була б можливою без використання спеціальних засобів проектування розподілених систем, які значно підвищують якість розробки та зменшують час необхідний для неї.

Зазначена дисципліна включена до циклу “Професійно орієнтованих дисциплін за переліком програми”. У структурно-логічній схемі навчання зазначена дисципліна розміщена у 8 семестрі. Загальна кількість навчального часу – 108 годин. В тому числі: лекцій – 32, лабораторних – 32 та СРС – 48.

Метою курсу є вивчення та практичне засвоєння сучасних технологій розробки розподілених інформаційних систем.

В курсі вивчаються технології розробки та використання систем “Клієнт–сервер” на базі використання технологій COM/DCOM та CORBA. Розглядаються питання з створення систем “Клієнт–сервер”, ґрунтуючись на різних варіантах застосування елементів технологій COM/DCOM та CORBA, вивчаються механізми синхронного погодження даних та тиражу-

вання даних у розподілених системах обробки інформації. Також в курсі розглядаються технології розробки інформаційних систем з використанням моделей розподілених компонентів. Вивчаються засоби розробки і використання компонентних об'єктів COM/DCOM, JavaBeans і CORBA-технологій.

Зміст навчального матеріалу дисципліни:

Розділ 1. Загальні відомості про розподілені системи обробки інформації

Тема 1.1 DLL – базове рішення з організації функціонального забезпечення.

Розділ 2. Модель COM/DCOM

Тема 2.1 Базова модель COM-технології.

Тема 2.2 Модель автоматизації в COM-технології.

Тема 2.3 Модель автоматизації зі зворотним викликом в COM-технології.

Розділ 3. Модель CORBA

ТЕМА 3.1 Структура клієнт–серверної системи в моделі CORBA.

В результаті вивчення дисципліни студент повинен ЗНАТИ:

- загальні теоретичні відомості щодо принципів розробки корпоративних систем обробки інформації та їх застосування на основі використання базових серверів COM-технології;

- загальні теоретичні відомості щодо принципів розробки корпоративних систем обробки інформації та їх використання при створенні інформаційних систем на основі систем клієнт–сервер на базі серверів автоматизації COM-технології;

- загальні теоретичні та практичні знання щодо створення систем клієнт–сервер на основі COM-технології з використаннями інтерфейсів зворотного виклику;

- загальні теоретичні відомості щодо принципів побудови розподілених систем обробки інформації та їх використання на основі використання систем клієнт–сервер на базі CORBA-технології.

В результаті вивчення дисципліни студент повинен УМІТИ:

- створювати розподілені системи обробки інформації на основі використання базових систем клієнт–сервер на основі COM-технології;

- створювати розподілені системи обробки інформації на основі використання систем клієнт–сервер на базі серверів автоматизації COM-технології;

- розробляти системи клієнт–сервер на базі серверів автоматизації COM-технології з використанням інтерфейсів зворотного виклику;

- розробляти розподілені системи обробки інформації на основі використання систем клієнт–сервер на базі CORBA-технології.

Проектування систем з розподіленими базами даних

Дисципліна “Проектування систем з розподіленими базами даних” –

складова у підготовці спеціалістів напряму “Комп’ютерні науки”, тому що розробка прикладного програмного забезпечення для автоматизації адміністративної та виробничої діяльності являє собою систему, яка забезпечує обробку інформаційної моделі систем управління або виробництва. Інформаційна модель в більшості випадків є базою даних досить складної структури, яка містить опис зміст багатьох інформаційних об’єктів, що містяться в різних місцях зберігання, тобто створюють розподілену в просторі базу даних. Така розробка проводиться з залученням систем автоматизованого проектування програмного забезпечення, що надають можливість виконувати необхідні операції над розподіленими базами даних, які є важливою частиною програмного забезпечення ПЕОМ. Програми, розроблені з використанням спеціальних засобів проектування розподілених систем та баз, значно підвищують якість розробки та зменшують час, необхідний для неї.

Дисципліна розміщена у 10 семестрі. Загальна кількість навчального часу – 162 годин. В тому числі: лекцій – 34, лабораторних – 34 та СРС – 96.

Дисципліна читається на п’ятому курсі спеціальності. Метою курсу є вивчення та практичне засвоєння засобів розробки систем з розподіленими базами даних, які забезпечують функціонування складних інформаційних систем, з використанням SQL-серверів баз даних, засобами середовища візуального проектування Delphi.

Розглядаються механізми використання можливостей проектування інформаційних систем з використанням засобів серверів баз даних, які підтримують платформу розробки клієнт-серверних систем.

Розглядаються властивості SQL-серверу InterBase щодо підтримки роботи з розподіленими базами даних, особливо питання розробки дволанкової та триланкової архітектури побудови та використання систем з розподіленими базами даних, а також відповідні механізми управління транзакціями.

Зміст навчального матеріалу дисципліни:

Розділ 1. Проектування структури інформаційних систем з розподіленими базами даних

Тема 1.1 Загальне уявлення та класифікація розподілених інформаційних систем.

Розділ 2. Розробка та використання розподілених баз даних

Тема 2.1 Визначення, властивості та класифікація розподілених баз даних.

Тема 2.2 Технології побудови систем з розподіленими базами даних.

Тема 2.3 Технології інтеграції інформаційних систем.

Розділ 3. Інструментальні засоби розробки систем з розподіленими

БД

Тема 3.1 Використання дволанкової архітектури клієнт-сервер бази даних

Тема 3.2 MIDAS (DATASAP) – технологія створення клієнт-сервер-

них додатків триланкової архітектури.

Тема 3.3 ADO – технологія створення триланкової архітектури клієнт-сервер.

Тема 3.4 Використання CORBA-технологій для створення систем клієнт-сервер.

В результаті вивчення дисципліни студент повинен ЗНАТИ:

- теоретичні основи моделювання інформаційних систем, які працюють з розподіленими базами даних в середовищі Windows-орієнтованих операційних систем;
- теоретичні основи можливостей системи автоматизованого проектування програмного забезпечення Delphi відносно обробки інформації, яка зберігається в розподілених базах даних;
- принципи та технологічні процедури звернення до баз даних, які є керованими різноманітними системами керування базами даних як в дволанковій, так і в триланковій архітектурі;
- технологічні принципи, покладені в основу функціонування інформаційних систем на основі використання можливостей, які надає MIDAS (DATASAP), ADO та CORBA-технології для розробки систем з розподіленими базами даних.

В результаті вивчення дисципліни студент повинен УМІТИ:

- виконувати моделювання інформаційних систем, які працюють з розподіленими в просторі базами даних, засобами спеціалізованих мов моделювання, зокрема мови UML.
- створювати засобами візуального проектування Delphi інформаційні системи з обробки інформації, яка зберігається в розподілених базах даних;
- використовувати технологічні процедури звернення до баз даних, які є керованими різноманітними системами керування базами даних як в дволанковій, так і в триланковій архітектурі
- застосовувати можливості MIDAS (DATASAP), ADO та CORBA-технологій для розробки систем з розподіленими базами даних.

Висновки

Для досягнення мети підготовки спеціалістів з проектування розподілених інформаційних систем доцільно використати наступну базову послідовність викладання дисциплін присвячених згаданім питанням:

1. Організація баз даних
2. Основи автоматизованого проектування складних інформаційних об'єктів і систем
3. Розподілені системи обробки інформації
4. Проектування систем з розподіленими базами даних

За наведеною схемою в НТУУ “КПІ” на протязі кількох років успішно здійснюється підготовка бакалаврів, спеціалістів та магістрів з спеціальностей 7.080403 – “Програмне забезпечення автоматизованих системи” та 7.080407 – “Комп’ютерний еколого-економічний моніторинг”.

ПОХОДЖЕННЯ ТЕРМІНУ “ІНФОРМАЦІЯ” ТА РОЗУМІННЯ ЙОГО СУТНОСТІ В РАМКАХ ВУЗІВСЬКОГО КУРСУ ІНФОРМАТИКИ

О.В. Гладченко

м. Ірпінь, Національна академія державної податкової служби України
gov@ufei.ukrsat.com

Інформатика як наука проходить етап становлення. Проте деякі теми, поняття ми вже розглядаємо як фундаментальні. До таких понять можна віднести і поняття інформація.

Питання методики ознайомлення з поняттям інформації учнів середніх навчальних закладів було висвітлено М.І. Жалдаком та Н.В. Морзе в статтях [3], [4].

Недостатньо розробленим залишається дане питання з навчально-методичного боку у вищих навчальних закладах.

Одним з ефективних методів викладу навчального матеріалу у вузі є комбінована проблемна лекція, яка поряд з лекцією, бесідою, поясненням та у поєднанні з іншими методами навчання дозволяє озброювати студентів більш глибокими і повними знаннями, забезпечувати високий науковий рівень і творчий характер підготовки студентів. При розкритті навчального матеріалу з теми “Поняття інформації” у вузі доцільно використовувати як щойно перераховані у вищій школі методи, так і вище запропоновані авторами статей [3], [4].

Доцільно ґрунтовно розглянути дане поняття у вузі, забезпечуючи принципи науковості, доступності, наступності (фрагментарно, частково) в знаннях студентів. В даній темі є можливість відображення відповідного профілю вищого навчального закладу. Потрібно сказати, що дана тема є надзвичайно сприятливою для забезпечення міжпредметних зв'язків інформатики з філософією, мовою. На заняттях з філософії студентів можна познайомити з основними існуючими філософськими течіями, теоріями інформації; а на заняттях з мови – з не ідентичністю понять дані, відомості, повідомлення і поняття інформація.

Вже на початку заняття можна познайомити студентів з походженням слова “інформація”.

Ще з школи відомо, що термін “інформація” походить від латинського *informatio*, що означає роз'яснення, виклад, обізнаність.

В давні часи слово “інформація” використовувалося виключно у значенні “ідея, наука”.

Слово “інформація” спочатку з'явилося і використовувалося не в розмовній мові, а в словниках, науковій літературі, документах.

Так, в словнику Ф. Толля друком від 1863 року, інформація розуміється як «дане, дещо відоме, незаперечне в науці, факт, який служить основою для якого-небудь висновку, розрахунку або заключення» [9].

Вважається, що слово інформація прийшло до нас із польської мови і не використовувалося в повсякденному мовленні до початку ХХ століття [1].

Першими почали вживати і досліджувати поняття інформації журналісти в 20-х роках минулого століття.

Отже на теренах нашої країни дане поняття спочатку з'явилося в газетах, журналах, пізніше в словниках.

У вітчизняних тлумачних словниках поняття “інформація” не зустрічається аж до початку ХХ століття [1]. Вперше про інформацію йдеться в словнику, який вийшов друком у 1935 році і визначається як дія по дієслову “інформувати”, або як повідомлення, інформування про стан справ чи діяльність.

Подивимося, як класики української літератури обходилися без слова інформація та похідних від нього слів:

- Звістіть мене, якою дорогою послали Ви книжку. Відповідно до вашого повідомлення я постараюсь ужити всі заходи, щоб дістати книжку (Коцюбинський).

- Зробіть у книгарні наказ, щоб вони вислали ... всі належні мені примірники, рівномірно повідомляючи мене, що й коли вислане (Коцюбинський).

- Дуже нас сполошила відомість про ваш приїзд (Франко).

- Мої відомості про літературний рух у нас дуже невеликі (Коцюбинський).

Академік М.М. Моїсєєв вважав, що “не потрібно вводити термін “інформація”, якщо в цьому немає необхідності, якщо без цього можна обійтись” [7].

Початком бурхливого розвитку даного поняття, мабуть, слід вважати включення його в корінь слова “інформатика”. Л.М. Калініна [6] вважає, що справжня історія теорії інформації починається з 1948 року, коли незалежно один від одного відомі зарубіжні вчені Н. Вінер, К. Шеннон та Р. Фішер запропонували статистично визначити кількість інформації, не даючи власне визначення поняттю “інформація”. Тобто, поняття інформації є значимим, важливим, науковим поняттям лише півстоліття, що для наукового поняття не є великим терміном.

Проте, за цей період з'явилася численна кількість течій, теорій, напрямків, які розглядають сутність даного терміну, написано багато книжок, статей, брошур, існує велика кількість тлумачень поняття “інформація”. Так, Л.М. Калініна [6] нарахувала 86 означень поняття “інформація”, серед них виділено основних видозмінених 22. Проте М.М. Моїсєєв вважав, що “строге і достатньо універсальне визначення інформації навряд чи можливе” [7]. Отже, ці факти говорять про невизначеність, неоднозначність щодо самого поняття. А.П. Суханов зазначає: “сама наука про інформацію ще досить молода, щоб у ній все можна було “розложити по полицкам” і кінце-

во розставити акценти” [10]. С.В. Симонович пише: “фундаментальної науки, яка займається дослідженням природи інформації, до цього часу не було. Лише в останні роки інформатика почала формуватися як природничо-наукова дисципліна, але вона ще не вийшла за рамки прикладної технічної науки” [5].

Для кращого розуміння теми потрібно навчити студентів чітко розмежувати поняття “інформації” з поняттями “дані”, “відомості”, “повідомлення”.

Аналізуючи поняття “повідомлення”, “відомості”, “дані”, “факти” за допомогою тлумачного словника, в результаті аналізу і обговорення сутності цих понять, використовуючи активні методи навчання, і звернувши увагу на неоднозначність, круговорот і заплутаність в формулюванні цих понять ми приходимо до висновку, що дані терміни можна ототожнювати, сприймати як слова синоніми в залежності від сфери застосування, від змістовної сторони речення. Так, наприклад, слово дані, як правило, коректно вживати по відношенню до чисел, слово повідомлення – до тексту, символічних повідомлень, а слово відомості може вживатися як в першому, так і в другому значенні. Слово факти коректно вживати в юриспруденції.

Проте терміни “повідомлення”, “відомості”, “дані”, “факти” не можна ототожнювати з поняттям “інформація”. Хоча б тому, що інформація не є об’єктом матеріальної природи, а повідомлення, дані, відомості мають матеріальну основу, тобто це щось конкретне, вони мають реальне проявлення, а ми – люди, живі істоти, маємо можливість їх сприймати, тобто бачити візуально, чути на слух, відчувати на смак чи нюх.

Наступним етапом може бути наведення прикладів, в яких із стилістичної точки зору коректно вживати одне з слів дані, відомості, повідомлення і недопустимо вживати інше слово.

- По каналам зв’язку передаються текстові повідомлення і числові дані.
- З радіо і телебачення ми чуємо повідомлення.
- Книги і газети містять відомості, бази даних – дані.
- Ми отримуємо повідомлення зведень про погоду.
- Опитуючи свідків, ми отримуємо факти або повідомлення. В даному випадку з стилістичної точки зору некоректно вживати слово дані або відомості. А вже з отриманих фактів (з аналізу показів (із слів) свідків), в результаті взаємодії таких методів, як логічне мислення, уява та прогнозування, слідчий отримує інформацію про об’єкт, який його цікавить. Слідчий повинен врахувати, що свідчення або можна сказати повідомлення свідків не завжди несуть адекватну інформацію, оскільки в результаті спостереження і логічного мислення свідків одні і ті ж об’єктивні дані або факти, тобто те, що свідок бачив, можуть інтерпретуватись ними по-різному, висловлюватися не зовсім точними, однозначними словами в їх показах і перетворюватися в різну інформацію.

Таким чином, ми підійшли до розгляду сутності поняття інформації. Повторюючи шкільний матеріал з даної теми [2, 8], викладач інформатики повинен поглибити уявлення, знання про поняття інформації. Викладачу можуть стати в нагоді висловлювання видатних людей, які досліджували сутність даного поняття. Цим самим ми забезпечимо принцип науковості при поданні матеріалу.

Епіграфом до всього заняття можуть стати чудові рядки академіка В.М. Глушкова: “Інформацію несуть не тільки помережані літерами аркуші книг чи людська мова, а й пасма гірських хребтів, шум водограю, шепотіння листя і так далі” (російський варіант: “Информацию несут в себе не только испещренные буквами листы книг или человеческая речь, но и солнечный свет, складки горного хребта, шум водопада, шелест листвы и т.д.”).

Філософи щодо сутності поняття інформація дискутують вже не один десяток років, та прийти до спільної думки так і не можуть. Правда всі вони погоджуються з тим, що інформація є філософською категорією і її потрібно розглядати як одну із сторін відображення.

Але яку саме? Це питання є найбільш дискусійним серед філософів. Поняття відображення – більш широке, більш змістовне, ніж поняття інформація.

А ось як говорив про інформацію академік А.П. Суханов [10]:

- Поняття інформація в самому загальному вигляді є абстракцією. Інформація взагалі, інформація як така існує лише в людській свідомості і лише для того, щоб виділити істотне, основне, головне, те, що ... ми спостерігаємо в житті;

- ... немає в природі інформації взагалі, інформації “в чистому вигляді”;

- Інформація безперервно поступає, незалежно від того, чи здатні ми сприймати її і використовувати для поповнення своїх знань, чи ні.

Щоб говорити про інформацію, необхідно одночасно мати у взаємодії [1]:

- ♦ об’єкт, який несе повідомлення (джерело відображення); таким об’єктом можуть бути як живі, так і неживі матеріальні об’єкти (всі види матерії);

- ♦ об’єкт, який сприймає повідомлення (приймач); ним може бути жива істота, яка має свідомість;

- ♦ канал, по якому передається повідомлення.

Процес отримання інформації триває рівно стільки, скільки існує взаємодія між джерелом відображення і сприймаючою істотою, а в інший час маємо дані, повідомлення [5].

Інформація функціонує лише на ... стадії контактній (ще є доконтактна і післяконтактна), коли між предметами встановлюється “місток” взаєморозуміння, коли з’являється необхідність і можливість (зі сторони фізичних, хімічних процесів, процесів мислення і т.д.) для отримання інформації [10].

Інформацію отримують з навколишнього середовища за допомогою органів чуттів і шляхом умозаключень як результат роботи мозку.

Інформація є основним матеріалом мислення і лежить в основі всякої розумової діяльності.

Викладач повинен повторити (або ввести) поняття шуму, розглядаючи достатню кількість прикладів; звернути увагу студентів на властивостях інформації; наголосити на правильності вживання фрази “одиниці вимірювання даних або повідомлень” і ні в якому разі не “одиниці вимірювання інформації”.

У вищій школі на дану тему варто виділити принаймні одне лекційне і одне практичне заняття, провівши останнє у вигляді семінару.

На понятті інформація базуються такі поняття як економічна, фінансова, правова інформація, які розглядаються на старших курсах при вивченні спеціальних дисциплін та курсу “Інформаційні системи”.

Література:

1. Джинчарадзе Н.Г. Інформаційна культура особи: формування та тенденції розвитку (соціально-філософський аналіз): Дис... д-ра філос. наук: 09.00.03. – К.: Київський ун-т ім. Т. Шевченка, 1997. – 452 с.
2. Жалдак М.І., Морзе Н.В. Інформатика–7: Експериментальний навчальний посібник для учнів 7 класу загальноосвітньої школи. – Київ: ДіаСофт, 2000. – 208 с.
3. Жалдак М.І., Морзе Н.В. Методика ознайомлення учнів з поняттям інформації // Комп’ютер у школі та сім’ї. – 2000. – № 4. – С. 11–16.
4. Жалдак М.І., Морзе Н.В. Методика ознайомлення учнів з поняттям інформації // Комп’ютер у школі та сім’ї. – 2001. – № 1. – С. 14–18.
5. Информатика для юристов и экономистов / Симонович С.В. и др.– СПб.: Питер, 2001. – 688 с.
6. Калініна Л.М. Інформація: суть і специфіка // Комп’ютер у школі та сім’ї. – 2002. – № 3. – С. 13–17.
7. Моисеев Н.Н. Алгоритмы развития. – М.: Наука, 1987. – 302 с.
8. Морзе Н. В. Методика навчання інформатики: Навч. посіб.: У 4 ч. / За ред. акад. М. І. Жалдака. – К.: Навчальна книга, 2003. – Ч. II: Методика навчання інформаційних технологій. – 287 с.
9. Настольный словарь для справок по всем отраслям знаний (справочный энциклопедический лексиконь): В 3 т. / Под ред. Ф. Толля. – Санкт-Петербург, 1863. – Т.1. – 800 с.
10. Суханов А.П. Информация и прогресс. –Новосибирск: Наука, 1988. – 192 с.

ПІДБІР І ПОСЛІДОВНІСТЬ ВИВЧЕННЯ ТЕМ У КУРСІ ІНФОРМАТИКИ

Я.М. Глинський, В.А. Ряжська
м. Львів, Національний університет «Львівська політехніка»
hlynsky@polynet.lviv.ua

Проблема підбору тем до курсів інформатики у загальноосвітній школі, а також базового курсу інформатики у вищій школі, і вибору правильної послідовності їх вивчення завжди є актуальною, оскільки її розв'язування дає відповідь на запитання, що і як вивчати на тому чи іншому етапі становлення цієї відносно нової навчальної дисципліни. У цьому році курсу шкільної інформатики виповнюється двадцять років. До значних здобутків за цей час можна віднести створення великої кількості навчальних програм, зокрема, для профільного навчання, а до недоліків – майже повну відсутність протягом усього часу навчальних посібників і підручників, які б відповідали цим програмам. У статті розглянуто можливі шляхи подолання деяких проблем методики навчання інформатики, а також деякі аспекти неперервності курсів інформатики в загальноосвітній і вищій школах, які впливають на підбір і послідовність вивчення тем, оскільки відповідні базові курси інформатики (для студентів некомп'ютерних напрямків підготовки) тісно переплітаються між собою за цілями навчання, побудовою та змістом основних розділів.

1. Розглянемо головні теми, які вивчаються в загальноосвітніх навчальних закладах відповідно до типових навчальних програм [1]:

1. Інформація та інформаційні процеси.
2. Інформаційна система.
3. Операційні системи.
4. Основи роботи з дисками.
5. Прикладне програмне забезпечення навчального призначення.
6. Прикладне програмне забезпечення загального призначення.
7. Глобальна мережа Інтернет.
8. Основи алгоритмізації і програмування.

Всього є $8! = 40320$ різних послідовностей вивчення цих тем. Постає питання: яка з послідовностей є найкращою?

Твердження 1. Немає послідовності, яку можна було би вважати абсолютно недопустимою.

Доведення твердження випливає з того, що можна обґрунтувати допустимість розпочати вивчення курсу з будь-якої із наведених вище теми, а також факту, що немає двох чи більше тем, які необхідно вивчати у строгій послідовності.

Звідси випливає, що задача оптимального тематичного планування немає однозначного розв'язку. Тому варто говорити про деяку абстрактну

множину **A** рівносильних у певному сенсі нормальних розв'язків і про те, що лише емпіричний досвід може відбракувати завідомо неоптимальні схеми навчання. Автори програм і навчальних посібників та підручників з інформатики повинні мати широкий простір для вибору й обґрунтування тої чи іншої схеми навчання і її змістовного наповнення.

Твердження 2. Послідовність вивчення тем, передбачувана типовими навчальними програмами, є неоптимальною.

Зауважимо, що критеріїв оптимальності і побудови множини **A** у цій короткій статті дати неможливо. Доведення твердження випливає з факту, що не може належати до множини **A** схема, в якій тема «Основи алгоритмізації і програмування» є останньою. По-перше тому, що олімпіади з інформатики переважно базуються на задачах з цієї теми. По-друге, ця тема важко сприймається учнями саме в останньому семестрі навчання у школі після вивчення попередніх більш легких, майже розважальних тем.

Розглянемо послідовність висвітлення тем у підручнику «Інформатика» [2, 3], який на сьогоднішній день визнаний Міністерством освіти і науки України як базовий:

Частина 1.

1. Інформація і комп'ютер.
2. Апаратна частина комп'ютера.
3. Огляд програмного забезпечення.
4. Операційна система MS-DOS.
5. Операційна система Windows.
6. Текстовий редактор.
7. Графічний редактор.
8. Інтернет.

Частина 2

9. Електронні таблиці.
10. Бази даних.
11. Алгоритмізація і програмування.

Розташування розділу «Алгоритмізація і програмування» в кінці підручника і неможливість переставляти теми також роблять схеми навчання за цим підручником неоптимальними.

Тепер розглянемо наступну послідовність вивчення тем курсу:

1. Інформація і комп'ютер (інформація та інформаційні процеси, інформаційна система, робота з дисками).
2. Алгоритмізація і програмування.
3. Візуальне програмування.
4. Комп'ютери і суспільство.
5. Математичні основи інформатики.
6. Операційні системи.
7. Прикладне програмне забезпечення:
 - 7.1. Графічний і музичні редактори (оглядово).

- 7.2. Текстовий редактор.
- 7.3. Видавнича система.
- 7.4. Електронні таблиці.
- 7.5. Бази даних.
- 7.6. Презентації
8. Глобальна мережа Інтернет.

Така послідовність реалізована в навчальному посібнику «Інформатика» [4, 5]. Тут охоплено усі теми діючої навчальної програми, а також деякі теми, вивчення яких передбачається згідно із стандартом з інформатики. Зупинимось коротко на декількох особливостях схеми навчання. Схема враховує, зокрема, фактор, що все більше і більше учнів мають вдома комп'ютери або мають доступ до комп'ютерів. Незабаром вже у 8-9 класах більшість учнів будуть сформованими користувачами персонального комп'ютера, а в деяких специфічних питаннях будуть орієнтуватися краще за вчителів. Тому гасло 90-х років «Готуємо користувачів ПК» поступово стає неактуальним, а навчальні програми цього не враховують. Крім цього, важливо подати інформатику як науку з властивими їй твердженнями, строгими правилами, висновками та рекомендаціями. Розділи інформатики, де в основі навчання лежать навички клацання мишею, для цього підходять мало. Тому чим раніше буде організовано вивчення теми «Алгоритмізація і програмування», тим краще.

Наведену схему можна критикувати, зокрема, за розташування теми «Глобальна мережа Інтернет» останньою. Але схема передбачає можливість переставляти теми місцями. Так, теми 4 і 5 можна переставити в кінець курсу чи винести на самостійне вивчення з метою подати тему про Інтернет скоріше. Важливою особливістю є базування навчання на концепції конструктивної індивідуальної чи колективної діяльності учнів над реалізацією проєктів. Останній проєкт курсу у цій схемі – створення особистого сайту, сайту класу чи школи – дасть змогу зберегти зацікавленість учнів предметом до останніх уроків курсу.

Твердження 3. Запропонована схема навчання є нормальною в сенсі належності до множини **A**.

Це твердження головно базується на емпіричному досвіді, авторській апробації схеми у двох Львівських гімназіях (українській приватній та академічній) і аналізу апробації попередніх видань посібника у багатьох школах регіону і м. Києва. Ще раз підкреслюємо, що строге формулювання критеріїв належності до множини **A** виходить за межі даної статті і може бути предметом наукового дослідження, оскільки висновки, які впливатимуть з нього, матимуть принципово важливе значення як для побудови різновидів курсу, так і для становлення методики викладання інформатики, яка й надалі знаходиться у зародковому стані.

2. Кожний розділ курсу складається з тем, послідовність вивчення яких також має важливе значення. Розглянемо послідовність вивчення тем у роз-

ділі «Алгоритмізація і програмування».

Сьогодні вже не потрібно агітувати за доцільність надавати більше уваги цьому розділу у курсі шкільної інформатики. На нашу думку він має бути центральним. Причини його незадовільного сприйняття у 90-х роках полягали у відсутності підручників, недосконалості методики навчання, відсутності навиків у вчителів і поганій мотивації навчання, оскільки тоді все блідо перед так званим «користувацьким підходом». Нажаль, слабкою ланкою й надалі залишається розробка правильної методики навчання саме для цього розділу курсу. Методичних розробок у цьому напрямку недостатньо. Звідси і погані навчальні програми і ще гірші підручники, оскільки за останні роки програми дещо поліпшилися, а підручники навпаки – погіршилися.

Розглянемо головні теми розділу «Алгоритмізація і програмування» і послідовність їх вивчення відповідно до типових навчальних програм [1]:

1. Інформаційна модель.
2. Алгоритми.
3. Програма. Мова програмування
4. Звернення до алгоритмів і функцій.
5. Вказівки повторення й розгалуження.
6. Табличні величини.
7. Рядкові величини.
8. Графічні операції.

Всього є $8! = 40320$ різних послідовностей вивчення цих тем. Наведена вище послідовність є нормальною в сенсі належності до деякої абстрактної множини **V** нормальних схем навчання. Зауважимо, що межі такої множини ми не беремося визначити у цій статті. Можна навести багато інших нормальних схем навчання, а також обґрунтувати значно кращі схеми, але це питання є непринциповим. Можна теоретично обґрунтувати можливість розпочати вивчення розділу з будь-якої теми, що дає право авторам програм і підручників пропонувати різні схеми навчання. Але це також несуттєво. Більш гостро стоїть проблема з підбором власне тем до розділу і з їх якісним змістовним наповненням.

Твердження 4. Відсутність теми «Файли даних» є суттєвим недоліком розглянутої вище схеми.

Ця тема має важливе пізнавальне значення і є простою для засвоєння. Через неї легко перейти до вивчення теми «Бази даних» і можна достатньо повно її висвітлити засобами мови програмування. А це перспективно в плані можливого переходу у майбутньому до використання ліцензійно чистого програмного забезпечення, оскільки у вільнопоширюваному офісному пакеті OpenOffice під Linux немає окремої програми для роботи з базами даних на зразок програми Access, доступної на машинах з операційною системою Windows. По-друге, на олімпіадах усіх рівнів робота з даними здійснюється із застосуванням файлів як структури даних. А оскільки ці поняття

виходять за межі навчальної програми, то й постановка відповідних задач на олімпіадах є некоректною по відношенню до учнів.

Твердження 5. Розкриття тем розділу з використання понять навчальної алгоритмічної мови є недоцільним у курсі сучасної інформатики.

Такий підхід продуктивно використовувався у 80-90-х роках минулого століття за умов неналежного забезпечення шкіл комп'ютерною технікою, підручниками тощо. Сьогодні все змінилось. Для освоєння програмування достатньо вивчити десяток англійських слів і ніяких проблем не буде, якщо методика навчання побудована правильно за принципом «навчання з метою розв'язування цікавих задач, а не заради вивчення мовних конструкцій», якщо є достатня мотивація навчання і кожний учень має нормальну книжку.

У [4] пропонується такий набір тем і послідовність їх вивчення, яку вважаємо нормальною:

1. Алгоритми.
2. Інформаційна модель.
3. Програма. Мова програмування.
4. Лінійні програми.
5. Логічні вирази.
6. Циклічні програми.
7. Програми з розгалуженнями.
8. Процедури і функції.
9. Графіка.
10. Рядкові величини.
11. Табличні величини.
12. Файли даних.
13. Візуальне програмування.

Коротко розглянемо деякі особливості схеми. Вивчення розділу за даною схемою можна реалізувати за будь-яку розумну кількість годин. Навчальна алгоритмічна мова лише згадується на рівні першої теми. Тема «Процедури і функції», яку вважається модно подавати чим скоріше, тут розглядається лише після вивчення головних алгоритмічних конструкцій, що дає змогу формулювати більш цікаві задачі. Циклічні програми випереджають програми з розгалуженнями – тепер можна без зволікань розв'язувати задачі зі значною кількістю обчислень, де застосовуються циклічні операції, а це важливо для мотивації навчання. Відпадає гостра потреба у вивченні таких команд як: IF-THEN-ELSE, CASE, FOR, REPEAT. Згодом можна розглянути команду циклу FOR, оскільки вона суттєво використовується в темі «Табличні величини».

Твердження 6. Тему «Табличні величини» можна вилучити з курсу чи зробити оглядовою, чи винести на самостійне вивчення без будь-якої суттєвої шкоди для базового курсу.

Цю тему варто замінити темою «Файли даних» – весь розділ стане чіткішим, доступним і завершеним. Більше того, можна обґрунтувати можли-

вість і доцільність розпочати вивчення розділу з теми «Файли даних».

Твердження 7. Тему «Візуальне програмування» доцільно вивчати після процедурного.

Вивчення цієї теми передбачено в стандарті з інформатики, вона вже є в деяких навчальних програмах. У авторів є достатній досвід викладання цієї теми для студентів та учнів і тому ми вважаємо, що візуальне програмування доцільно вивчати тільки після освоєння алгоритмічних конструкцій і розв'язування типових навчальних задач обчислювального характеру і задач пошуку інформації. Якщо ж алгоритмічні конструкції вивчати в рамках візуального програмування (а саме це пропонується в деяких програмах), то може статися як у прислів'ї «За лісом дерев не видно». Головна причина розмежування полягає в тому, що ці два підходи у програмуванні призначені для розв'язування принципово різних задач, тому суміщати їх хоч і можна (і це де-не-де обгрунтовується в зарубіжній методичній літературі), але недоцільно і неефективно.

3. Розглянемо деякі аспекти неперервності курсів інформатики у загальноосвітній і вищій школі. Оскільки чітке розмежування курсів у загальноосвітній і вищій школі на теперішньому етапі неможливе, то навчання відбувається за принципом спіралі: у базовому курсі інформатики у вищій школі розглядаються майже ті самі теми, але на вищому теоретичному і практичному рівнях. Перехід на профільне навчання суттєво ускладнив процеси розмежування. Тепер викладач вузу не зможе розпочати курс інформатики, виходячи зі знань випускників, оскільки цілком можливо, що навіть на комп'ютерному факультеті будуть студенти, які вчилися у школі в гуманітарному класі. Крім цього, школа поки що не гарантує твердого засвоєння знань з інформатики і отримання відповідних практичних навиків простою більшістю випускників в рамках задекларованих навчальних програм. Слід зауважити, що й методичної літератури, розрахованої на розмежування, дуже мало. Це питання є актуальним і чекає на своїх дослідників.

Одна з можливих концепцій спіралеподібного навчання реалізована в комплекті навчальної літератури [4–11], де є посібники для учнів [4, 5] і для студентів [6–11]. Посібники [6–11] мають гриф «Рекомендовано МОНУ» і призначені для студентів самих різних (некомп'ютерних) напрямків підготовки. Вони можуть бути корисними для різних форм навчання, а також для самоосвіти. З електронними версіями двох навчальних посібників [7, 8] можна ознайомитися на сайті www.hlynsky.lviv.ua. Три навчальні посібники [7, 8, 9] перекладені російською мовою та вийшли друком у видавництві «Діа-софт» у Санкт-Петербурзі.

Сподіваємось, що запропоновані підходи будуть корисними для розвитку методики неперервного вивчення інформатики в школі та вузі і стануть у пригоді багатьом вчителям та викладачам.

Висновок. У цій статті ми розглянули фрагменти комплексу навчальних засобів з інформатики для загальноосвітніх шкіл і вищих навчальних закла-

дів [4–11]. Ці методичні засоби пройшли масову апробацію і низку удосконалень під час декількох перевидань, що здійснювалися коштами видавництва «Деол». Ми показали шлях, яким на нашу думку мав би здійснюватися процес створення підручника з інформатики, зокрема для шкіл, оскільки шлях без достатньої апробації та науково-методичної експертизи, яким пройшли деякі підручники, наприклад [2, 3], є неправильним, некоректним по відношенню до колег, вчителів та учнів і веде до неефективного витрачання державних коштів. На протязі останніх десяти років Міністерство освіти і науки України тричі видавало масовими тиражами офіційний підручник з інформатики і тричі це було зроблено поспіхом, недбало і невдало, книжки «мертво» лягали на полиці бібліотек, а учні і надалі вивчають предмет без підручників або ж шукають альтернативну літературу. У зв'язку з цим ми звертаємося до колег-авторів і працівників Міністерства освіти і науки підтримувати правила чесного, коректного і відкритого висування і просування навчальної літератури на шляху до отримання відповідних грифів і державних замовлень на видання. На нашу думку, Міністерству освіти і науки України варто відмовитися від формування держзамовлень на навчальну літературу з інформатики і повністю довірити зацікавленим видавництвам здійснення усіх процесів створення і забезпечення навчального процесу конкурентноздатною літературою з цієї дисципліни.

Література:

1. Інформатика. Програми для загальноосвітніх навчальних закладів. – Запоріжжя: Прем'єр, 2003. – 304 с.
2. Інформатика: Підручник для 10–11 кл. загальноосвіт. навч. закладів / І.Т. Зарецька, А.М. Гуржій, О.Ю. Соколов. У 2-х ч. Ч. 1. – К.: Форум, 2004. – 392 с.
3. Інформатика: Підручник для 10–11 кл. загальноосвіт. навч. закладів / І.Т. Зарецька, А.М. Гуржій, О.Ю. Соколов. У 2-х ч. Ч. 2. – К.: Форум, 2004. – 288 с.
4. Глинський Я.М. Інформатика: Навч. посіб. для 8–11 кл. загальноосвіт. навч. закладів. У 2-х част. Ч. 1: Алгоритмізація і програмування. 4-те вид. – Львів: Деол, 2004. – 256 с.
5. Глинський Я.М. Інформатика: Навч. посібн. для 8–11 кл. загальноосвіт. навч. закладів. У 2-х част. Ч. 2: Інформаційні технології. 4-те вид. – Львів: Деол, 2004. – 264 с.
6. Глинський Я.М. Практикум з інформатики: Навч. посібн., 7-ме видання. – Львів, Деол, 2004. – 224 с.
7. Глинський Я.М., Анохін В.Є., Рязьська В.А. Паскаль. Turbo Pascal і Delphi: Навч. посібн., 5-те доп. вид. – Львів: Деол, 2004. – 176 с.
8. Глинський Я.М., Анохін В.Є., Рязьська В.А. Бейсик. Qbasic, Visual Basic і VBA: Навч. посібн., 4-те доп. вид. – Львів: Деол, 2004. – 160 с.
9. Глинський Я.М., Рязьська В.А. Інтернет. Сервіси, HTML і web-

дизайн. 2-ге доп. вид. – Львів: Деол, 2003. – 192 с.

10. Глинський Я.М., Анохін В.Є., Рязська В.А. С++ і С++ Builder: Навч. посібн., 2-ге випр. вид. – Львів: Деол, 2004. – 192 с.

11. Глинський Я.М., Рязська В.А. Ліпих-практикум з інформатики: Навч. посібн. – Львів: Деол, 2004. – 248 с.

РОЛЬ МЕТОДІВ БАГАТОМІРНОГО СТАТИСТИЧНОГО АНАЛІЗУ У ФАХОВІЙ ПІДГОТОВЦІ ЕКОНОМІСТА І МЕТОДИКА ЇХ ПРАКТИЧНОГО ОПАНУВАННЯ

П.М. Григорук
м. Хмельницький, Хмельницький національний університет
pa@ief.tup.km.ua

Удосконалення освітньої діяльності базується на застосуванні інноваційних педагогічних технологій, метою яких є підвищення якісного рівня підготовки фахівців. Комп'ютерно-орієнтовані технології навчання відкривають перед педагогами нові можливості пошуку форм навчально-виховної роботи, вироблення у студентів навичок розв'язання практичних завдань.

В умовах інтенсивного збільшення обсягу інформації, відновлення бази знань особливого значення набуває підготовка фахівців, що мають високий загальнонауковий і професійний рівень. Сьогодні важко уявити сучасного фахівця-економіста без базових знань в галузі математики, інформатики, економіки й інших дисциплін. Ефективним способом досягнення й зміцнення цих знань є економіко-математичне моделювання та застосування для цього обчислювальної техніки. Уміння розв'язувати задачі на ЕОМ є однією з головних практичних цілей як курсу інформатики, так і дисциплін економіко-математичного напрямку і дає можливість студентам розширити свої знання в області фундаментальних наук.

Сучасні комп'ютери надають широкі можливості для моделювання різних економічних явищ і процесів. На даний час одним з найбільш вживаних способів розв'язування прикладних задач з економіки і управління є використання розрахунків у електронних таблицях. Як відомо, опрацювання даних за допомогою електронних таблиць включає в себе:

- зручні можливості по наочному відображенню як результатів розрахунків, так і вихідних даних;
- графічну інтерпретацію даних;
- великий набір функцій як загального призначення, так і спеціалізованих;
- можливість використання різноманітних надбудов, які розширюють обчислювальні можливості;
- можливість поповнення функціонального набору власними процедурами та функціями.

Останнім часом попри проблему ліцензування програмного забезпечення найбільш поширеним у навчальній та практичній діяльності є табличний процесор Microsoft Excel. Окрім проведення обчислень з використанням загальноживаних функцій значна увага повинна приділятися опануванню функцій, які дозволяють провести однакові розрахунки для великих діапазонів даних. Такі завдання доцільно розв'язувати із застосуванням матри-

чних операцій.

Серед можливостей поглибленого використання цього додатку у фаховій підготовці студентів-економістів найбільш вживаним є використання надбудов “Поиск решения” та “Пакет анализа”. Вони дозволяють проводити аналіз планів великої розмірності; ефективно розв’язувати екстремальні завдання з обмеженнями складного типу, проводити аналіз множини розв’язків у вигляді сценаріїв, розв’язувати завдання кореляційно-регресійного аналізу тощо.

Особливості підготовки студентів економічного профілю в галузі економіко-математичного моделювання прикладі викладання дисциплін, пов’язаних з використанням методів багатомірного статистичного аналізу (БСА) [1]. Останнім часом серед фахівців, що займаються економіко-математичним моделюванням, ці методи стають все більш популярними. Така тенденція викликана в першу чергу стрімким розвитком засобів обчислювальної техніки.

Тривалий час практичне застосування методів БСА не було активно здійснюване, що пояснюється низкою причин:

- недостатньою висвітленістю теоретичного підґрунтя методів в науковій та навчальній літературі;
- необхідністю глибокої математичної підготовки для засвоєння методів;
- складністю практичної реалізації методів через велику кількість обчислень, що без застосування комп’ютера робило їх проведення неможливим;
- складністю інтерпретації одержаних результатів.

Методи БСА сьогодні часто називають інтелектуальним інструментарієм дослідника. Постійно зростаючий інтерес до них пояснюється в першу чергу широкими можливостями у відображенні та моделюванні реальних явищ та процесів, які мають багатознакову природу. Всі новітні напрямки досліджень, пов’язані з моделюванням катастроф, розпізнавання образів, сценарного прогнозування і т.д. передбачаючи багатомірне подання об’єктів спостереження. Звичайні статистичні одномірні методи дозволяють провести розрахунок статистичних показників (середніх величин, дисперсії, показників варіації) для окремо взятих ознак об’єктів. У тому випадку, коли потрібно провести дослідження для декількох ознак одразу, використовується комбінаційне групування. Тоді спочатку здійснюється розбиття об’єктів на групи за однією ознакою, а далі в межах кожної групи – за іншою. Недоліком такого підходу є те, що у випадку трьох і більше ознак одержується громіздка комбінаційна таблиця, яка може містити мало заповнені або й взагалі порожні групи. Крім того, процедура одержання такої таблиці є також досить громіздкою. Зазначені методи БСА дозволяють ефективно проводити багатомірну класифікацію, виділяючи їх природні скупчення об’єктів незалежно від кількості об’єктів за всіма ознаками одночасно.

До найбільш поширених класів методів БСА відносять:

- багатомірне шкалювання;
- кластерний аналіз;
- дискримінантний аналіз;
- канонічний аналіз;
- факторний аналіз.

Успішне застосування зазначених методів для розв'язання практичних завдань на ПК можливе шляхом використання стандартних пакетів статистичних розрахунків, одним з яких є STATISTICA [2, 3]. Він містить модулі для всіх наведених класів методів.

Однак використання лише цього пакету в навчальному процесі не є достатнім і ефективним для опанування студентами методів БСА. Успішність досягнення поставлених цілей можлива лише тоді, коли студенти досконально володіють інструментарієм методів, здатні виявити критичні ситуації при їх застосуванні, критично оцінити одержані результати. Крім того, разом із здобуттям практичних навичок використання методів БСА студенти повинні одержати навички детального аналізу досліджуваних явищ та процесів, змістовної інтерпретації одержаних результатів. Це можливе за умови здійснення автоматизованого застосування зазначених методів. Саме тому в такому випадку якнайкраще підходить табличний процесор. Часто застосування зазначених методів є частиною розв'язання більш складного завдання, що також більш доцільно здійснювати за допомогою використання електронної таблиці.

Аналіз перелічених класів методів БСА показує, що вони основані на достатньо складних обчисленнях, які не можливо повністю автоматизувати вбудованими засобами табличного процесора, наведеними раніше. Одним з видів таких обчислень, який притаманний практично всім методам, є знаходження власних значень та власних векторів дійсної симетричної матриці.

Для ефективного застосування табличного процесора при розв'язанні наведеного завдання пропонується використання відповідної функції користувача, оформленої у вигляді макросу.

З одного боку, такий підхід дозволить скоротити кількість виконуваних рутинних операцій, не пов'язаних напряму з сутністю розв'язуваного завдання і зосередити основну увагу на опануванні змістовної частини методу, вивченні його сутності.

З іншого боку, використання макросу є виправданим, оскільки методи знаходження власних значень і власного вектора студентами вивчались в курсах вищої математики і чисельних методів і не складають труднощів з точки зору їх розуміння.

Завдяки такому підходу з'являється можливість скорочення часу розв'язання окремого завдання, і, як наслідок, збільшення та урізноманітнення кількості завдань, які будуть розглянуті протягом заняття, збільшення часу на якісну та змістовну інтерпретацію одержаних результатів.

Слід зазначити, що макроси можуть бути підготовлені як викладачем, так і самими студентами в ході вивчення відповідних дисциплін, передбачених навчальним планом і пов'язаних з програмуванням.

Особливо слід відмітити ефективність наведеного підходу розв'язання практичних завдань при опануванні зазначених методів БСА в ході дистанційного та заочного навчання. Студенти зазначених форм навчання вимагають особливого ставлення, оскільки умови їх навчальної діяльності істотно відрізняються від умов навчання студентів денної форми. Це і скорочення обсягу аудиторної роботи, обмежений доступ до навчальної літератури, неможливість повноцінного консультативного спілкування з викладачем. Саме тому успішне виконання студентами контрольної роботи та можливе тоді, коли реалізація складних обчислювальних процедур буде значно спрощена, а відповідні макроси, що реалізують часткові алгоритми обчислень, надані студентам разом у вигляді електронних таблиць разом із завданням контрольної роботи. Наведений прийом дозволяє студентам дистанційної та заочної форм навчання значно скоротити час підготовки та виконання контрольної роботи, краще опанувати навчальний матеріал, більш раціонально організувати свою самостійну роботу.

Зазначений підхід до розв'язання практичних завдань можливий і при використанні інших методів: БИВЕС-оцінки, методу складаного ножа тощо, які використовуються в дисциплінах економіко-математичного циклу.

Література:

1. Дубров А.М., Мхитарян В.С., Трошин Л.И. Многомерные статистические методы: Учебник. – М.: Финансы и статистика, 2000. – 352 с.: ил.
2. Янковой А.Г. Многомерный статистический анализ в системе STATISTICA. – Одесса: Оптимум, 2001. Вып. 1. – 216 с.
3. Янковой А.Г. Многомерный статистический анализ в системе STATISTICA. – Одесса: Оптимум, 2001. Вып.2. – 325 с.

ЧАСТОТНО-АМПЛІТУДНИЙ АНАЛІЗ МАЙЖЕ ПЕРІОДИЧНИХ ПРОЦЕСІВ ТА ЙОГО ЗАСТОСУВАННЯ ДЛЯ ПРОГНОЗІВ

О.М. Дреєв^{1α}, З.Ю. Філер^{2β}

¹ м. Кіровоград, Кіровоградський національний технічний університет

² м. Кіровоград, Кіровоградський державний педагогічний університет імені Володимира Винниченка

^α drej_sanya@mail.ru

^β filer@kw.ukrtel.net

Дослідження детермінованих сигналів в більшості випадків пов'язано з пошуком простих гармонічних складових, тим більше це важливо для стохастичних стаціонарних процесів. Використання рядів Фур'є доцільне лише при аналізі періодичних процесів з чіткою основною частотою ω та кратних їй частот; при цьому ω вважається відомим. Тут виникають задачі пошуку ω та амплітудно-фазових складових кратних частот. При проведенні аналізу майже періодичних сигналів задача пошуку спектру сигналу значно ускладнюється, коли сигнал має невідомий частотний спектр $\omega_0, \omega_1, \dots, \omega_k, \dots$, де частоти ω_k не обов'язково є кратними. Такий сигнал не є періодичним. Знаходження значень складових частот у прийнятому діапазоні та відповідних амплітудно-фазових характеристик є важливою проблемою.

Постановка задачі. В результаті проведених вимірювань (чи іншим чином) отримано табличні значення функції $y_i=f(x_i)$, де x_i – відповідне значення аргументу функції. Необхідно визначити циклічну частоту ω таким чином, щоб середнє квадратичне відхилення

$$I(\omega) = \frac{1}{N} \sum_{i=1}^N (a_0(\omega) + a_1(\omega) \sin(\omega \cdot x_i) + a_2(\omega) \cos(\omega \cdot x_i) - y_i)^2 = \min \quad (1)$$

приймало мінімальне значення, де a_0, a_1, a_2 – деякі функції частоти, що дають мінімальну суму при визначеному ω . Головною проблемою тут є пошук ω .

1. Визначення a_0, a_1, a_2 при відомому ω . Для знаходження мінімуму $I(\omega)$ візьмемо частинні похідні:

$$\begin{cases} I'_{a_0}(\omega) = 2(a_0 + a_1 \sin(\omega \cdot x) + a_2 \cos(\omega \cdot x) - y) = 0 \\ I'_{a_1}(\omega) = 2(a_0 \sin(\omega \cdot x) + a_1 \sin^2(\omega \cdot x) + a_2 \cos(\omega \cdot x) \sin(\omega \cdot x) - y \sin(\omega \cdot x)) = 0 \\ I'_{a_2}(\omega) = 2(a_0 \cos(\omega \cdot x) + a_1 \sin(\omega \cdot x) \cos(\omega \cdot x) + a_2 \cos^2(\omega \cdot x) - y \cos(\omega \cdot x)) = 0 \end{cases} \quad (2)$$

Тут для скорочення запису ризикою позначено середнє значення виразу для всіх значень аргументу. З системи (2), маючи конкретне ω , можна отримати значення a_0, a_1, a_2 . За для спрощення запису та унаочнення запишемо систему (2) у матричному вигляді:

$$\begin{bmatrix} 1 & \overline{\sin(\omega \cdot x)} & \overline{\cos(\omega \cdot x)} \\ \overline{\sin(\omega \cdot x)} & \overline{\sin^2(\omega \cdot x)} & \overline{\sin(\omega \cdot x) \cos(\omega \cdot x)} \\ \overline{\cos(\omega \cdot x)} & \overline{\sin(\omega \cdot x) \cos(\omega \cdot x)} & \overline{\cos^2(\omega \cdot x)} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \overline{y} \\ \overline{y \sin(\omega \cdot x)} \\ \overline{y \cos(\omega \cdot x)} \end{bmatrix}. \quad (3)$$

Фактично ця система задає функції $a_0(\omega)$, $a_1(\omega)$, $a_2(\omega)$. Тоді функція

$$g(x) = a_0(\omega) + a_1(\omega) \sin(\omega \cdot x) + a_2(\omega) \cos(\omega \cdot x) \quad (4)$$

дає найкраще наближення за критерієм мінімізації її середньоквадратичного відхилення від функції $f(x)$ при заданому ω .

2. Знаходження мінімуму $I(\omega)$. На жаль, $I(\omega)$ має невизначену в загальному випадку аналітичну форму, тому відшукування мінімуму проводиться будь-яким чисельним методом. Обмеження шуканої частоти ω при рівномірному розбитті часу досліджуваних даних дає теорема Найквіста-

Котельнікова [1]: $\omega \leq \frac{\pi}{2h}$, де $h = x_{i+1} - x_i = \text{const}$. Тому шукаємо ω в діапазоні $(0; \pi/2h]$, тобто з періодом $T \geq 4h$. Використовуючи умову рівномірного розбиття, можна значно спростити знаходження середніх арифметичних значень при побудові системи (3):

$$\begin{aligned} \overline{\cos(ax)} &= \frac{\cos(ax_0 + \omega h / 2) - \cos(ax_0 + n \omega h + \omega h / 2)}{2n \sin(\omega h / 2)}, \\ \overline{\cos(ax)} &= - \frac{\sin(ax_0 + \omega h / 2) - \sin(ax_0 + n \omega h + \omega h / 2)}{2n \sin(\omega h / 2)}, \\ \overline{\sin(ax) \cos(ax)} &= \frac{\cos(2ax_0 + \omega h) - \cos(2ax_0 + 2n \omega h + \omega h)}{4n \sin(\omega h)}, \\ \overline{\sin^2(ax)} &= \frac{1}{2} - \frac{\sin(2ax_0 + \omega h) - \sin(2ax_0 + 2n \omega h + \omega h)}{4n \sin(\omega h)}, \\ \overline{\cos^2(ax)} &= \frac{1}{2} + \frac{\sin(2ax_0 + \omega h) - \sin(2ax_0 + 2n \omega h + \omega h)}{4n \sin(\omega h)}. \end{aligned}$$

де n – кількість точок у масиві даних.

При виконанні умови Найквіста-Котельнікова в наведених формулах швидкого розрахунку запобігається ситуація рівності знаменників дробів нулю.

3. Пошук рівняння регресії експериментальних даних у вигляді тригонометричного поліному. Нехай, маючи вхідну функцію $f_0(x)$, потрібно представити її тригонометричним многочленом з, можливо, некрatними частотами. Отримаємо початкове наближення $g_0(x)$ функції $f_0(x)$ і розглянемо функцію $f_1(x_i) = f_0(x_i) - g_0(x_i)$, яка задає нев'язку наближення. Наступний крок наближення функції $f_1(x_i)$ дасть функцію $g_1(x)$ та нову функцію $f_2(x_i)$. Продовжуючи покроковий процес, отримаємо середньоквадратичне наближення:

$$f(x_i) \approx \sum_{k=0}^K g_k(x_i), \quad (5)$$

де $K+1$ – кількість врахованих гармонік.

4. Використання рівняння регресії у формі тригонометричного поліному для екстраполяції. Вважатимемо досліджуваний сигнал випадковим, тобто, що він містить у собі випадковий набір гармонічних коливань. Такий процес є ергодичним і для нього справедливе співвідношення:

$$\tau_p = \frac{1}{R_y(0)} \int_0^{\infty} R_y(s) |ds|. \quad (6)$$

Тут τ_p – порядок часу, на який має сенс прогнозування процесу, а

$$R_y(s) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} y(x)y(x-s)dx - \left(\lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} y(x)dx \right)^2$$

є кореляційною функцією досліджуваного сигналу [1].

При пошуку рівняння регресії у вигляді тригонометричного поліному за алгоритмом, наведеним у п. 3, автоматично, як побічний результат обчислень, дістанемо середнє квадратичне відхилення $I_k(\omega_k)$ і, також, таблиця функції відхилень $f_{k+1}(x_i)$.

Якщо прийняти відхилення $f_{k+1}(x_i)$ випадковою величиною, підпорядкованою нормальному закону розподілу з нульовим математичним сподіванням та середнім квадратичним відхиленням $I_k(\omega_k)$, можна оцінити час надійного прогнозування. Нехай допустима норма відхилення прогнозу є b з ймовірністю P_H ; тоді ймовірність отримання прогнозу в межах допустимих значень буде

$$P_H \leq \left(\frac{1}{\sqrt{2\pi \cdot I_k(\omega_k)}} \int_{-b}^b \exp(-l^2 / (2I_k(\omega_k))) dl \right)^n,$$

де n – кількість кроків прогнозу; l – змінна інтегрування. В цьому випадку час прогнозування $t_H = hn$ з ймовірністю P_H є надійним для дозволеного відхилення $[-b; b]$.

Для більш надійної оцінки потрібно врахувати, що середньоквадратичне відхилення $I_k(\omega_k)$ отримане з N випробувань і не є точним значенням, але лежить з певною надійністю в околі значення $I_k(\omega_k)$.

5. Практичне застосування методики частотно-амплітудного аналізу для прогнозування. На базі запропонованої методики розроблено програму забезпечення для прогнозування поведінки майже періодичних процесів. Програма, маючи на вході масив даних аргумент-значення функції, проводить частотний аналіз, будуючи регресійне рівняння у вигляді тригонометричного поліному з шуканими частотами. Це рівняння використовується при побудові прогнозу поведінки системи.

При створенні програмного продукту використовувалось середовище програмування C++ Builder 5 з метою спрощення побудови інтерфейсної

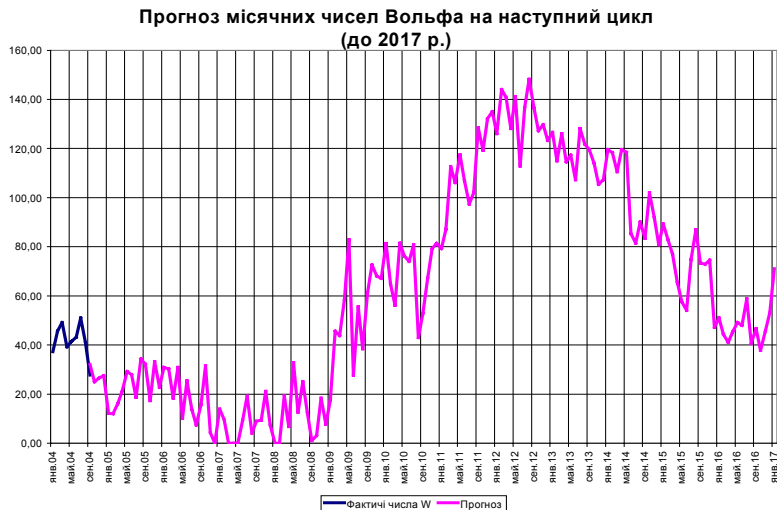
частини. Програма має обмеження на кількість вхідних даних у текстовому форматі до 64 Кб та кількість шуканих частот обмежена до 255, що при необхідності можна й збільшити. Для унаочнення проведення аналізу передбачено вивід графіку за вхідними даними та одночасний вивід графіку наближення з прогнозом. Окремий модуль дозволяє робити розрахунки за побудованим наближенням та видавати результат у вигляді текстової таблиці, що легко переноситься до Excel, MathCad та інших математичних пакетів.

Алгоритм проведення розрахунків було побудовано для аналізу даних, що не обов'язково є рівномірними в часі, тобто $h=x_{i+1}-x_i \neq \text{const}$. Завдяки такому рішенню, знижуючи швидкість розрахунків, метод зроблено нечутливим до змін інтервалу вимірів x_i .

Для визначення взаємної кореляції двох майже періодичних процесів в програму додано окремий модуль, який не лише визначає ступінь кореляції, а й оптимальний час запізнення процесу-наслідку відносно процесу-причини [1].

Авторами було використано програмний продукт для аналізу та прогнозу змін сонячної активності (СА) та її вплив на соціальні та природні процеси. Одним з результатів досліджень стало виявлення значного зв'язку зміни СА зі зміною середньої успішності учнів та студентів, врожайності основних сільськогосподарських культур, соціальної активності людських мас. Прогнозування СА є дуже важливою задачею, бо знаючи наслідки її змін, можна будувати прогнози зростання чи спаду врожаїв, кількості нещасливих випадків та аварій, успішності учнів, активності людських мас.

Останнього часу в ряді СМІ публікуються прогнози авторів впливу СА на стан фізичного та психічного здоров'я людей, враховуючи закони психології, на два тижні [3].



Як приклад наведемо прогноз середньомісячних значень чисел Вольфа – характеристику СА до 2017 року. Він побудований на основі даних за 64 роки, коли $n=748$. Очевидно, що наступний мінімум очікується з середини 2006 року до початку 2009 року. Тому наступні вибори до Верховної Ради України відбудуться на етапі спокою в суспільстві; явка виборців на дільниці буде порівняно невисокою. Наступний підйом СА сприятиме активності мас під час президентських виборів 2009 року. Максимум СА прогнозується з початку 2012 до середини 2014 року, після чого очікується наступний нерівномірний спад СА.

У нашій доповіді [2] був наведений прогноз щоденних значень СА на 2004 рік. Він непогано справдився, зокрема, різке зростання чисел Вольфа було в кінці жовтня, перед першим туром президентських перегонів.

Література:

1. Сергиенко А.Б. Цифровая обработка сигналов. – Санкт-Петербург: Питер, 2002. – 604 с.
2. Дресев О.М., Філер З.Ю. Спектральний аналіз майже періодичних сигналів // Теорія та методика навчання математики, фізики, інформатики: Збірник наукових праць. Вип. 4: Т. 3: Теорія та методика навчання інформатики. – Кривий Ріг: Видавничий відділ НМетАУ, 2004. – С. 64-70.
3. Філер З., Дресев О. Людина і Сонце. Прогноз здоров'я // Моя Батьківщина, вересень 2004 р. – С. 8.

ПЕДАГОГІЧНІ ОСНОВИ ВИКОРИСТАННЯ АКТИВНИХ МЕТОДІВ У НАВЧАННІ ІНФОРМАТИКИ

А.П. Забарна

м. Сміла, Природничо-математичний ліцей
zam@smela.com.ua, liceum@ck.ukrtel.net

1. Загальна постановка проблеми та її актуальність

Бажання зробити навчальний процес орієнтованим на особистість учня спонукає вчителів до пошуку інноваційних методів в освіті та до впровадження в навчально-виховний процес різноманітних освітніх технологій, таких, як технології саморозвитку, технології модульно-розвивального навчання, технології формування творчої особистості, проектної технології, нових інформаційних технологій навчання та інших. Проте будь-яка педагогічна технологія в загальнопедагогічному розумінні характеризує цілісний освітній процес з його метою, змістом і методами навчання. Крім традиційних [2], у практиці педагогіки середньої школи за останні роки великого розповсюдження та застосування набули й інші методи навчання, які називають нетрадиційними. Серед них певної уваги заслуговують так звані “активні” методи навчання.

2. Аналіз останніх досліджень

Огляд бібліографії, присвяченої активним методам навчання, дав можливість зробити висновок, що на сьогодні є кілька класифікацій активних методів навчання і майже не зустрічаються розробки із практичними рекомендаціями щодо застосування активних методів у навчанні інформатики. Частіше всього наводиться така класифікація активних методів навчання [1, 3, 4, 7, 8]:



Імітаційні заняття обов’язково передбачають імітацію індивідуальної або колективної діяльності у вибраній сфері, наявність імітаційної моделі об’єкту, процесу, діяльності. До ігрового імітаційного навчання відносять ділові ігри, розігрування ролей, ігрове проектування. До неігрових імітаційних занять відносять метод конкретних ситуацій, імітаційні вправи, індиві-

дуальні тренінги.

Неімітаційні заняття не вимагають такої моделі, і активізація тут забезпечується системою діючих прямих та зворотних зв'язків між учнем та вчителем. До неімітаційних методів більшість вчених відносять проблемне навчання, проектну діяльність, лабораторну роботу, практичне заняття, семінари, конференції.

Основними особливостями активних методів навчання є:

- забезпечення “вимушеної” активізації діяльності учня, навіть незалежно від його суб'єктивного бажання;
- забезпечення підвищеного ступеню мотивації та емоційності;
- забезпечення прямих та зворотних зв'язків по взаємодії учня з учителем або при рольовому колективному навчанні – один з одним.

Активні методи навчання являють собою технологію, спрямовану на формування і розвиток творчої особистості. Активний метод пізнання – це одночасно процес і результат активної діяльності учня. Найважливішими умовами активізації методів і форм навчання вважаються врахування рівня розвитку інтересів школярів, їхня співпраця між собою і з учителем, внесення елементів ігрової діяльності в навчальний процес. Активні методи навчання дозволяють особистості самій «будувати» свої знання, активно і творчо користуватися ними в житті як своїм надбанням для формування особистості.

Для активних методів навчання характерним є [5]:

- відношення вчителя до учня як до собі рівного;
- не просте повідомлення знань як беззаперечних істин, а самостійне “будівництво” знання учнем засобом критичного відношення до існуючих відомостей, інформації тощо, самостійного розв'язання творчих задач;
- плюралізм думок, підходів, поважне відношення до думки іншого при вирішенні певної проблеми.

У практиці педагогіки середньої школи за останні роки великого розповсюдження та застосування набули такі активні методи навчання, як метод проєктів, ділова гра, аналіз конкретних ситуацій, метод ігрового проєктування, стажування, проблемна лекція, метод активного програмового навчання тощо. Автор статті має певний досвід застосування активних методів у навчанні інформатики. Так протягом останніх десяти років учні-члени Смілянського відділення Малої академії наук працюють над дослідними навчальними проєктами. Результатом учнівських розробок стали програмні засоби “Астрономія для всіх”, “Графічна інтерпретація розв'язків систем нерівностей”, “Екологічний практикум”, “Японський кросворд” та ін. Крім цього учні Смілянського природничо-математичного ліцею беруть участь в телекомунікаційних проєктах програми I*EARN (Міжнародна освітня та ресурсна мережа). Цікаві дослідницькі роботи були проведені в проєктах “Дніпра жива вода”, “Куди прямує світ?”, “Ця незвичайна звичайна парабола”, “Цікава екологія”, “Ми – сучасники”, “Я – господар”, “За чистоту рід-

ної мови”, “Історія інформатики для допитливих” тощо.

3. *Невирішені частини проблеми та мета роботи (постановка завдання)*

Застосовуючи на уроках активні методи навчання, вчитель прагне створити такі умови, в яких кожному учневі буде надано можливість повірити в те, що немає жодних обмежень для його творчості. В цьому випадку і уява дитини піде шляхом пошуку істинного рішення. При активних методах навчання іноді паралельно відбувається робота в парах, в групах та індивідуальна робота.

Розглянемо деякі аспекти методичної системи навчання інформатики, що дають підстави стверджувати, що використання активних методів у навчанні інформатики є доцільним та ефективним.

4. *Виклад та обґрунтування основного матеріалу дослідження*

Серед найістотніших особливостей шкільного курсу інформатики Морзе Н.В. виділяє такі [6]:

- міжпредметний характер курсу;
- комп’ютер на уроках інформатики є і об’єктом навчання, і одночасно засобом навчально-пізнавальної діяльності, і інструментом для вирішення навчальних задач;
- зростає роль організації самостійної роботи учнів, оскільки з’являються можливості значної інтенсифікації навчального процесу та активізації навчально-пізнавальної діяльності;
- суттєво зростає роль учителя в управлінні навчально-пізнавальною діяльністю учнів і навчальним процесом узагалі, оскільки в умовах інтенсифікації навчання і активізації навчально-пізнавальної діяльності частіше виникають проблемні ситуації і питання, розв’язання яких потребує втручання та участі учителя;
- на відміну від інших предметів, в інформатиці трапляються випадки, коли окремі питання учні можуть знати краще, ніж учитель, тобто відбувається взаємонавчання учня і вчителя.

На нашу думку, врахувати та використати ці особливості для підвищення ефективності навчання інформатики дозволить застосування активних методів навчання.

Як відомо, принципи навчання дають відповідь на запитання “Як організувати навчальний процес?”. Принципи навчання – це керівні ідеї, нормативні вимоги до організації і здійснення освітнього процесу. Дидактими [6] (М.П. Скаткін, М.А. Данилов, І.Я. Лернер, Ю.К. Бабанський, М.І. Махмутов) доведено, що для ефективності навчання незалежно від предмету необхідно керуватися загально дидактичними принципами і правилами навчання. Дидактичні принципи виражають закономірності процесу навчання і дотримання їх є необхідною умовою успіху педагогічної діяльності вчителя. Виходячи з цього, можна зробити висновки, що методика навчання інформатики повинна спиратися на закономірності процесів формування в учнів

знань, умінь і навичок і одночасно відповідати загально дидактичним принципам навчання. Отже, у зв'язку з особливостями навчальної діяльності учнів на уроках інформатики важливим постає питання реалізації деяких дидактичних принципів, серед яких – принцип активності і самостійності та принцип індивідуалізації і колективності навчання. Розглянемо основні положення цих принципів, які викладено у методичній літературі (див., наприклад, [6]).

Принцип активності і самостійності. Знання – це не річ, не предмет, який можна якось передати учням. Це якісна зміна особистості, яка відбувається лише внаслідок власної цілеспрямованої активності того, хто навчається. Активність учня реалізується через його діяльність.

Принцип активності в навчанні інформатики передбачає розумову активність учнів у процесі оволодіння діяльністю. Цей принцип має велике значення для правильної організації процесу навчання. В методиці навчання інформатики розрізняють інтелектуальну та емоційну активність. Інтелектуальна активність забезпечується постановкою проблемних завдань, які стимулюють пізнавальний інтерес. Емоційна активність пов'язана з питанням: цікаво чи не цікаво учням вивчати ті чи інші питання. Позитивне емоційне ставлення до предмету відіграє велику роль у досягненні успіхів. Негативні емоції блокують активність учня.

Форми прояву активності можуть бути різними, наприклад: самоконтроль через рефлексію власної діяльності, контроль за роботою товариша, модифікація готових і розробка власних алгоритмів та програм, участь у проектній діяльності тощо. Навчання ефективне, коли учень розуміє необхідність свого навчання, ставить чи приймає цілі заняття, бере участь у плануванні й організації своєї діяльності, в її розумінні, самоконтролі і самооцінці.

Для реалізації принципу активності вчителю доцільно:

- учити дітей ставити запитання, оскільки запитання, в деяких ситуаціях, важливіше за відповідь, бо відкриває, а не закриває пізнання;
- у навчанні завжди використовувати альтернативні підходи, позиції і точки зору до будь-якого суттєвого питання;
- знаходити з учнями причини того, що вивчається, спонукати гіпотези учнів, працювати з їхніми версіями як з особистісним змістом освіти;
- у процесі навчання робити зупинки і пропонувати учням проаналізувати їхню діяльність (рефлексивна позиція потребує часу, але збільшує результативність засвоєння курсу інформатики).

Самостійність учня також є метою і умовою успішного вивчення інформатики. Вона слідує за активністю: учень може активно сприймати лекцію проблемного типу, але це ще не самостійність. Можливі етапи зростання самостійності: від повного управління вчителем через оперативну допомогу до самоуправління пізнавальною діяльністю за допомогою комп'ютера. При переході до творчої діяльності самостійність реалізується

повністю. Звертання учня за допомогою є проявом активності, але це ще не самостійність.

Внаслідок зростання самостійності зростають і продуктивність навчання, вміння самостійно знаходити виходи зі скрутних ситуацій, користуватися літературою і комп'ютерними засобами допомоги. Ознакою високого рівня самостійності, окрім інших, є також пошукова діяльність за комп'ютером, у комп'ютерній мережі за допомогою звернення до пошукових машин.

Принцип індивідуалізації і колективності навчання. Індивідуалізація і колективність навчання доповнюють одне одного, особливо при навчанні інформатики. Тільки організувавши колективну або групову роботу, можна знайти час для занять з сильнішим і слабшими учнями.

Індивідуалізація визначається як організація навчального процесу, під час якого при доборі способів, прийомів, темпу навчання враховуються індивідуальні особливості учнів, рівень розвитку їхніх здібностей до навчання.

Індивідуалізацію можна розглядати з точки зору процесу навчання, змісту навчання і будови шкільної системи. Перша з цих точок зору передбачає вибір форм, методів і прийомів навчання, друга – створення навчальних планів, програм, навчальної літератури і складання завдань для учнів, третя – формування різних типів шкіл і класів. Індивідуалізація можлива під час роботи учнів з програмними засобами, що вивчаються, зі своїм індивідуальним темпом, своїми шляхами подолання труднощів та за допомогою гнучкого налагодження навчальної програми. Принцип індивідуалізації реалізується у навчанні інформатики шляхом урахування індивідуально-психологічних особливостей учнів, які значно впливають на успішність оволодіння основами інформаційної культури. Забезпечення індивідуалізації навчання можливе тільки за умови обізнаності вчителя з такими особливостями своїх учнів та способами індивідуалізованого навчання.

Для реалізації принципів активності і самостійності, індивідуалізації і колективності можна рекомендувати вчителям інформатики ширше використовувати саме активні методи навчання.

5. Висновки та перспективи подальшого розвитку програми

Якщо говорити про перспективи шкільного курсу інформатики, то на сьогодні виокремлюється чотири напрями щодо цілей цього курсу [6]:

- практичний (базові поняття – комп'ютер та його інформаційне забезпечення);
- програмістський (базові поняття – алгоритм, програма);
- науково-технічний (базові поняття – повідомлення, інформація, інформаційні процеси);
- дослідницький (базові поняття – модель, моделювання, експеримент, імітація, спостереження тощо).

Дослідницький напрям передбачає те, що зміст курсу інформатики мо-

же і повинен бути направлений на формування і розвиток творчих здібностей, дослідницьких умінь і навичок школяра. На уроках інформатики школяр, як справжній дослідник, спостерігає різноманітні явища і процеси, на основі спостережень висуває гіпотезу, перевіряє її, робить відповідні висновки й узагальнення, синтезує нові знання, нову інформацію. В дослідницькому напрямі ключовим словом є – творчість. Саме виникнення дослідницького напрямку дає підстави стверджувати, що застосування активних методів навчання дозволить більше реалізувати творчий потенціал кожного учня.

Отже, для реалізації принципів активності і самостійності, індивідуалізації та колективності навчання, врахування особливостей курсу інформатики та розвитку його дослідницького напрямку ми вважаємо доцільним використання активних методів навчання на уроках інформатики. Сучасна педагогічна наука має розробити теоретичні основи методичної системи навчання інформатики з використанням активних методів навчання та надати практичні рекомендації їх застосування у навчальному процесі в умовах профілізації школи.

Література:

1. Алексюк А.М. Методи навчання і методи учіння. – К.: Знання, 1980. – 48 с.
2. Бабанский Ю.К. Методы обучения в современной общеобразовательной школе. – М.: Просвещение, 1985. – 208 с.
3. Бессонова Г.А. Активные формы и методы обучения в вузе. – <http://www.kamcoop.ru/mucckb/naukametod1.htm>, 11.11.2004 р.
4. Бондарчук Л.І. Федорчук Е.І. Методи активного навчання в курсі “Основи педагогічної майстерності” // Вища і середня пед. освіта. – 1993. – №16.
5. Братцева Г.Г. Активные методы обучения и их влияние на смену педагогической парадигмы. – http://anthropology.ru/ru/texts/bratseva/educphil_45.html, 11.11.2004 р.
6. Морзе Н.В. Методика навчання інформатики: Навч. посіб.: У 4 ч./ За ред. акад. М.І. Жалдака. – К.: Навчальна книга, 2003. – Ч. I.: Загальна методика навчання інформатики. – 254 с.: іл.
7. Немеш Н.В. Методи навчання у вищій школі. – <http://www.socwd.uzhgorod.ua/Herald/herald3/28.htm>, 1.10.2004 р.
8. Новые педагогические и информационные технологии в системе образования: Учеб. пособие для студ. пед. вузов и системы повыш. квалиф. пед. кадров / Е.С. Полат, М.Ю. Бухаркина, М.В. Моисеева, А.Е. Петров; Под ред. Е.С. Полат. – М.: Издательский центр «Академия», 2003. – 272 с.

ПРИМЕНЕНИЕ ЧИСЛЕННЫХ МЕТОДОВ К ЗАДАЧАМ ОБЩЕГО КУРСА ФИЗИКИ СО СЛОЖНОЙ ЗАВИСИМОСТЬЮ МЕЖДУ ПАРАМЕТРАМИ

З.Г. Зуйкова

г. Донецк, Донецкий национальный университет

tisn55@skif.net

Изложение курса «Численные методы» для студентов специализации «Физика и основы информатики» имеет свои особенности и цели. Одна из них – выявить дополнительные возможности, которые предоставляют изучаемые методы при решении традиционных задач курса общей физики.

Например, численное интегрирование и построение интерполяционного полинома применяются для функций, заданных либо таблично, либо аналитически. Последнее проиллюстрируем на одной из трех задач, относящихся к разным разделам курса физики, но требующим единообразного подхода к решению. Простейшие варианты этих задач взяты из сборника [1], рекомендуемого для студентов физических и инженерно-технических специальностей вузов.

1. Найти массу m идеального газа с молярной массой μ , который находится под давлением P между двумя одинаковыми горизонтальными пластинами площадью S . Температура газа растет линейно от T_1 у нижней пластины до T_2 у верхней. Расстояние между пластинами H . (Ответ: $m=(P\mu S/R)Z$).

2. Найти емкость C плоского конденсатора, зазор между обкладками которого заполнен изотропным диэлектриком, чья проницаемость ε изменяется в перпендикулярном к обкладкам направлении по линейному закону от ε_1 до ε_2 , причем $\varepsilon_1 < \varepsilon_2$. Площадь каждой обкладки S , расстояние между ними H . (Ответ: $C=\varepsilon_0 S/Z$).

3. Найти сопротивление R_k плоского конденсатора, зазор между пластинами которого заполнен неоднородной слабо проводящей средой, удельная проводимость которой изменяется в направлении, перпендикулярном к пластинам, по линейному закону от σ_1 до σ_2 . Площадь каждой пластины S , ширина зазора H . (Ответ: $R_k=Z/S$).

(R – универсальная газовая постоянная, ε_0 – электрическая постоянная).

Для ответа на вопрос каждой задачи необходимо вычислить интеграл вида

$$Z = \int_0^H \frac{dx}{f(x)}, \quad (1)$$

где в качестве функции f координаты x выступают температура T , диэлектрическая проницаемость ε и удельная проводимость σ соответственно. Ось x перпендикулярна пластинам, начало координат совмещено с нижней пластиной, другая имеет координату H .

Если $f(x)$ – линейная функция, как оговорено в условии, то вычислить интеграл достаточно просто. Однако, усложнив ее вид, сталкиваемся с затруднениями, при которых численное интегрирование оказывается более предпочтительным или даже единственно возможным.

Рассмотрим первую из задач. При произвольной функции $f(x)=T(x)$ интеграл (1) можно рассчитать численно, воспользовавшись, например, формулой трапеций [2]:

$$Z \approx h((U_0+U_n)/2+U_1+U_2+\dots+U_{n-1}), \quad (2)$$

где $U_j=T(x_j)$; $x_j=jh$, $j=0, 1, 2, \dots, n$. n – число равных частей, на которые разбит отрезок интегрирования $[0, H]$, $h=H/n$ – шаг интегрирования.

Z является функционалом функции $f(x)=T(x)$ с фиксированными значениями T_1 и T_2 на концах отрезка. Если рассмотреть функции вида

$$T(x)=T_1+(T_2-T_1)(x/H)^y, \quad (3)$$

зависящие от параметра y ($y>0$), то интеграл (1) станет уже функцией этого параметра: $Z=Z(y)$.

Вычислив для нескольких его значений интеграл по формуле (2), получим набор узлов интерполяции y_i ($i=0, 1, \dots, k$) со значениями функции в них $Z_i=Z(y_i)$.

Теперь можно решить обратную задачу: найти параметр y , соответствующий произвольному Z из рассчитанного диапазона его изменения. Для этого используем интерполяционный полином Лагранжа [2]:

$$y = \sum_{i=0}^k y_i \frac{(Z-Z_0)(Z-Z_1)\dots(Z-Z_k)}{(Z_i-Z_0)(Z_i-Z_1)\dots(Z_i-Z_k)}. \quad (4)$$

Таким образом, найдем распределение температуры (3), при котором масса газа равна $m=(P\mu S/R)Z$.

Повторно вычислим интеграл по формуле (2) с найденным значением y . Обозначим результат Z_h .

Студентам рекомендуется провести численный эксперимент и исследовать влияние шага интегрирования h и числа узлов интерполяции k на величину отклонения $\delta=|Z-Z_h|$.

Литература:

1. Иродов И.Е. Задачи по общей физике. – М.: Наука, 1988. – 416 с.
2. Копченова Н.В., Марон И.А. Вычислительная математика в примерах и задачах. – М.: Наука, 1972. – 368 с.

МЕТОДИКА ВИКЛАДАННЯ ДИСЦИПЛІНИ “ІНФОРМАТИКА ТА КОМП’ЮТЕРНА ТЕХНІКА” З УРАХУВАННЯМ ВИМОГ БОЛОНСЬКОЇ ДЕКЛАРАЦІЇ

О.Ф. Клименко^а, Н.Р.Головко^б

м. Київ, Київський національний економічний університет

^а OFKlim@online.com.ua

^б GolovkoNat@ukr.net

Згідно рішення колегії Міністерства освіти і науки України “Про проведення педагогічного експерименту щодо запровадження кредитно-модульної системи організації навчального процесу”, кінцевою метою цього експерименту є підвищення якості освітніх послуг та конкурентноспроможності фахівців, тобто вироблення і запровадження нових підходів до організації навчального процесу, забезпечення гнучкості системи підготовки фахівців, опрацювання системи залікових кредитів по дисциплінах навчального плану на принципах, що передбачені Європейською Системою залікових кредитів (ECTS – European Credit Transfer System) або Європейської кредитно-трансферної та акумулюючої системи (СКТАС) [1].

Розвиток та упровадження у всі сфери економіки сучасних інформаційних технологій викликають необхідність постійно змінювати та удосконалювати програму дисципліни “Інформатика та комп’ютерна техніка”. В Київському національному економічному університеті на кафедрі інформатики під час викладання дисципліни “Інформатика та комп’ютерна техніка” особлива увага приділяється таким розділам: комп’ютерні мережі; глобальна мережа INTERNET; системи обробки тексту; системи табличної обробки даних; системи управління базами даних; офісне програмування; перспективи розвитку комп’ютерних технологій.

Розділ “Комп’ютерні мережі” містить топологію, апаратні засоби, методи доступу в локальних та глобальних комп’ютерних мережах, мережеву архітектуру та технології, мережеві операційні системи.

Розділ “Глобальна мережа INTERNET” містить класифікацію користувачів, способи підключення до INTERNET, служби і сервіси мережі INTERNET, доменні зони, IP–телефонія, захист інформації в INTERNET.

В розділі “Системи обробки тексту” увага приділяється роботі з графічними об’єктами та різними додатками, роботі з таблицями, створенню та використанню шаблонів.

Розділ “Системи табличної обробки даних” містить засоби фінансово-економічного аналізу: створення звітної, фінансової та економічної документації; створення зведених таблиць та сценаріїв; використання засобів роботи з базами та сховищами даних (пошук, вибірка, групування тощо); приділена увага розв’язанню оптимізаційних задач; прогнозуванню і статистичному аналізу; використанню макросів та функцій користувача.

Розділ “Системи управління базами даних” містить такі питання: архітектура системи баз даних; моделі баз даних; принципи та етапи проектування баз даних; створення, редагування та використання реляційних баз даних у середовищах СУБД “ORACLE” та “Microsoft Access”; структурована мова запитів SQL; створення макросів, модулів та меню; створення проєктів.

Розділ “Офісне програмування” містить два підрозділи: 1) Основи програмування мовою Visual Basic for Applications (Створення форм користувача, користувацький інтерфейс, автоматизація комп’ютерних проєктів; 2) Web–програмування (Засоби Web-програмування, основи HTML, створення Web-сторінок).

Розділ “Перспективи розвитку комп’ютерних технологій” містить два підрозділи: 1) Розвиток комп’ютерних технологій (Web-орієнтовані комп’ютерні технології, ієрархічні та мережеві моделі сховищ даних, бази знань та нейромережі); 2) Інтернет-технології в бізнесі (модель електронного бізнесу, електронні ринки, розвиток систем Intranet та Extranet, Web-інтеграція, інтерактивне банківське обслуговування, системи електронних платежів, безпека електронної комерції).

Важливими елементами Європейського простору вищої освіти є: постійне навчання протягом усього життя, а також мотивоване залучення студентів до навчання. Це насамперед стосується нової системи оцінювання знань студентів з урахуванням поточної успішності.

Система ЄКТАС [2] передбачає уніфікацію шкали оцінювання знань і введення семи оцінок:

Оцінка ЄКТАС	Значення оцінки
A	Відмінно – видатна робота з мінімальними помилками
B	Дуже добре – вище середнього стандарту, але з деякими поширеними помилками
C	Добре – в цілому хороша робота, але з помітними помилками
D	Задовільно – пристойно, але із значними помилками
E	Достатньо – задовольняє мінімальним вимогам
FX	Не прийнято – необхідне доопрацювання
F	Не прийнято – необхідно переробити

Згідно схваленого Вченою радою університету “Порядку оцінювання знань студентів з урахування поточної успішності” оцінювання знань студентів з дисципліни “Інформатика та комп’ютерна техніка” на кафедрі інформатики проводиться у двох напрямках:

1. Контроль систематичності та активності роботи студента протягом семестру над вивченням програмного матеріалу дисципліни.
2. Контроль за виконанням модульних завдань.
Перший напрямок контролю передбачає оцінку оперативної роботи:
 - відвідування практичних та лабораторних занять,

- активність та рівень знань при обговоренні питань за розглядуваною темою (оперативний контроль),
- виконання лабораторних робіт та їх захист,
- виконання практичних робіт,
- завдання для самостійного вивчення матеріалу,
- підготовка та захист рефератів,
- участь у студентських конференціях, тощо.

Другий напрямок контролю передбачає оцінку модульних завдань:

- результатів виконання контрольних тестів,
 - результатів контрольних робіт за модулями (блоками).
- Використовуються п'ять форм оцінювання поточної роботи студентів:
1. Контрольний тест (виконується на комп'ютері).
 2. Контрольна робота (виконується письмово).
 3. Оперативний контроль:
 - усне опитування,
 - письмовий тест,
 - тестове завдання на комп'ютері.
 4. Оцінка виконання та захисту лабораторних робіт та рефератів.
 5. Оцінка виконання практичних робіт.

Час проведення контрольних робіт та тестів визначений розробленою на кафедрі “Технологією проведення поточного контролю” [3].

Оперативний контроль проводиться під час занять, на яких “Технологією проведення поточного контролю” не передбачено проведення контрольних робіт та тестів, і оцінюється в 10, 5 та 0 балів.

Якщо проводяться дві оперативні форми під час одного заняття, тобто:

- усне опитування + письмовий тест, або
- усне опитування + тестове завдання на комп'ютері,

то кожна з цих форм оцінюються в 0 або 5 балів, а результати підсумовуються.

Якщо проводиться одна з форм, тоді ставляться оцінки 0, 5 або 10 балів. Проводити одну чи обидві форми одночасно вирішує викладач, що проводить практичне (лабораторне) заняття.

Дисципліна “Інформатика та комп'ютерна техніка” викладається на всіх факультетах усіх форм навчання протягом двох семестрів.

Всього максимально можлива кількість балів за перший семестр 400 балів.

Форма підсумкового контролю за 1 семестр – залік.

Отримана кількість балів за семестр перераховується, і згідно ухвали Вченої ради Київського національного економічного університету “Про заходи щодо реалізації положень Болонської декларації в навчальному процесі КНЕУ”, виставляється остаточна оцінка, при цьому розроблена така шкала переведення:

Отримана кількість балів	Перерахунок	Результат	
385–400	100	A	З а р а х о в а н о
365–380	95		
345–360	90		
325–340	85		
305–320	80		
285–300	75	C	
265–280	70		
245–260	65		
225–240	60		
205–220	55	D	
185–200	50		
165–180	45	FX	Н е з а р а х о - в а н о
145–160	40		
125–140	35		
105–120	30		
85–100	25		
65–80	20	F	
45–60	15		
25–40	10		
20	5		
0–15	0		

У другому семестрі оцінювання лабораторних робіт, рефератів, модульного та оперативного контролю проводиться аналогічно оцінюванню у першому семестрі, і максимальна кількість балів за другий семестр також складає 400 балів.

Отримана кількість балів у семестрі перераховується згідно поданої нижче таблиці.

Отримана кількість балів	Перерахунок
355–400	40
305–350	35
275–300	30
235–270	25
200–230	20
Менше 200	0

Форма підсумкового контролю за 2 семестр – іспит.

Іспит проводиться у письмовій формі і включає 6 завдань, кожне з яких оцінюється в 0, 5 або 10 балів. Загальна оцінка формується як сума балів, отриманих протягом 2 семестру і балів, отриманих за результатами іспиту.

Така система оцінювання знань, починаючи від “Технології проведення поточного контролю” і до підсумкових результатів іспиту ґрунтується на прозорості навчального процесу. Нині всі наші студенти вже на початку

семестру поінформовані про те, що вони мають опанувати, що від них вимагається, якими будуть критерії оцінювання їх знань, скільки балів і за що вони зможуть отримати під час поточних і підсумкових контрольних заходів.

Отримана кількість балів	Результат	
85–100	A	відмінно
80	B	добре
65–75	C	
50–60	D	задовільно
30–45	FX	незадовільно (перездача)
0–25	F	незадовільно (повторне прослуховування)

Література:

1. Павленко А.Ф. До європейського простору вищої освіти. – К.: КНЕУ.
2. Основні засади розвитку вищої освіти України в контексті Болонського процесу. – К.: КНЕУ
3. Клименко О.Ф., Красюк Ю.М., Петренко Л.М., Соловйова В.В. Робочий зошит з дисципліни “Інформатика та комп'ютерна техніка”. – К.: Брама, 2004. – 206 с.

СТАНОВЛЕННЯ ТВОРЧОЇ ПРОФЕСІЙНОЇ ДІЯЛЬНОСТІ СТУДЕНТІВ ВИЩИХ АГРАРНИХ НАВЧАЛЬНИХ ЗАКЛАДІВ У ПРОЦЕСІ ВИВЧЕННЯ ДИСЦИПЛІН ІНФОРМАЦІЙНОГО ЦИКЛУ

О.В. Ключко

м. Вінниця, Вінницький державний аграрний університет

klochkoob@rambler.ru

Оскільки розвиток агропромислового комплексу України потребує швидкого впровадження сучасних інформаційно-комунікаційних технологій у виробництво, прийняття оптимальних управлінських рішень, ефективного функціонування ринкових і фінансово-кредитних механізмів, то розв'язання питань інформатизації можливе за умов досконалого володіння спеціалістами АПК найновішими засобами комп'ютерної техніки, засобами інтегрованих програмних середовищ, на базі яких створюються сучасні інформаційні системи в АПК.

Метою групи дисциплін інформаційного циклу є засвоєння студентами сучасних інформаційних технологій в контексті оволодіння необхідним комплексом знань, умінь та навичок для запровадження цих технологій в практичну діяльність.

У процесі вивчення курсу дисциплін інформаційного циклу повинен відбуватись розвиток і творча реалізація студента як суб'єкта професійної праці.

Необхідно відзначити, що родовим поняттям творчості виступає діяльність, а видовим – соціальна значущість і новизна. Важливо з'ясувати, що варто розуміти під словом «нове». Під «новим» розуміються «цінності, що володіють суспільною значущістю», а також те, чого «ніколи не було». Л.С. Виготський підкреслює, що «... вищі вираження творчості доступні тільки деяким обраним геніям людства, – але в повсякденному навколишньому житті творчість є необхідна умова існування» [1].

Розрізняють чотири рівні творчості за соціальною значущістю і новизною:

I-й рівень процесу творчості приводить до принципово нового результату для всього людства або у космічному масштабі;

II-й рівень – результат є новим для досить великого кола людей, скажімо для певної країни світу;

III-й рівень – творчий продукт є новим для значно меншого, обмеженого кола людей;

IV-й рівень – новизна продукту є суб'єктивною, відносною, значущою тільки для самої людини, що творить.

У процесі навчання, як правило, реалізується четвертий рівень творчості, але обмежений обсяг цього рівня творчості на заважає йому бути чи не найважливішим, початковим етапом в оволодінні вищими рівнями творчос-

ті, формуванні вмій і навичок загальної креативності [2].

Викладання предметів інформаційного циклу у аграрних вищих навчальних закладах (АВНЗ) починається з першого курсу і продовжується 3-6 років в залежності від спеціалізації і рівня підготовки. Як правило, викладання предметів інформаційного циклу відбувається в АВНЗ у послідовності, представленої в таблиці 1.

Таблиця 1.

Агрономічний факультет	
1-й курс	“Інформатика і комп’ютерна техніка”, “Математика з основами моделювання”
2-й курс	–
3-й курс	–
4-й курс	–
5-й курс	“Інформаційні технології та комп’ютеризація агротехнологій”
Магістратура	“Математичне моделювання”, “Інформаційні технології”, “Розробка та управління проектами”
Зооінженерний факультет	
1-й курс	“Обчислювальна техніка і програмування”, “Прикладна математика”
2-й курс	–
3-й курс	–
4-й курс	–
5-й курс	“Інформаційні технології та комп’ютеризація агротехнологій”, “Моделювання технологічних процесів тваринництва”
Магістратура	“Математичне моделювання”, “Інформаційні технології”, “Розробка та управління проектами”
Факультет механізації сільського господарства	
1-й курс	“Основи комп’ютерної техніки і програмування”, “Прикладна математика”
2-й курс	–
3-й курс	–
4-й курс	“Системи автоматизованого управління”
5-й курс	“Науково-інформаційне забезпечення в АПК”
Магістратура	“Математичне моделювання”, “Математичні методи оптимізації та планування експерименту”
Економічний факультет	
1-й курс	“Основи комп’ютерної техніки і програмування”, “Прикладна математика”
2-й курс	“Дослідження операцій”, “Програмне забезпечення ЕОМ”
3-й курс	“Інформаційні системи в менеджменті”, “Математичне програмування”, “Логістика”, “Економетрія”

4-й курс	“Методи оптимізації в економіці”, “Прогнозування”, “Комп’ютерні мережі”
5-й курс	“Автоматизовані робочі місця спеціаліста”, “Теорія систем”, “Проектування макроекономічних систем”
Магістратура	“Проектування баз даних”, “Математичне моделювання”, “Математичні методи оптимізації та планування експерименту”, “Інформатика для економістів”
Обліково-фінансовий факультет	
1-й курс	“Інформатика і комп’ютерна техніка”, “Прикладна математика”
2-й курс	“Дослідження операцій”, “Програмне забезпечення ЕОМ”, “Інструментальні програми”
3-й курс	“Математичне програмування”, “Інформаційні системи та технології обліку і аудиту”
4-й курс	“Математичне програмування”, “Комп’ютерні мережі”
5-й курс	“Автоматизовані робочі місця спеціаліста”, “Теорія систем”, “Проектування макроекономічних систем”
Магістратура	“Проектування баз даних”, “Математичне моделювання”, “Математичні методи оптимізації та планування експерименту”, “Інформатика для економістів”

Логічна структура навчання творчості, як логічна структура послідовного викладання предметів інформаційного циклу являє собою єдність етапів, що послідовно розвиваються:

Неодмінною умовою досягнення творчого підходу в фаховій діяльності є збір, аналіз, обробка і використання інформації. Спеціаліст повинен володіти як високорозвинутою індивідуальною культурою обробки інформації, так і вміти організовувати й управляти інформаційними потоками в реальній ситуації, у тому числі за допомогою сучасних комп’ютерних технологій. На першому-другому курсах з метою реалізації даної задачі викладаються предмети “Інформатика і комп’ютерна техніка”, “Обчислювальна техніка і програмування”, “Програмне забезпечення ЕОМ”, “Інструментальні програми” і подібні. Основною метою яких є формування комп’ютерної грамотності студентів, тобто формування сукупності знань, умінь та навичок, які забезпечують студентам можливість застосовувати обчислювальну техніку у навчанні, а згодом у професійній діяльності [3].

На другому, третьому, четвертому курсах студентам пропонуються дисципліни “Дослідження операцій”, “Математичне програмування”, “Інформаційні системи”, “Методи оптимізації”, “Економетрія”, “Прогнозування” і подібні. Основною метою яких є вивчення в систематизованій формі та активне засвоєння студентами основних методів аналізу, прогнозування, побудови економічних моделей, методів пошуку оптимальних рішень на множині допустимих варіантів у широкому спектрі теорети-

ко-економічних та практичних проблем на всіх рівнях ієрархії управління. Спеціалісти зобов'язані не лише знати експлуатаційні можливості сучасної обчислювальної і організаційної техніки, але й володіти реальними способами вирішення поставлених задач з використанням сучасних засобів автоматизації. У процесі вивчення цих дисциплін відбувається інтелектуально-творчий тренінг, тобто впроваджується розгорнута система гнучких впливів на особистість, спрямована на формування здатності до творчості. Психологічні особливості винахідницької діяльності (самостійна постановка задач, неможливість правильного рішення – “відповіді”, їх потенціальна варіантність, багаторівневість винахідницької задачі та ін.) сприяють тому, що навіть у поставленій й розв'язаній задачі тривалий час зберігається “інерція” об'єктивності її творчого змісту, об'єктивної новизни. Саме цей заряд об'єктивної новизни, що міститься у винахідницькій задачі, потрібний для розв'язання творчої діяльності й зберігання всіх психологічних особливостей об'єктивної творчості [2].

На четвертому, п'ятому курсах, магістратурі у процесі вивчення дисциплін “Автоматизовані робочі місця спеціаліста”, “Теорія систем”, “Проектування макроекономічних систем”, “Математичне моделювання”, “Математичні методи оптимізації та планування експерименту”, дипломного проектування і подібних відбувається засвоєння системних уявлень про процеси функціонування і управління підприємством АПК і організацію автоматизації даних процесів за допомогою сучасної обчислювальної техніки, прикладного програмного забезпечення та інструментальних засобів, поглиблення практичних навичок застосування комп'ютерних технологій на підприємствах АПК для організації процесів управління підприємством. Відбувається професійно-вибіркове розпізнання й осмислення, усвідомлення проблемної ситуації і постановка проблеми; логічне обґрунтування висунутої творчої ідеї; реалізація творчої ідеї на практиці. У результаті чого реалізується основна мета навчання - є розвиток і творча реалізація людини як суб'єкта професійної праці.

Логіка розвитку фахової творчості може бути подана в такій послідовності [4]:

- професійно-вибіркове розпізнавання, осмислення й усвідомлення проблемної ситуації;
- виділення головної проблеми, відділення її від другорядної, постановка проблеми;
- пошук підходів до вирішення проблеми, вибір стратегії вирішення проблеми;
- розробка ідеальної моделі плану реалізації обраної стратегії;
- логічне і математичне обґрунтування прийнятої ідеальної моделі;
- вирішення проблеми через реалізацію творчої ідеї на практиці.

Якісні вираження фахової творчості визначаються світоглядною, методологічною і методичною озброєністю суб'єкта фахової діяльності, вищим

ступенем оволодіння своїм фахом. Критеріями фахової творчості можна визнати такі:

- філософські (світоглядна позиція, творче фахове пізнання);
- психолого-педагогічні (зміст і рівень мотивації, фахова спрямованість особистості на творчість, соціальна установка особистості, особливості самооцінки в системі відношень її до різноманітних сторін діяльності, рівень розвитку творчого фахового мислення, уяви і спроможностей до класифікації і систематизації, рівень підготовленості до фахової діяльності);
- акмеологічні (прагнення до творчого професіоналізму, саморозвитку і самовдосконалення, професійно-моральна активність) і т.п.

Таким чином, фахова творчість є елементом теорії пізнання, фахового мислення, здібності до продуктивної спеціальної діяльності. У розумовій діяльності необхідно виділяти інтелектуально-логічну й інтелектуально-евристичну діяльність, спрямовану на вирішення власне творчих задач у звичайних і екстремальних умовах. Процес творчого мислення реалізується на основі єдності емоційно-почуттєвого і раціонального пізнання за допомогою строго логічних і логіко-психологічних (інтуїтивних) засобів. Він здійснюється як єдність суб'єктивізації об'єктивного й об'єктивізації суб'єктивного, що відображує єдність розумової і практичної діяльності.

Оскільки творчість – це вид людської діяльності, то у процесі навчання предметів інформаційного циклу студенти вчать виявляти мету творчості (творчої задачі), а також об'єктивні (соціальні, матеріальні) та суб'єктивні (особистісні якості – знання, уміння, позитивна мотивація, творчі якості) передумови для творчості.

У навчальній діяльності студентів виникають суперечності, пов'язані з розходженнями між новими пізнавальними цілями, завданнями і попереднім досвідом студентів, між узагальненнями, які вже склалися, і новими фактами. Навчання систематично спричинює виникнення внутрішніх суперечностей, їх усвідомлення студентами, розгортання активної діяльності, спрямованої на їх усунення. Подолання кожної суперечності вимагає розв'язання нових для студентів завдань пошуку нових способів дій, формування досконаліших операцій, тобто здійснення наступного кроку вперед у розвитку пізнавальної діяльності особистості. Такі суперечності притаманні і трудовій діяльності студентів, у якій поєднуються пізнання і праця, проектується майбутній продукт, виконуються дії, спрямовані на його виготовлення [2].

Використання НІТН переконує студентів в тому, що за допомогою комп'ютерів можна не лише виконувати обчислення, але й застосовувати їх в процесі розв'язування важливих практичних і теоретичних завдань, які потребують використання творчих методів, різноманітних форм діяльності фахівця. Студенти залучаються до різноманітної діяльності з навчальним матеріалом, переконуються в тому, що використання комп'ютера створює умови для розв'язування складних творчих завдань.

Форми і засоби розвитку майбутнього спеціаліста в значній мірі визначаються його концептуально-психологічним базисом. Для цього доцільно виділити два основних принципи розробки, організації і реалізації стратегії навчання:

1) студент спроможний до продуктивного фахового зростання там і тоді, де і коли йому надана можливість для стимульованого творчого розвитку, у процесі якого будується досвід досягнення, здійснюється осмислення, створення продукту, що ведуть до самостійного вибору нових задач і цілей. Інакше кажучи, успішний розвиток творчого потенціалу студента передбачає формування добре рефлексованого досвіду;

2) розвиток творчого потенціалу студента й у цілому професійної спрямованості може бути продуктивним тільки при наявності взаємозв'язку між його змістом і засобами, з одного боку, і змістом, цілями навчання - з іншого.

У якості основних задач здійснення продуктивного розвитку творчого потенціалу студента можна виділити:

- розширення можливостей компетентного вибору кожним студентом різноманітних напрямків навчальної діяльності, що найкраще відповідають стратегії навчання через формування індивідуальних навичок, умінь організації перспективи майбутньої професійної діяльності;

- забезпечення можливостей для пошуку відповідності задач професії до власних потреб, а також шляхів і засобів змісто- і цілетворення, що визначають формування відповідної готовності і підготовленості до досягнення мети.

Своє, найбільш цілісне, вираження можливості вирішення цих задач набуває у формі розробки і реалізації навчальних середовищ, призначених для здійснення ефективної підготовки спеціалістів. При цьому її стратегія полягає у здійсненні різноманітних взаємодій із чинниками навчального середовища, покликаних забезпечити як особистісне зростання, так і формування в них психологічних і змістовних новоутворень, що складають різноманітні аспекти концептуальної моделі творчого професіонала. При цьому передбачається два основних моменти.

По-перше, розуміння і прийняття людиною позиції продуктивної взаємодії з навчальним середовищем, насамперед, за рахунок використання умов, впливів і можливостей, наданих цим середовищем для своєї самоактуалізації і самореалізації. Це забезпечує найбільш повна і гармонійна взаємодія зовнішніх умов і суб'єктивних особливостей. Формування і прийняття студентом такої позиції можливе в результаті не випадкового, а творчого самовизначення, при якому він установлює ступінь відповідності (невідповідності) особистісних передумов до даної діяльності і конкретної фахової діяльності, глибини розуміння й осмислення змісту існуючих нормативних уявлень освоюваної професії.

По-друге, побудова студентом суб'єктивно прийнятних моделей про-

фесіонала і вибір індивідуально-адекватних засобів і стратегій оволодіння ними. Оволодіння студентом особистісним, предметним і операціональним змістом також безпосередньо пов'язане з реалізацією ним процесів особистісного і фахового самовизначення.

Література:

1. Выготский Л.С. Психология. – М.: ЭКСМО–Пресс, 2000. – 1008 с.
2. Психологія: Підручник/ Ю.Л. Трофімов, В.В. Рибалка, П.А. Гончарук та ін.; за ред. Ю.Л. Трофімова. – К.: Либідь, 1999. – 558 с.
3. Морзе Н.В. Методика навчання інформатики. Частина 1. Загальні питання методики навчання інформатики. – К.: Навчальна книга, 2003. – 189 с.
4. Психология и педагогика. Учебн. пособие / Под ред. К.А. Абульхановой, Н.В. Васиной, Л.Г. Лаптева, В.А. Слостенина. – М.: Совершенство, 1998. – 320 с.

МЕТОД ПРОЕКТИРОВАНИЯ АЛГОРИТМОВ ДЛЯ ДИАГНОСТИРОВАНИЯ МИКРОСХЕМ ПАМЯТИ

И.А. Колпаков, В.Г. Рябцев

г. Черкассы, Черкасский государственный технологический университет
volodja18@ukr.net

Для оценки работоспособности микросхем памяти необходимо осуществлять тестовое диагностирование на реальной рабочей частоте. Однако максимальная частота устройств тестового диагностирования ограничена быстродействием блоков формирования тестовых воздействий и обработки считанных реакций. Значительного повышения быстродействия диагностических устройств можно достичь за счет распараллеливания микроопераций формирования кодов адреса, данных и обработки считанных реакций мультипроцессорным устройством тестового диагностирования [1]. Однако усложнение структуры диагностических устройств увеличивает трудоемкость проектирования алгоритмов и программ тестов, так как необходимо управлять одновременной работой нескольких групп формирователей адреса и данных.

Схемы алгоритмов удобно представлять графами с нагруженными дугами [2]. Вершинами таких графов служат состояния схемы $A = \{a^{(0)}, a_1, a_2, \dots, a_{m-1}, a^{(1)}\}$, где $a^{(0)}$ – начальное состояние, $a^{(1)}$ – заключительное состояние. Если осуществляется смена состояния $a_i \xrightarrow{u/q} a_{i+1}$, то вершины a_i и a_{i+1} соединяются дугой, и эта дуга нагружается парой u/q , отображающей условие перехода u и выполняемую микрооперацию q .

Для описания параллельных микроопераций, выполняемых в тестах диагностирования микросхем памяти, предлагается использовать векторную форму записи состояний схем алгоритмов:

$$a \xrightarrow{U/Q} a'.$$

В [3] подробно изложен метод распараллеливания микроопераций в алгоритмах тестов диагностирования быстродействующих микросхем оперативной памяти, основанной на преобразованиях матриц инцидентности графов адресных переходов.

При выбранном коэффициенте k распараллеливания операций диагностическим устройством данная запись принимает вид:

$$a \left[\begin{array}{c|c} x & q_0 \\ x & q_1 \\ \vdots & \vdots \\ u_{k-1} & q_{k-1} \end{array} \right] \rightarrow a',$$

где $U = \begin{bmatrix} x \\ x \\ \vdots \\ u_{k-1} \end{bmatrix}$ – вектор-предикат базовых условий переходов состояния

схемы алгоритма, применяемый в текущий момент времени;

x – безразличное состояние условия перехода;

u_{k-1} – используемое условие перехода, формируемое $(k-1)$ -ым формирователем;

$Q = \begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_{k-1} \end{bmatrix}$ – вектор-столбец базовых операторов, которые выполняются

соответственно 0, 1, ..., $(k-1)$ -ым формирователями кодов адреса и данных.

Условия переходов, отмеченные символом “ x ”, в реализации ветвления программы не участвуют.

Такая особенность в записи переходов обусловлена тем, что обращение к конечной запоминающей ячейке с координатами адреса ($x = nx$) и ($y = ny$) или к начальной ячейке с координатами адреса ($x = gx$) и ($y = gy$) обычно выполняется только при помощи только одного адресного формирователя.

Рассмотрим пример синтеза алгоритма теста галоп по строке, во время выполнения которого осуществляются операции попарного считывания ячеек, расположенных на одной строке. U - Y -схема алгоритма теста галоп по строке для микросхемы емкостью 256 байт приведена на рис. 1.

Для подготовки адресного конвейера устройства, имеющего 4 операционных процессора, необходимо выполнить следующие операции, которые можно представить в виде векторов-операторов:

$$a^{(0)} \xrightarrow{\begin{bmatrix} 0>t, gx>x, gy>y \\ 0>t, gx>x, gy>y \\ 0>t, gx>x, gy>y \\ 0>t, gx>x, gy>y \end{bmatrix}} a_1, a_1 \xrightarrow{\begin{bmatrix} x>x \\ x+1>x \\ x+2>x \\ x+3>x \end{bmatrix}} a_2.$$

Подсчет количества строк в матрице запоминающих элементов обеспечивается программным счетчиком q , в который записывается следующие значение $q := \sqrt{n} - 1$. Запись инверсных данных в смежные строки обеспечивается операторами Q_2 и Q_3 :

$$\mathcal{Q}_2 = \begin{bmatrix} w, x + 4 > x \\ w, x + 4 > x \\ w, x + 4 > x \\ w, x + 4 > x \end{bmatrix}, \mathcal{Q}_3 = \begin{bmatrix} w, \text{nott} > t, x + 4 > x, y + 1 > y^* \\ w, \text{nott} > t, x + 4 > x, y + 1 > y^* \\ w, \text{nott} > t, x + 4 > x, y + 1 > y^* \\ w, \text{nott} > t, x + 4 > x, y + 1 > y^* \end{bmatrix}.$$

В качестве условия перехода U_2 используется признак неравенства нулю индексного регистра i , в который предварительно заносится следующее значение $i := (\sqrt{n} - 4)/4$.

Переход из состояния a_3 в состояние a_2 осуществляется по вектору-предикату U_3 :

$$U_3 = \begin{bmatrix} x \\ x \\ x \\ (x \neq nx) \vee (y \neq ny) \end{bmatrix}.$$

Попарное считывание ячеек, расположенных в одной строке осуществляется при помощи оператора \mathcal{Q}_7 :

$$\mathcal{Q}_7 = \begin{bmatrix} r, a \\ r, a, x + 2 > x \\ r, a \\ r, a, x + 2 > x \end{bmatrix}.$$

Все ячейки в строке могут быть базовыми и использоваться для реализации всех возможных переходов в пределах выбранной строки. Смена номера базовой ячейки обеспечивается использованием базового условия и операторов:

$$[U_8 / \mathcal{Q}_8] = [h \neq 0] \begin{bmatrix} R, A, x + 1 \rangle x \\ R, A, x + 3 \rangle x \\ R, A, x + 1 \rangle x \\ w, x + 3 \rangle x \end{bmatrix}.$$

Изменение номера диагностируемой строки обеспечивают базовое условие U_9 и операторы \mathcal{Q}_9 :

$$[U_9 / \mathcal{Q}_9] = [q \neq 0] \begin{bmatrix} \text{nott} \rangle t, y + 1 > y \\ \text{nott} \rangle t, y + 1 > y \\ \text{nott} \rangle t, y + 1 > y \\ \text{nott} \rangle t, y + 1 > y \end{bmatrix}.$$

Проектирование и отладку программы теста удобно осуществлять при помощи интерпретирующей системы Prover [3]. При помощи интерпретирующей системы Prover можно осуществлять отладку программ тестов для

полупроводниковой памяти емкостью 256 байт. Дальнейшее увеличение емкости диагностируемой памяти при отладке программ тестов не целесообразно, так как при этом возрастает время интерпретации программы теста.

Автоматизированное формирование программ тестов для диагностирования запоминающих устройств заданной емкости можно осуществить заменой в тестах, которые хранятся в библиотеке, тех участков программ, которые связаны с организацией циклов проверки ячеек памяти. Такая замена дает возможность организовать циклические операции для диагностирования микросхем памяти требуемой емкости.

Литература:

1. Мельников А.В., Рябцев В.Г. Контроль модулей памяти компьютеров. – К.: “Корнійчук”, 2001. – 172 с.
2. Капитонова Ю.В., Летичевский А.А. Математическая теория проектирования вычислительных систем. – М.: Наука, 1988. – 296 с.
3. Проектирование и диагностика компьютерных систем и сетей / М.Ф. Бондаренко, Г.Ф. Кривуля, В.Г. Рябцев, С.А. Фрадков, В.И. Хаханов. – К.: НМЦВО, 2000. – 306 с.

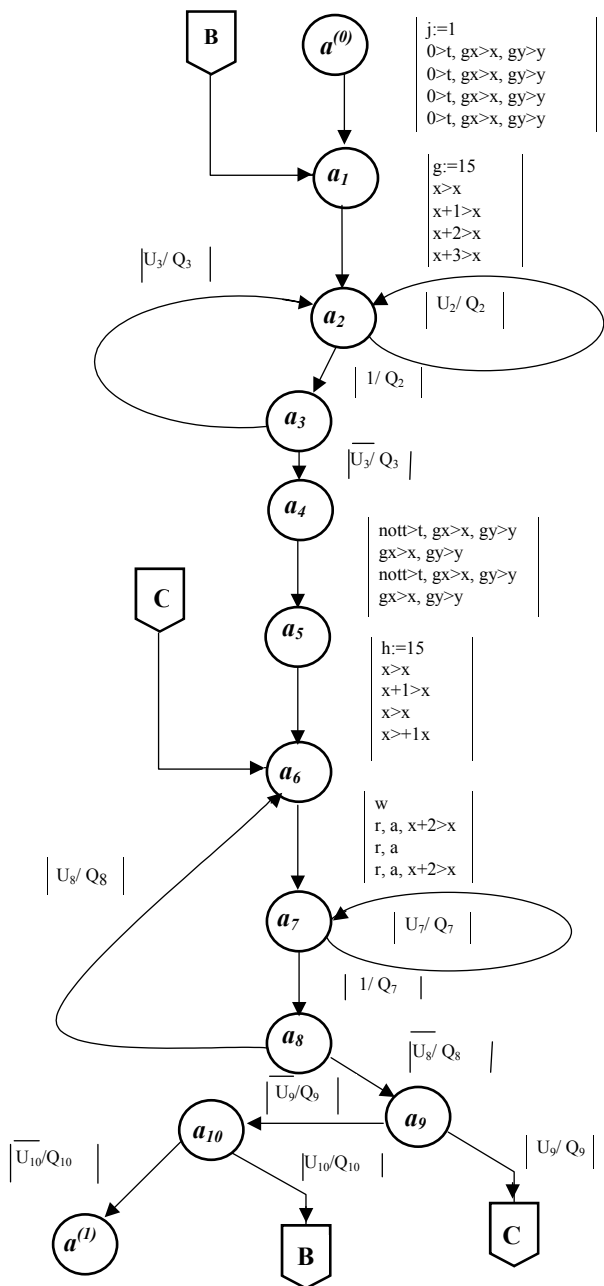


Рис. 1. U-Y-схема алгоритма теста галоп по строке

О ПОТЕНЦИАЛЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И МАТЕМАТИЧЕСКОМ ОБРАЗОВАНИИ СПЕЦИАЛИСТА ПО ЭКОНОМИЧЕСКОЙ ИНФОРМАТИКЕ

В.Н. Комличенко¹, С.А. Поттосина¹, Н.А. Кириенко²

¹ Беларусь, г. Минск, Белорусский государственный университет
информатики и радиоэлектроники

² Беларусь, г. Минск, Объединенный институт проблем информатики
Национальной академии наук Беларуси
kir@newman.bas-net.by

В связи со стремительным развитием информационных технологий на смену индустриальной экономике приходит информационная экономика. В таких условиях возросла потребность в специалистах экономического профиля, обладающих высоким уровнем знаний в области информационных систем и технологий, т.е. владеющих как информатикой, так и экономическими знаниями. Стандарт специальности «Информационные системы и технологии (в экономике)» определяет базовые требования к уровню и содержанию знаний, умений и навыков выпускников, получающих квалификацию «инженер-программист-экономист». Данная специальность открыта на экономическом факультете БГУИР, подготовка по ней осуществляется кафедрой экономической информатики.

После окончания вуза выпускник может оказаться на предприятиях, находящихся на любой стадии компьютеризации, и должен в него вписаться. Деятельность предприятий самым тесным образом связана с управлением производственными процессами и своевременным принятием решений, позволяющих устанавливать равновесие между разнообразными потребностями и целями компаний. Система хозяйствования, с которой мы постепенно расстаемся, имела дело исключительно с материальными ресурсами. В новой экономике покупается и продается «концентрированное знание», вследствие чего знания и информация становятся первичным сырьем экономики и ее важнейшим ресурсом. Выделяются три типа знаний: знания предметной области, знания людей, знания процессов. Преобразование данных в знания является одним из аспектов управления знаниями. Управление знаниями – это интегрированный подход к созданию, сбору, доступу, обработке и использованию информационных ресурсов организации (структурированные базы данных, текстовая информация или документы). Управление знаниями – одно из ключевых понятий, определяющих стратегическую цель разработки и применения информационных технологий. Формирование информационного общества и информационной экономики характеризуется созданием и накоплением информационных ресурсов и информационных знаний как нового вида стратегических ресурсов. Последнее требует разработки документационных систем управления и формирование на их

основе интеллектуальных технологий принятия решений.

Современные достижения и развитие информационных и коммуникационных технологий привели к созданию глобальной электронной среды, так называемой «сетевой экономики». Для формирования и расширения масштабов сетевой экономики необходимо развитие и распространение Интернет-технологий. Наиболее перспективной формой электронной коммерции на сегодня являются Интернет-магазины, электронные аукционы, сделки по недвижимости. Отдельным направлением электронного бизнеса является электронный маркетинг, электронные системы обслуживания клиентов, когда предметом торговли становятся энергетические и коммуникационные ресурсы. Заметно развитие сетевой экономики в сфере финансовых рынков, электронные платежные системы для бизнес-операций, использующие протоколы *SET*, которые обеспечивают надежную защиту конфиденциальной информации и позволяют осуществить аутентификацию всех участников сделки. Широкое распространение получили дистанционные трудовые отношения, позволяющие снижать остроту транспортных проблем, работать практически в любой стране мира, расширяют возможности трудоустройства для инвалидов. Наконец, виртуальные кафедры, университеты, фирмы, предприятия становятся объективной реальностью. На решение этих задач, прежде всего, и направлены передовые информационные технологии.

Рассмотрим более подробно структуру учебного плана специальности, позволяющего приобрести знания и умения в области информационных технологий. Потенциал информационных технологий, приобретаемый выпускниками, можно условно разделить на три блока: блок 1 – общенаучных и общеобразовательных дисциплин, блок 2 – специальных дисциплин, блок 3 – дисциплин направления.

В блок 1 входят такие дисциплины как «Основы информатики и программирование» (семестр 1, 2), «Основы и лингвистическое обеспечение баз данных» (семестр 4), «Компьютерные сети» (семестр 5). Блок 2 наполняют такие дисциплины, как «Объектно-ориентированное проектирование и программирование» (с разделами «Визуальные средства разработки приложений», «Языки программирования для разработки сетевых приложений») (семестры 3-6, курсовой проект), «Прикладные системы обработки данных» (семестр 3), «Системный анализ и проектирование систем (с разделом «Проектирование баз данных и знаний») (семестр 6-7, курсовой проект), «Операционные системы» (семестр 5), «Криптография и охрана коммерческой информации» (семестр 9). Блок 3 содержит такие дисциплины, как «Сетевые информационные технологии» (с разделами «Технологии WWW», «Web-дизайн», «Разработка приложений для WWW») (семестр 8), «Проектирование распределенных информационных систем» (с разделом «Корпоративная информационная система») (семестр 8), «Интеллектуальные информационные системы в экономике» (семестр 8), «Современные техноло-

гии обработки экономической информации» (с разделом «Технологии автоматизации делопроизводства») (семестры 8-9, курсовой проект).

Базовыми языками программирования выбраны *C*, *C++*, *JAVA*. На курсовом проектировании студенты усваивают *CASE*-технологии (для функционального и информационного моделирования). Для описания и разработки образовательных ресурсов предлагается спецификация *IMS*, языки разметки *HTML*, *XML*.

Одним из важнейших курсов блока 2 является курс «Визуальные средства разработки приложений». Цель изучения данной дисциплины – овладение знаниями и навыками использования языка *C++*, библиотеки классов *MFC* и среды *Microsoft Visual Studio* для разработки *Windows*-приложений, применяемых при автоматизации решения экономических задач.

Важность этого курса объясняется тем, что он является одним из первых курсов на пути практической работы с классами, основополагающими понятиями объектно-ориентированного программирования. Большое значение приобретает возможность за короткие сроки (благодаря использованию библиотеки *MFC*) разработать приложение, совмещающее сложную обработку данных с богатыми средствами их отображения на экране. *Visual C++* дает возможность использования технологии клиент-сервер, программирования для *Internet*, автоматизированной разработки справочной системы.

Программа курса «Визуальные средства разработки приложений» (семестр 5, 34 часа лекций, 34 часа лабораторных работ (л.р.), 17 часов практических занятий (п.з.)) включает следующие основные разделы: *технология проектирования приложений под Windows; обзор среды Microsoft Visual Studio; программирование Windows-приложений на языке C++ и библиотека MFC; архитектура Document-View, SDI- и MDI-приложения; сохранение-восстановление объектов и работа с файлами; диалоги, классы окон, элементы управления; доступ к базам данных; современные методы программирования; введение в технологии OLE и ActiveX.*

Лекционный материал закрепляется практической работой на компьютере. Студентам предлагается выполнить 8 лабораторных работ и 3 индивидуальные задания. Уровень индивидуальных заданий рассчитан на среднего студента. Для продвинутых студентов есть возможность усложнить задания введением новых визуальных средств в интерфейс приложения. Разработанные индивидуальные задания служат основой для проверки практических знаний студентов на экзамене.

Примерный перечень лабораторных работ следующий: создание *SDI*-приложения; рисование в программе, использующей *MFC*; хранение данных; диалоговые окна и элементы управления; стандартные элементы управления; разработка приложения управления базой данных; программирование доступа к записям базы данных; программирование работы с несколькими таблицами базы данных.

Основные трудности, которые возникают у студентов при изучении

курса, связаны с освоением и восприятием довольно большого количества новых для них терминов и понятий стандартов программирования под Windows. Среди наиболее важных тем следует отметить изучение многообразия ресурсов, с которыми может работать приложение, выработка навыков обработки сообщений в различных классах приложения, рассмотрению файлов и классов, из которых состоит проект. Работе с базами данных следует уделять особое внимание, так как возможно использование этих технологий *Visual Studio* для работы над дипломным проектом.

В процессе реализации специальности на кафедре экономической информатики читаются следующие дисциплины экономико-математического профиля: основы дискретной математики и теории алгоритмов, экономико-математические модели и методы (блок 1), эконометрика (блок 2), исследование операций в экономике, математика рынка ценных бумаг (блок 3). Ниже приведено краткое содержание этих дисциплин, ориентированных на выпускников технического университета.

Программа курса «Основы дискретной математики и теории алгоритмов» (семестр 4, 34 часа лекций, 17 часов п.з.): *Множества. Отношения. Алгебры. Элементы математической логики. Элементы логики предикатов. Элементы теории графов. Задачи комбинаторики и методы комбинаторного поиска. Элементы теории алгоритмов и автоматов.*

Изучение всех разделов данного курса сопровождается практическими занятиями и выполнением студентами индивидуальных заданий, среди которых есть задачи на разработку и реализацию алгоритмов комбинаторного поиска, алгоритмов анализа графов с оценкой их вычислительной сложности.

Программа курса «Экономико-математические модели и методы» (семестр 6, 34 часа лекций, 34 часа п.з.) состоит из следующих разделов: *Линейные балансовые модели. Модели сетевого планирования и управления. Модели линейного программирования. Модели простой и множественной линейной регрессии. Модели управления запасами. Модели многомерного статистического анализа.*

При закреплении материала студенты выполняют систему индивидуальных заданий, а практические занятия проводятся в компьютерном классе. На кафедре подготовлены учебное пособие по дисциплине и лабораторный практикум с использованием электронных таблиц «Excel».

Курс «Эконометрика» (семестр 7, 34 часа лекций, 17 часов п.з., 17 часов л.р.) содержит тот минимум знаний по эконометрике, которым должен владеть каждый инженер-экономист:

Эконометрические модели (множественная регрессия с переменными параметрами, системы одновременных уравнений и их идентификация, оценивание регрессионных моделей в условиях мультиколлинеарности, гетероскедастичности и автокорреляции). *Эконометрические методы* (обобщенный метод наименьших квадратов, двухшаговый метод наименьших

квадратов, косвенный и двойственный методы наименьших квадратов). *Эконометрические приложения* (производственные функции, функции инвестиций, функции спроса, проблемы оценивания и агрегирования). *Анализ и прогнозирование временных рядов* (авторегрессионные модели скользящей средней, авторегрессионные интегрированные модели скользящей средней, адаптивные модели краткосрочного прогнозирования, анализ временных рядов при случайных моментах измерений).

Изучение эконометрики дополняется работой на практических занятиях, лабораторными работами на компьютере с привлечением современных информационных технологий статистической обработки данных, в частности, пакета прикладных программ «Статистика», выполнением системы индивидуальных заданий.

Программа курса «Исследование операций в экономике» (семестр 8, 48 часов лекций, 17 часов п.з.) состоит из четырех разделов: *Детерминированные оптимизационные модели исследования операций* (задачи нелинейного программирования). *Игровые модели исследования операций* (игры с нулевой и ненулевой суммой, кооперативные игры с побочными платежами). *Оптимизационные задачи на сетях и графах* (задачи о покрывающих множествах, задачи о кратчайших цепях, достижимость и исследование структуры организаций, задачи о размещении центров и медиан, потоки в сетях). *Модели массового обслуживания*.

Достаточное внимание в лекциях и на практических занятиях уделяется прямым приложениям математической теории игр для анализа микроэкономических проблем. В частности, применение теории игр для анализа рыночного равновесия как кооперативной игры многих лиц, применение статистических функций решений в сфере деятельности промышленных и торговых предприятий, принятие макроэкономических решений в условиях неопределенности и риска. Рассматриваются вопросы принятия решений при нестохастической неопределенности для многокритериальной функции полезности (метод анализа иерархий Саати), а также возможности использования в задачах экономики аппарата теории нечетких множеств.

В разделе «Оптимизационные задачи на сетях и графах» особое внимание уделяется задачам и алгоритмам поиска кратчайших путей и близких к ним, таких как наиболее надежные пути, пути с максимальной пропускной способностью, пути с «узкими» местами, пути с усилением. Все эти задачи иллюстрируются приложениями из экономической деятельности. Так, задачу о финансисте, который наилучшим образом распределяет во времени вложение своего капитала в различные активы, можно рассматривать как задачу о путях с усилением в некотором графе.

Для исследования структуры руководства или влияний некоторой организации полезно знание некоторых фундаментальных понятий, касающихся достижимости и связности графов, а также алгоритмов для определения базы и антибазы графа, графа конденсаций, сильной и ограниченной

базы, сильной компоненты графа. С задачей о покрытии булевой матрицы тесно связана задача построения диагностического теста, решение которой позволяет найти минимальное подмножество внешних признаков, позволяющее диагностировать (опознавать) некоторое явление (процесс), о котором мы можем только догадываться. Организация проверки значений некоторого признака связана с определенными затратами, отсюда и появляется необходимость в минимизации числа признаков, образующих тест.

Данный раздел сопровождается выполнением студентами индивидуальных заданий. В каждом задании предлагается определенная экономическая, организационная или управленческая задача, решение которой необходимо свести к решению некоторой оптимизационной задачи на графах, предложить алгоритм решения и реализовать его в виде программы. При этом используются такие методы анализа графа как методы поиска в глубину и ширину, комбинаторные алгоритмы, методы ветвей и границ

В разделе «Модели массового обслуживания» достаточное внимание уделяется марковским случайным процессам. Марковские модели представлены цепями Маркова с дискретным и непрерывным временем, марковскими цепями с доходами и переоценкой доходов, марковскими моделями систем массового обслуживания (СМО). Это позволяет решать задачи, связанные с марковскими моделями принятия решений и расчетом характеристик функционирования простейших СМО. Демонстрируется применение марковских цепей в качестве вероятностных моделей различных финансово-экономических ситуаций, а также возможность с помощью потоков Эрланга сводить немарковские процессы к марковским. В настоящее время разрабатывается цикл лабораторных работ по курсу, связанный с имитационным моделированием процессов, протекающих в финансово-экономической сфере. Данный раздел также сопровождается выполнением студентами индивидуальных заданий.

ЗМІСТ КУРСУ «НОВІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ» ДЛЯ СТУДЕНТІВ ФАКУЛЬТЕТУ ІНОЗЕМНИХ МОВ

С.В. Кондратенко

м. Кривий Ріг, Криворізький державний педагогічний університет
ksv_ipm@mail.ru

Широке використання комп'ютерів у різних сферах діяльності потребує подальшого підвищення якості підготовки спеціалістів, що використовують персональний комп'ютер (ПК) як інструмент у своїй основній роботі. Нажаль, непрофесійні користувачі ще й досі складають основну частину у загальній кількості користувачів ПК. Не становлять виключення і випускники педагогічних вузів. Виявляється, що студенти не отримують достатньо практичних навичок з використання комп'ютерів для розв'язання конкретних прикладних задач. Внаслідок цього виникає певний психологічний бар'єр між непрофесійними користувачами і комп'ютерами.

«Нові інформаційні технології» (НІТ) є нормативною дисципліною природничо-наукового циклу для спеціальностей гуманітарного напрямку педагогічних ВНЗ. Основна мета даного курсу полягає у формуванні у студентів знань, умінь та практичних навичок, необхідних для роботи з сучасними високопродуктивними інформаційними технологіями, що забезпечить можливість здійснення комп'ютерної обробки інформації при написанні науково-дослідних робіт, а також при проведенні статистичного аналізу даних, отриманих в результаті наукового експерименту.

Акцентуючи увагу на формуванні спеціальних умінь роботи з сучасними технологіями і системами для кваліфікованого проведення наукових досліджень, поза увагою залишається фахова спрямованість навчання студентів педагогічного ВНЗ.

Саме тому, формування спеціальних знань, умінь та навичок роботи з апаратними і програмними засобами НІТ, як на те націлює основна мета курсу, ми здійснювали через навчальні завдання, що демонстрували можливість використання отриманих знань і сформованих навичок у професійній діяльності вчителя сучасної школи і класного керівника. Адже в процесі навчання на основі НІТ у школярів з'являється стійкий інтерес до учбової та пізнавальної діяльності, формуються пізнавальні мотиви, потреба в самоосвіті, потреба у самовизначенні з усвідомленням особистої відповідальності за результати навчання, потреба в колективній роботі, спрямованій на отримання спільного результату [2].

Вивчення курсу «Нові інформаційні технології» у нашому навчальному закладі триває вже не один рік. За цей час склався і затвердився основний зміст його розділів. Викладаючи даний курс для студентів другого курсу факультету іноземних мов, ми не вносили суттєвих змін до цього змісту.

Так, на лекціях (4 години) проводимо короткий огляд апаратних і про-

грамних засобів нових інформаційних технологій за таким планом:

1. Комп'ютер як інформаційно-обчислювальна система. Принцип функціонування ЕОМ. Основні складові апаратної частини комп'ютера.

2. Периферійні пристрої, їх призначення та принципи роботи.

3. Накопичувачі на зовнішніх носіях (дискети, оптичні диски, DVD-диски, флеш-карти). Основні характеристики: ємність, надійність, доступність у застосуванні.

4. Локальні та глобальні мережі. Internet-технології.

5. Програмне забезпечення комп'ютера: класифікація, коротка характеристика його видів.

Розглядаючи питання програмного забезпечення спеціального призначення, ми провели короткий порівняльний аналіз поширених програм-перекладачів (Сократ, Promt, MagicGoody), програм-тестів і тренажерів з іноземних мов для дітей різного віку, відмітили їхні переваги і недоліки.

Зміст лабораторного практикуму, розрахованого на 48 аудиторних годин, подано у вигляді таблиці:

№	Теми лабораторних робіт	Кількість годин
1.	Апаратне забезпечення комп'ютера	2
2.	Операційна система Windows	4
	Робота з вікнами. Панель задач. Головне меню. Робота з основними об'єктами операційної системи Windows.	2 2
3.	Текстовий процесор Word	14
	Word. Введення тексту та редагування тексту.	2
	Word. Форматування тексту.	2
	Word. Робота з декількома документами	4
	Word. Робота з таблицями.	2
	Графічні об'єкти в Word.	2
	Залікове завдання	2
4.	Електронні таблиці Microsoft Excel	6
	Структура таблиці та основні типи даних. Введення даних до таблиці. Формули і статистичні функції.	2
	Логічні функції. Побудова діаграм.	2
	Залікове завдання	2
5.	Система керування базами даних Microsoft Access	8
	Створення та заповнення таблиць бази даних.	2
	Схема даних. Робота з формами.	2
	Робота з фільтрами та запитам.	2
	Залікове завдання	2
6.	Редактор презентацій Microsoft PowerPoint	4
	Створення презентації за зразком.	2
	Створення власної презентації	2

№	Теми лабораторних робіт	Кількість годин
7.	Розробка Web-сторінок засобами мови HTML	8
	Форматування тексту на сторінці. Списки.	2
	Робота з таблицями.	2
	Графічні об'єкти. Гіперпосилання.	2
	Залікове завдання	2
8.	Архіватор WinRar	2
<i>Разом:</i>		48

Кожен розділ лабораторного практикуму містив набір стандартних навчальних завдань-прикладів, які були оформлені у вигляді лабораторних робіт на зразок того, як це пропонує Я.М. Глинський у своєму лабораторному практикумі [1]: кожне нове завдання супроводжувалося чіткими коментарями щодо його виконання. Цілком очевидно, що організована у такий спосіб діяльність студентів на лабораторних заняттях є прямою реалізацією індивідуального підходу у навчанні. Деякі лабораторні роботи курсу були запропоновані в авторському варіанті, але від більшої частини робіт, запропонованих Я.М. Глинським, ми відмовилися, через їхній переважно економічний характер, і навчали на задачах шкільного курсу.

Для здійснення поточного контролю вивчення того чи іншого розділу завершувалося виконанням підсумкового комплексного завдання, під час якого студенти мали продемонструвати практичні навички роботи з вивчених програм.

Так в якості підсумкового завдання вивчення можливостей текстового процесора Word студентам було запропоновано розробити електронний звіт по виконанню лабораторних робіт з теми. Задача студентів – об'єднати в один документ методичні розробки лабораторних робіт та результати їх власного виконання і здійснити його кваліфіковане форматування.

Для перевірки отриманих знань з теми «Табличний процесор Excel» студентам було запропоновано розробити «Табель успішності класу» – електронний звіт класного керівника.

Продовжуючи ідею оптимізації діяльності класного керівника за рахунок використання комп'ютерної техніки, підсумковим завданням з теми «СУБД. Microsoft Access» було завдання розширити навчальну базу даних «Школа» так, щоб вона стала електронним довідником завуча чи директора школи.

На вивчення можливостей редактора презентацій Microsoft PowerPoint було виділено лише чотири години, але цього було досить, щоб викликати інтерес студентів до використання презентацій у подальшій професійній діяльності та науково-дослідній роботі. Більшість студентів захопилася ідеєю розробки демонстрації до захисту кваліфікаційної роботи. Підсумковим завданням даного розділу стала розробка презентації-фотоекскурсії по країнах світу (за матеріалами електронної енциклопедії).

Для закріплення умінь по розробці Web-сторінок засобами мови HTML, студентам пропонувалося розробити прототип сайту групи.

Поза увагою на лабораторних заняттях залишилися інформаційні ресурси мережі Internet, пошукові системи та огляд можливостей електронного листування. Через обмеженість доступу до мережі Internet в університеті ці питання були винесені на самостійне опрацювання. Студенти отримали по два завдання:

1. Користуючись засобами однієї з пошукових систем (Rambler.ru, Yahoo.com, Yandex.com, Altavista.ru, Google.ru), знайти відповіді на поставлені запитання [3, с. 369-371], але при цьому обов'язково підтвердити її посиленням на Інтернет-ресурси, де була знайдена відповідь.

2. На одному із безкоштовних серверів (наприклад, mail.ru) зареєструвати поштову скриньку. Відіслати звіт про виконання першого завдання за адресою викладача.

Завдання з розробки власних презентацій і самостійне набуття навичок роботи в Інтернет викликали найбільшу зацікавленість.

Питання курсу, запропоновані до вивчення, дублюють теми шкільного курсу “Основ інформатики та обчислювальної техніки”. Відмінність вузівського курсу від шкільного полягає у його орієнтації на засвоєння техніки розв'язання професійних задач за допомогою ЕОМ і найсучасніших інформаційних і телекомунікаційних технологій.

Зміст курсу “Нові інформаційні технології” потребує періодичного оновлення й удосконалення відповідно до розвитку інформаційних технологій, адже кожен науковець і вчитель сучасної школи повинен володіти навичками роботи з провідними світовими інформаційними технологіями.

Література:

1. Глинський Я.М. Практикум з інформатики: Навч. посіб., 7-ме видання. – Львів: Деол, 2004. – 224 с.
2. Головань М.С. Вплив засобів нових інформаційних технологій на методичну систему навчання математики // Педагогіка та психологія: збірник наукових праць Харківського державного педагогічного університету. Вип. 11-14. – 2000. – Вип.13. – С. 11-14.
3. Сафронов И.К. Задачник-практикум по информатике. – СПб.: БХВ-Петербург, 2002. – 432 с.: ил.

МОЖЛИВОСТІ MACROMEDIA FLASH В АСПЕКТІ СУЧАСНИХ КОНЦЕПЦІЙ ПОБУДОВИ ІНТЕРФЕЙСУ КОРИСТУВАЧА

В.В. Кравченко

м. Кривий Ріг, Криворізький державний педагогічний університет

Взаємодія об'єктів довільної природи заснована перш за все на обміні інформацією. Назвемо цей процес *спілкуванням*. Ефективність спілкування визначається багатьма чинниками, серед яких основними є здатність сторін до передавання та сприймання інформації. Здатність об'єктів до передавання інформації визначається їх умінням здійснювати оптимальний вибір засобів її передавання, логічно будувати цей процес, а також вмінням об'єкту зосереджувати увагу респондента на процесі її сприйняття. У людському спілкуванні ця здатність визначається якостями особистості.

Швидкий розвиток комп'ютерних технологій призвів до виникнення зовсім неочікуваного застосування комп'ютерів у різних галузях діяльності. Проникнення комп'ютерної техніки в різні сфери людської діяльності стало причиною того, що комп'ютер виявився невід'ємним учасником процесу обміну інформацією, зайнявши при цьому проміжну ланку серед об'єктів, що приймають участь в обміні інформацією. Здатність до передавання інформації комп'ютером визначається інтерфейсом. Під інтерфейсом у даному випадку слід розуміти не тільки зображення на екрані монітора, а й способи взаємодії комп'ютерної системи з користувачем [2].

Головне завдання будь-якого інтерфейсу – полегшити користування об'єктом, якому він належить. Щоб привернути увагу до програми, однієї тільки гарної ідеї стосовно її призначення мало, необхідно її втілення на професійному рівні. Нажаль, на сьогоднішній день більшість комп'ютерного програмного забезпечення не зовсім відповідає цій вимозі через відсутність природності елементів інтерфейсу. Наприклад, такий інтерфейсний елемент, як скролінг знаходиться в протиріччі з одним із основних принципів психології сприйняття: у людини є тільки одна точка активної уваги. Використання скролінгу ж передбачає зосередження уваги не тільки на зображенні, яке прокручується, а ще й на самому елементі скролінгу, що призводить до великої кількості операцій неправильної взаємодії з даним об'єктом. Таких неприродних об'єктів в елементах програмного інтерфейсу дуже багато. Проте відсутність можливості вибору призводить до повного звикання щодо їх використання.

На жаль, багато програмістів демонструють повне безсилля в створенні нормальних користувацьких інтерфейсів для своїх програм. Часто вони цим навіть пишаються – мовляв, інтерфейси з різними новинками нікому не потрібні, головне в програмі – функціональність. Причина подібного відношення – цілковита неосвіченість в області розробки інтерфейсів. Звичайно, краса програми, як і літака, – не в зайвих прикрасах. Літак, обвішаний бан-

тиками, з художнім ліпленням і різьбленими крилами червоного дерева не пролетить і ста метрів. Його краса обумовлюється доцільністю всієї конструкції. Ще в більшій мірі це стосується програм. Вважається гарним тоном мати охайно розташовані кнопки однакового розміру. Зокрема, кнопки різних діалогів у стандартному віконному інтерфейсі можна сприймати як сигнали до натискання. Але ці сигнали доволі слабкі, оскільки мають однаковий вигляд і відрізняються лише текстом. Виключенням є кнопка Ок, яка сприймається користувачем як мнемонічний знак. Як підтвердження – більшість регіональних версій будь-якої програми не містять переклад тексту цієї кнопки на відповідну мову. Та різноманітність ще не означає естетичного порушення. Прикладом є пульти дистанційного управління, або skin-технологія, використана в мультимедійному програвачу WinAmp.

Перелічимо ще декілька вимог, які можна поставити до розробника прикладних програмних пакетів. Для схожих функцій бажано використовувати і схожі форми, інакше програма буде для користувача видаватись новою. Якщо користувач звик до чого-небудь, він швидше навчиться працювати і буде одержувати більше задоволення від роботи з вашою програмою, тому що зможе використовувати раніше надбані навички. Базове запозичення – це використання стандартних елементів, загальних для всіх прикладних програм. Наприклад, у MS Office такими елементами є меню, списки, кнопки і т.п. Більш тонке запозичення – це запозичення популярної метафори. Тільки робити це треба обережно. У якості прикладу можна навести інтерфейси комунікаційної програми Trio Communication і записної книжки Lotus Organizer. Trio нагадує справжній телефон, а Organizer – записну книжку. Тільки чомусь першою програмою користуватися досить складно, а друга – легка і зрозуміла. Чому? Автори Trio переробили всі елементи керування на свій смак. Програма оздоблена настільки, що на засвоєння прийомів роботи з її оригінальним інтерфейсом витрачається маса зусиль. А ось Organizer для стандартних функцій використовує стандартні засоби.

Ніким не заборонено запозичати з будь-яких програм зовнішній вигляд, команди, і деякі (але не всі) вдалі інтерфейсні рішення і т.п. Найкраще, коли інтерфейс вашої програми буде схожий на інтерфейс іншої програми, елементи якої вже стали звичними.

Будь-яка задача повинна розв'язуватись мінімальною кількістю дій; логіка цих дій повинна бути очевидною для користувача. Навіть рух курсору і ока користувача мають бути узгодженими. Метод drag'n'drop – “перетягни-і-залиш” – гарна ілюстрація цього принципу. Це абсолютно природна дія, виконувана одним рухом миші [3].

Найважливіша інформація й елементи керування повинні бути “на очах”, легко доступними, а менш важлива – ховатися де-небудь у меню. Інтерфейс програми повинен бути побудований навколо об'єктів, з якими маніпулює користувач, і відбивати стан поточного об'єкта. Гарний приклад щодо цього – панелі керування в Corel Draw 8.0. Вони постійно змінюються

в залежності від того, з яким об'єктом у даний момент працює користувач.

Користувачеві бажано бачити, чим зараз займається програма, чи до чого призвела його дія. Якщо відбулася помилка, повідомлення про неї має пояснити, що саме відбулося і як це виправити.

Наступна вимога, це використання обмеженої кількості основних кольорів у програмі – не більше трьох, крім відтінків. Гарний приклад – стильний дизайн кнопок у Netscape Communicator 4.0. Слід також враховувати золотий перетин, 1:1,62 – найбільш приємна для ока пропорція.

Відомо, що людина здатна одночасно розподілити увагу не більше, як на семи об'єктах. Відповідно слід групувати сутності в програмі. Пункти меню, закладки, опції на цих закладках і т.п. бажано розташовувати з урахуванням цього правила, тобто не більше семи в групі, в крайньому випадку – дев'яти.

Одним з основних принципів побудови інтерфейсів є використання так званої “метафори”. Прикладом є інтерфейс операційної системи Windows, який побудовано за аналогією з робочим столом. У такий спосіб вирішується декілька проблем: 1) користувачеві легше розуміти та тлумачити інформацію, що відображується на екрані; 2) розуміння дій, які можна виконувати з об'єктами робочого столу, впливає природно з самої “метафори”; 3) користувач отримує відчуття психологічного комфорту, працюючи з чимось знайомим [3].

Проте використання метафор ще не дає змоги говорити про вдало організований інтерфейс, оскільки метафори зовнішнього середовища не завжди можна повністю відтворити у внутрішньому світі комп'ютера. Крім того, звикання до використовуваного раніше інтерфейсу з боку розробників нових програм призводить до створення інтерфейсів, які здаються їм природними, хоча для загального кола користувачів вони не є такими. Як приклад можна навести віконний інтерфейс ОС Windows 3.1, створення якого дозволило розв'язати лише задачу формування системи інтерфейсних елементів, але невизначеним залишився алфавіт взаємодії, вивчення якого дозволило користувачу легко виконувати будь-які дії, а також спосіб позначення окремих елементів та їхніх груп. Означена проблема була краще реалізована в останніх версіях ОС Windows, наприклад, в темах оформлення зовнішнього вигляду робочого столу.

Прикладом концептуально вдалого дизайну інтерфейсу є система дорожніх знаків, яка своїм внесенням не порушує природність середовища, поєднує в собі реалістичні зображення з абстрактними, працює, не вимагаючи від користувачів спеціальної освіти і т.д. Концептуальний дизайн повинен бути заснований на ідеї інтерфейсного середовища. Працюючи з програмою, користувач заглиблюється в середовище інтерфейсу, як турист, що приїхав на сафарі, заглиблюється в середовище дикої природи.

Поняття метафори та середовища тісно пов'язані. Якщо середовище за зовнішнім виглядом та вихідними елементами буде нагадувати користувачу

щось знайоме, то він швидше до нього пристосується. Але не слід фотографічно переносити схожість елементів середовища на елементи інтерфейсу, оскільки програма – це штучне середовище, яке не може відтворити всі елементи фізичного світу [2]. У цьому секрет невдалого поширення багатьох комп'ютерних ігор з гарними графічними можливостями. Але деякі інші ігри, такі як ТЕТРІС, КОСИНКА, а також велика кількість ігор у вигляді Flash-роликів, мають доволі просте й умовне середовище, яке забезпечує комфорт користувачеві і популярність цих ігор. З цього випливає необхідність балансу між інтерактивними можливостями програми і складністю її графічного відображення. Враховуючи таку необхідність, інтерфейс складних програм слід будувати таким чином, щоб складність підвищувалась поступово – користувач повинен крок за кроком засвоювати всі елементи інтерфейсу та можливості програми. Складне зображення на екрані викликає дискомфорт і відсутність бажання витратити години на вивчення властивостей елементів інтерфейсу та можливостей програми. Саме це найбільше відбиває проблему віконного інтерфейсу. Усі його елементи з'являються із самого початку і весь час присутні на екрані. У підсумку місця для змістовної інформації мало, а екран справляє враження засміченого робочого місця. Тому програма повинна надавати можливості до динамічного формування відображення елементів інтерфейсу.

Проте основною проблемою в інтерфейсі є точка синхронізації уваги користувача і активного елемента системи [2]. З одного боку користувач повинен вказати, де і що він хоче змінити, а з іншого – система повинна зосередити його увагу на точці найбільш актуальних змін. У контексті цієї проблеми слід враховувати, що рухомий предмет легше привертає увагу. Так при пошуку курсору на екрані, користувач не оглядає весь екран, а робить декілька рухів мишею.

В усіх всесвітньо відомих центрах з розробки нових інтерфейсів ведуться розробки різних концепцій дизайну інтерфейсів, що спираються на можливості анімації. Перевагою інтерфейсів з елементами анімації є наявність динамічних візуальних сигналів – зміни на екрані з метою надання певної додаткової інформації. Прикладом таких сигналів можна назвати зміну зовнішнього вигляду курсору при зміні стану системи, або “помічник” у прикладних пакетах MS Office. Навіть мигання курсору в текстових редакторах обумовлене необхідністю зосередження уваги користувача на місці введення інформації. Але розв'язуючи проблему для користувача, анімаційний інтерфейс ставить складні проблеми перед програмістами та дизайнерами в плані його реалізації. Після вибору візуального вирішення слід намалювати нерухомі складові інтерфейсу – “фон”. На фоні слід розташувати анімовані елементи взаємодії. І нарешті, найбільш складною проблемою є проектування візуальних переходів між суттєво різними станами системи, дотримуючись при цьому обраного стилю.

На перший погляд здається, що така складність невиправдана – корис-

тувач не помітить зручностей і переваг інтерфейсу, організованого у такий спосіб. Гарний інтерфейс, як гарне взуття – його ніхто не помічає. Головне завдання інтерфейсу не привертання уваги, а забезпечення оптимальних умов та ефективності процесу взаємодії системи з користувачем. Нажаль, слід констатувати, що стандартом став поганий інтерфейс, оскільки він з'являється в більшості випадків сам собою. Програмісти працюють над вдосконаленням алгоритмів отримання, обробки та передавання інформації, приділяючи мінімум уваги процесу передавання цієї інформації кінцевому користувачу. Адже легше написати систему довідки до незручного інтерфейсу, ніж вдосконалити самий інтерфейс. Прикладом є інтерфейс, який найчастіше зустрічається при використанні мережі Internet, створений на основі мови HTML. Альтернативою йому з кожним днем усе більше стає використання для розробки вебсторінок так званої флеш-анімації.

У 1997 році компанія Macromedia придбала в компанії FutureWave невелику графічну програму для Web, що називалася FutureSplash. Це була незвичайна програма, що вражала своїми можливостями генерування компактної векторної графіки для публікації в Web. Завдяки компанії Macromedia згадана програма в результаті певних доробок стала дуже популярною. Модуль вбудованого програвача Flash входить до складу більшості операційних систем і браузерів. Графіка Flash практично всюди використовується в Web, а кількість користувачів Flash продовжує неухильно зростати, дивуючи своїми темпами.

Програма Flash 5 значно розширила інтерактивні можливості і можливості для програмування фільмів Flash. Тепер ці фільми можуть безпосередньо взаємодіяти зі сценаріями і програмами, розташованими на сервері, використовуючи URL-змінні чи XML-формат. Звуки можна імпортувати й експортувати у форматі MP3, що характеризується високою якістю звуку і малим розміром. Безліч незалежних розробників при створенні власних прикладних пакетів враховують можливість збереження даних у кінцевому форматі SWF. Програма Flash вважається основним додатком для створення “крутого” інтерактивного вмісту для публікації в Web, який задовольняє низькій пропускну здатності Internet-з'єднань, [1].

Flash – це не тільки один інструмент, а й демонстраційна програма, редактор для графіки та звуку, а також засіб для анімації і машина для написання сценаріїв. І все це об'єднано в одній програмі.

Однак досвідчені користувачі використовують Flash не тільки для створення анімованих кнопок для HTML-документів Web, а й для створення інтерактивних і привабливих презентацій. Програма Flash 5 істотно вплинула на здатність фільму Flash взаємодіяти з власними інтерактивними елементами, наприклад, відеокліпами.

Flash 5 містить нову інтерактивну мову і синтаксис, що називається ActionScript, спрощуючи цим написання сценаріїв і робить їх більш гнучкими. Мова містить можливості роботи з різними типами даних (змінні, маси-

ви), роботи з функціями, використання Smart Clip, створення форми Flash і завантаження XML, а також використання текстових полів HTML і їхніх властивостей.

Коротко наведемо ще деякі з можливостей програми Flash:

- інтерактивна база даних, яка відсилає і отримує інформацію за допомогою сценаріїв, розмішених на сервері;
- інтеграція фільмів Flash у великий фільм ShockWave Director, який може програвати фільми Quick Time, аудіо в форматі Midi, а також інші формати, які Flash безпосередньо не підтримує;
- підтримка при перегляді оберненого зв'язку з користувачем;
- самостійні додатки, які можуть програватись як Web-браузером, так і певним програвачем, і створюються для розв'язання широкого кола задач;
- Flash-фільми не потребують зовнішніх ресурсів (наприклад, встановлення шрифтів), крім Flash-плеєра і мають кросплатформенний двійковий формат, сумісний з усіма основними операційними системами;
- використання для побудови зображень векторної графіки та особливих методів стиснення файлів, надали фільмам Flash таких властивостей, як: невеликий розмір файлу, висока швидкість відображення, масштабування зображення без втрати його якості [1].

Крім того, Flash-фільми можуть бути інтегровані у будь-який додаток, розроблений у якому-небудь інтегрованому середовищі, наприклад, Delphi чи C++Builder, через використання відповідного елемента ActiveX. Враховуючи таке широке коло можливостей Flash 5, програмістам при розробці інтерфейсів користувача необхідно якнайширше використовувати Flash-анімацію.

Література

1. Flash 5. Библия пользователя: Пер. с англ. – М.: Издательский дом «Вильямс». – 2002. – 1164 с.
2. Организация взаимодействия человека с техническими средствами АСУ. Том 4: Отображение информации, ред. В.Н. Четвериков. – М.: Высшая школа, 1993.
3. Рекомендации по общепользовательскому интерфейсу. – Microsoft, редакция 1995.

ДОСВІД ВПРОВАДЖЕННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ У НАВЧАЛЬНИЙ ПРОЦЕС ЛЬВІВСЬКОГО БАНКІВСЬКОГО ІНСТИТУТУ

Г.Т. Кравчук, Т.В. Шевчук
м. Львів, Львівський банківський інститут НБУ
kt@lbi.wubn.net

Економічна наука в Україні, її розвиток та викладання перебувають нині на новому етапі розвитку. Забезпечення сфери фінансів України кваліфікованими кадрами передбачає підготовку фахівців з економіки, котрі вміють використовувати інформаційні технології та створювати інформаційні системи для ефективного функціонування банку, підприємства, установи тощо.

Сучасні умови життя ініціюють появу економістів нової формації, котрі здатні самостійно розв'язувати економічні проблеми, володіють методами математичного моделювання соціально-економічних процесів, швидких і точних розрахунків, встановлення комунікаційних зв'язків із діловими партнерами за допомогою глобальних та локальних інформаційних мереж, вміють розпізнати причини кризових та ризикових явищ, спланувати ефективні заходи щодо їх запобігання та усунення, виробити ефективні рішення стосовно підвищення ефективності управління банками, підприємствами та установами [1, 3].

Головним завданням щодо підготовки фахівця у Львівському банківському інституті (ЛБІ) є розвиток і формування його особистості, здатної логічно і критично мислити, творчо діяти, знаходити оптимальні рішення, впроваджувати нові засоби і технології професійної діяльності. Наголосимо, що для такої підготовки в інституті впроваджено розвиваюче навчання, яке має дослідницький характер, відтворює суттєві риси предметного і соціального контексту майбутньої професійної діяльності, базується на сучасних досягненнях науки і техніки.

Впровадження у навчальний процес інституту сучасних комп'ютерних технологій відкриває можливості для моделювання різноманітних виробничих ситуацій, організації автоматизованої обробки і математичного аналізу даних для різних галузей економіки, а також призводить до більшого ускладнення змісту і прийомів навчальної діяльності студентів. Така діяльність характеризується рисами наукового пошуку, вона вимагає від студентів економічних і технологічних знань, оперативності при прийнятті рішень, вміння користуватися сучасними засобами управління інформаційними потоками [2].

Зауважимо, що питання запровадження сучасних інформаційних технологій у фінансовій, зокрема банківській сфері актуальні, перспективні, і, найголовніше, – прибуткові. Тому від фахівців вимагається високий рівень

знань і практичних навиків, набуття яких забезпечується вивченням у Львівському банківському інституті дисциплін “Інформатика і комп’ютерна техніка”, “Інформаційні системи і технології у банківській сфері”, “Інформаційні системи і технології у фінансах”, “Інформаційні системи і технології обліку”.

Базові знання в галузі інформаційних технологій студенти отримують на першому і другому курсах, вивчаючи сервісне програмне забезпечення, програмування, автоматизовані системи обробки даних, комп’ютерні мережі, серверні системи управління базами даних. На старших курсах підготовка студентів проводиться на базі сучасних інтегрованих автоматизованих систем обробки даних, що сприяє ефективності роботи фінансово-кредитних установ. Студенти вчать використовувати базові програмні засоби та готові пакети прикладних програм для розв’язання задач, що виникають у фінансовій, зокрема банківській діяльності, в умовах функціонування автоматизованих робочих місць (АРМ), а також в середовищі інтегрованої автоматизованої банківської системи.

Для відтворення реальних виробничих ситуацій, які вимагають від студентів уміння оцінити, проаналізувати та спрогнозувати виробничу ситуацію, знайти оптимальне рішення у навчальному процесі запроваджено реальні прикладні системи: система автоматизації банку (САБ) Scrooge II; система електронних платежів; електронна система передавання документів “Клієнт–Банк”; програмний комплекс “Торговий центр” (“Softmarket”); система банківського обслуговування населення (СБОН); комплексна автоматизація бухгалтерії “ІС:Бухгалтерія 7.7”.

Перелічені прикладні програми широко застосовуються у практичній діяльності фінансово-кредитної системи України. Зокрема, САБ “Scrooge II”, розроблена фірмою Line System (м. Донецьк), забезпечує комплексну автоматизацію банківської діяльності і успішно експлуатується більш ніж у 80-ти банках України. Модулі системи “Облік основних засобів і матеріальних цінностей”, “Заробітна плата працівників” автоматизують діяльність внутрішньої бухгалтерії банку. Модулі “Обробка грошово-розрахункових операцій”, “Кредитні операції”, “Депозитні операції”, “Валютні операції”, забезпечують автоматизацію банківської діяльності у названих напрямках [5].

Вважаємо за доцільне зупинитись на можливостях комп’ютерної програми “Scrooge II”. Відкритість архітектури цієї системи дозволяє взаємодіяти з багаточисельними зовнішніми додатками, в яких реалізовано увесь набір основних і додаткових банківських функцій системи, а саме:

- повноцінна підтримка багаторівневих, мультивалютних рахунків;
- автоматичне формування складних валютних документів з одночасним виконанням необмеженої кількості проводок;
- інтерфейс з довільними платіжними системами (система електронних платежів України, СВІФТ, система електронних платежів СНД), дозволяє

без проблем реалізувати моментальне шлюзування у ці системи;

- механізм “фільтрів” для створення запитів із довільних джерел даних з прив’язкою і настройкою конкретного користувача;

- підтримка розподілених баз даних, а також створення, збереження та обробку звітів безпосередньо на Report-сервері.

Для студентів, які навчаються за спеціальністю “Банківська справа”, адаптовано до навчального процесу такі підсистеми САБ “Scrooge II”: система електронних платежів, система “Клієнт–Банк”, валютний обмінний пункт, репозитарій, підсистема комунальних платежів, кредити / депозити / операційне обслуговування.

При використанні інноваційних комп’ютерних технологій традиційні форми і засоби навчання вільно поєднуються з індивідуальними, груповими і лекційними формами роботи в комп’ютерних лабораторіях. Так, з метою наближення студентів до практичної діяльності при АКБ “Трансбанку” створено філію кафедри комп’ютерних технологій Львівського банківського інституту, де проводяться практичні заняття в банку. Навчальні лабораторії обладнано комп’ютерною технікою із сучасним банківським програмним забезпеченням. Заняття проводяться викладачами кафедри та банківськими працівниками, які, змодельовавши реальну ситуацію в банку, пропонують студентові самостійно приймати рішення при виконанні завдання [4].

Студенти, які мають доступ у професійні банки і бази даних, знайомляться з сучасними науковими проблемами. Використання добре структурованої інформації, яка є у базах даних, надає можливість перевірити приватні гіпотези, допомагає студентам запам’ятати інформацію, сприяє формуванню прийомів логічного мислення.

Як свідчить наш досвід використання комп’ютерних засобів супроводу навчального процесу, без відповідної науково-методичної системи неможливо сподіватися на ефективне використання інформаційних технологій у навчальному процесі.

З цією метою науково-педагогічними працівниками кафедри розроблено комплекс навчально-методичного забезпечення дисциплін (інструкції для роботи з підсистемами, методичні рекомендації, завдання для проведення практичних занять, індивідуальні завдання для самостійної роботи студента), що охоплює різні напрями діяльності банківських послуг. Робота із системою “Scrooge II” починається із початкового впровадження, а саме з процесу реєстрації клієнтів банку і відкриття необхідних (передбачених завданнями) рахунків. Вона дає можливість студентам ознайомитися і використовувати для виконання практичних завдань реальних електронних банківських довідників, а саме плану рахунків, довідника типів операцій, довідника касових символів, довідника видів документів, довідника валют, довідника курсів валют.

Під час роботи з модулями “Обробка грошово-розрахункових операцій”, “Кредитні операції”, “Депозитні операції” студенти здійснюють відпо-

відно до запропонованих завдань автоматизовану обробку грошово-розрахункових і касових документів, настроювання правил обслуговування груп клієнтських рахунків; ведення депозитних договорів; ведення кредитних рахунків і договорів; нарахування відсотків за договорами згідно до раніше заданих правил обслуговування; побудову звітних форм – відомостей і меморіальних ордерів.

Технологія обробки міжбанківських платежів у “Scrooge II” забезпечується пакетом додатків, які підтримують інтерфейс з системою електронних платежів НБУ (ЕП НБУ). Система електронних платежів (СЕП) “Scrooge II” призначена для:

- керування циклом обробки зовнішніх платежів;
- відображення поточних і прогнозованих значень кореспондентських рахунків банку і його філій;
- перегляду, бракування і відновлення файлів СЕП;
- перевірки відповідності коррахунку в Розрахунковій палаті і в банку;
- візування платежів в “ручному” і автоматичному режимі як додатковий засіб для керування по черговістю руху платежів, а також для візуального контролю реквізитів платежів;
- інтерфейсу до інформаційно-пошукової системи записів про походження міжбанківських платежів в СЕП.

Відзначимо, що для проведення практичних занять у комп’ютерних класах встановлено стендову версію СЕП на прикладі двох регіонів (областей), в яких функціонує по три банки. Обробку документів, формування електронних платіжних документів, обмін файлами забезпечують автоматизовані робочі місця з відповідним комплексом програмно-технічного забезпечення. Формування пакета початкових платежів у банку забезпечує програмне забезпечення ОДБ (операційний день банку), взаємодію банку із регіональною розрахунковою палатою – АРМ-НБУ, ведення технічних коррахунків та формування квитанцій – АРМ-2. Студенти мають можливість відслідкувати проходження платежів в межах одного регіону та на міжрегіональному рівні. На підставі результатів аналізується робота банків.

Подібно до системи електронних платежів побудована система “Клієнт–Банк”, яка призначена для зв’язку між банком і його відділеннями або клієнтами, обміну електронними документами, оперативного управління клієнтськими рахунками в банку тощо.

Для студентів, які навчаються за спеціальностями “Фінанси”, “Облік і аудит” у навчальний процес впроваджуються програмні комплекси “Торговий центр”, “ІС:Бухгалтерія”.

Система “Торговий центр” (“Softmarket”) призначена для ведення обігу товарів та бухгалтерських документів торгової фірми. Пакет включає наступні модулі:

- модуль “Склад” призначено для обліку товарів та генерації документів;

- модуль “Каса” забезпечує виконання касових операцій на підключеному через комп’ютер касовому апараті;
- модуль “Адміністратор” здійснює базові настройки для роботи усіх програм пакету;
- модуль “Банк” – це програма призначена для генерації виписок з банку;
- модуль “Головний бухгалтер” призначено для ведення бухгалтерського обліку на підприємствах торгівлі з різними формами власності та напрямками діяльності;
- модуль “Звіти” призначений для підтримки створення та редагування звітів.

Викладачами кафедри комп’ютерних технологій розроблено комплекс навчально-методичного забезпечення щодо проведення практичних занять, в якому змодельовано діяльність торгової фірми. Студенти вчать менеджменту з постачальниками, клієнтами, банками, веденню бухгалтерської звітності та податкових документів, обліку складських товарів тощо.

Комп’ютерна програма “1С:Бухгалтерія” забезпечує комплексну автоматизацію бухгалтерського обліку підприємства, тому випускники спеціальності “Облік і аудит” можуть активно використовувати навички роботи в автоматизованій системі бухгалтерського обліку у різних напрямках діяльності.

Таким чином, діапазон можливостей інноваційних інформаційних технологій дозволяє ефективніше організувати навчальний процес та реалізувати індивідуальний підхід до студентів. З цієї точки зору використання комп’ютерних технологій забезпечує:

- індивідуалізацію і диференціацію процесу навчання за рахунок можливостей поетапного просування до мети за лініями різних ступенів труднощів;
- здійснення самоконтролю та самокорекції;
- можливість тренінгу і здійснення за його допомогою самопідготовки;
- посилення мотивації навчання (за рахунок зображувальних засобів комп’ютерних програм, застосування ігрових ситуацій);
- формування економічного стилю мислення (за рахунок систематичної логічної послідовності операцій, закладених у програму);
- розвиток творчих якостей студентів.

Отже, практична реалізація накопиченого практичного досвіду використання ПЕОМ як сучасного засобу навчання дозволяє стверджувати: програмні продукти є якісно новими елементами в системі підготовки висококваліфікованих фахівців, вони мають унікальні можливості інтенсифікації навчального процесу, що дозволяє вивільнити час студентів для творчої аналітичної роботи.

На закінчення зазначимо, що увійти у світ майбутньої професії й оцінити свої можливості та знання можливо шляхом участі студентів у динаміч-

них імітаційних та модельованих процесах, бо саме імітаційне моделювання виступає як засіб адаптації студентів до майбутньої професійної діяльності.

Література:

1. Єршоміна Н.В. Банківські інформаційні системи: Навч. посібник. – К.: КНЕУ, 2000. – 220с.
2. Костіна Н.І., Антонов В.М., Ганах Н.І. Банки: сучасні інформаційні технології: Навчальний посібник. – Ірпінь, 2001. – 359с.
3. Кредитна система України і банківські технології: Навч. посібник: у 3 кн. За заг. ред. д-ра екон. наук, проф. І.В. Сала. – Львів: ЛБІ НБУ, 2002.
4. Міщенко В.І., Шаповалов А.В., Юрчук Г.В. Електронний бізнес на ринку фінансових послуг. – К.: Знання, 2003. – 278 с.
5. Документація до банківської системи “Scrooge II” (www.scrooge.dn.ua).

ОСОБЛИВОСТІ ФОРМУВАННЯ СИСТЕМИ НАВЧАЛЬНИХ ЗАДАЧ З ІНФОРМАТИКИ

Ю.М. Красюк

м. Київ, Київський національний економічний університет
krasyuk_y@ukr.net

Виступаючи важливим компонентом навчальної та навчаючої діяльності, навчальна задача є одним з основних понять у дидактиці та педагогічній психології. Дослідженню особливостей навчальних задач, їх структури та способів розв'язування присвячено значну кількість наукових праць. У результаті сфера використання даного терміну розширилася від завдань з математики, хімії, фізики, коли він означав лише частковий вид завдань, які застосовувалися поряд з прикладами, вправами та самостійними роботами, до навчальних задач із біології, географії, інформатики і т.ін. З впровадженням у навчальний процес інформаційно-комунікаційних технологій навчання почали говорити про задачі на постановку діагнозу, задачі на моделювання виробничих та інших ситуацій. Проте загально визнаного тлумачення даного поняття так і не було прийнято.

Так на думку Д.Б. Ельконіна, особливість навчальної задачі полягає в тому, що передбачені викладачем способи її розв'язування повинні привести до зміни психічного розвитку самих студентів. Тобто навчальна задача вважатиметься розв'язаною лише за умови, що сталися заплановані зміни в суб'єкті навчальної діяльності (студента). При цьому Д.Б. Ельконін відзначає, що навчальна задача має місце лише там, де основна ціль тих, хто навчається, полягає в засвоєнні потрібного способу дій. Тому найбільш суттєвою характеристикою навчальних задач стає спрямованість на засвоєння студентами орієнтувальної частини способу дій [3, с. 215].

Водночас до класу навчальних задач Д.Б. Ельконін не включає конкретно-практичні задачі, які спрямовані на отримання результату, що міститься в умові задачі. Однак, у процесі навчання інформатики неможливо чітко відділити результат від способу дії, оскільки це може привести до помилкового оцінювання правильності розв'язування навчальної задачі. Наприклад, при виконанні навчальної задачі з використанням табличного процесора MS Excel студент, правильно демонструючи спосіб дії, допускає помилку під час запису формули. Згідно з вище наведеним тлумаченням навчальна задача вважається розв'язаною, хоча отриманий результат є помилковим.

Ю.І. Машбиць розширив тлумачення навчальної задачі [1, с. 59]. Він розглядає навчальну задачу як будь-яку задачу, яка пред'являється викладачем студентові (або яку ставить перед собою сам студент) та яка спрямована на досягнення навчальних цілей. Навчальна задача є компонентом як навчальної діяльності студентів, так і навчаючої діяльності викладача. В останньому випадку вона виступає в якості навчаючого впливу. Постановка на-

вчальної задачі в якості засобу управління навчальною діяльністю спочатку здійснюється викладачем, а потім довізначається кожним студентом відповідно до його сформованих цілей, що у великій мірі залежить від проведеної викладачем мотиваційної та пропедевтичної роботи. Адже, як зазначав С.Л. Рубінштейн, для того, щоб студенти по-справжньому включилися в роботу, необхідно зробити поставлені перед ними навчальні задачі не тільки зрозумілими, але й внутрішньо прийнятими ними, тобто щоб вони набули значимості для кожного студента та знайшли, таким чином, відгук та опорну точку в його переживаннях [2, с. 82].

Необхідною умовою віднесення задачі до класу навчальних є наявність зв'язку між процесом розв'язування задачі та досягненням навчальних цілей. При цьому управління навчальною діяльністю повинне забезпечувати як засвоєння системи засобів, що є необхідними та достатніми для успішної навчальної діяльності, так і просування до визначених навчальних цілей (близьких і віддалених).

Це можливе лише у тому випадку коли викладач буде конструювати не кожну окрему навчальну задачу, яка буде відірвана від всіх інших, а систему навчальних задач (для семінарського, практичного заняття або групи практичних занять, лабораторної роботи, самостійної роботи студентів і т.ін.) з чітко визначеними цілями та призначенням, місцем кожної в системі навчальних задач. Адже загальновідомо, що корисність навчальної задачі в великій мірі визначається тим, яка навчальна задача розв'язувалася перед нею, і яка буде розглядатися наступною. Тому під час формування системи навчальних задач викладач повинен враховувати:

- рівень складності кожної навчальної задачі. Він визначається її структурою: кількістю об'єктів, які входять до складу умови задачі, їх взаємозв'язками, кількістю операцій, які необхідно здійснити;

- рівень трудності кожної навчальної задачі. Він розглядається як функція двох змінних: ймовірності правильного розв'язування задачі певним контингентом студентів та часом, який витрачається на її розв'язування. При оцінці даного параметра потрібно враховувати як об'єктивно-логічні (перш за все рівень складності), так і суб'єктивні її характеристики (здібності студентів, особливості мотиваційної сфери, наявність необхідних знань, умінь та навичок);

- рівень проблемності кожної навчальної задачі, що вказує на вихід студента за рамки алгоритмів (якими він володіє) під час розв'язування задачі;

- фахову спрямованість навчальної задачі відповідно до спеціальності студентів.

Водночас необхідно пам'ятати – розв'язування навчальної задачі виступає не як ціль навчальної діяльності, а як засіб її досягнення, що визначає особливість контролю навчальної діяльності зі сторони викладача та студента [1, с. 60]. Викладачеві недостатньо перевірити тільки один

розв'язок задачі. Необхідно перевіряти також ті зміни (зміни в знаннях, уміннях використовувати поняття та узагальнювати, абстрагувати та конкретизувати, аналізувати та синтезувати), які відбулися в суб'єкті під час розв'язування задачі, оскільки саме вони виступають дійсним продуктом навчальної діяльності. Ю.І. Машбиць підкреслює, що для самого студента контроль за правильністю розв'язування задачі означає спрямованість свідомості на власну навчальну діяльність, на абстракцію та узагальнення дій, які він здійснює, що означає наявність рефлексивної саморегуляції [1, с. 60].

Зазначені вимоги до організації контролю навчальної діяльності студентів передбачають реалізацію індивідуалізованого підходу з урахуванням результатів комплексної діагностики на всіх етапах навчання та наявність постійного зворотного зв'язку за умови раціонального використання навчального часу. Для успішного вирішення вказаних завдань у процесі навчання інформатики поряд з традиційними методами контролю необхідно систематично та цілеспрямовано використовувати навчально-контролюючі системи та системи автоматизованого контролю знань.

Література:

1. Машбиць Е.И. Психолого-педагогические проблемы компьютеризации обучения: Педагогическая наука – реформе школы. – М.: Педагогика, 1988. – 192 с.
2. Рубинштейн С.Л. Основы общей психологии: В 2-х т. / АПН СССР. – М.: Педагогика, 1989. – Т.1. – 485 с.
3. Эльконин Д.Б. Избранные психологические труды. – М.: Педагогика, 1989. – 560 с.: ил.

ОБЩИЕ ПОДХОДЫ К ОБУЧЕНИЮ ПОЛЬЗОВАНИЮ ПРОГРАММНЫМИ СРЕДСТВАМИ ПАКЕТА MICROSOFT OFFICE

И.А. Кривель¹, А.Н. Моргун²

¹ г. Черкассы, Черкасский национальный университет
имени Богдана Хмельницкого

² г. Черкассы, Черкасский институт пожарной безопасности
имени Героев Чернобыля
a_n_m@rambler.ru

Учитывая современные тенденции и возрастающую роль самостоятельной работы при изучении информатики и информационных технологий студентами некомпьютерных специальностей вузов, необходимо критически подойти к сложившейся методике преподавания. Здесь, в первую очередь, требуют пересмотра содержание и стиль изложения тех вопросов, которые напрямую связаны с освоением наиболее популярных и практически значимых компьютерных технологий обработки данных.

Дисциплина «Информатика и компьютерная техника», изучаемая на просторах СНГ, как правило, студентами младших курсов вузов, с каждым годом становится всё более «фундаментальной». Всё чаще учебными требованиями предметов, ранее весьма далёких от вычислительной техники, предусматриваются основные составляющие компьютерной грамотности. Учащаяся молодёжь всё шире, зачастую «явочным» порядком, внедряет компьютерные технологии в свою учебную деятельность буквально с первого курса. Здесь следует вспомнить оформление курсовых работ, размножение конспектов, подготовку «шпаргалок», использование впечатляющих компьютерных презентаций во время защит, поиск литературных источников в Internet, «скачивание» оттуда же рефератов и многое другое.

Безусловно положительным фактором следует считать также и то, что в последние годы круг изучаемых разделов в рамках общего курса информатики существенно стабилизировался [1–3]. И главенствующие позиции по праву занимает в нём пакет прикладных программ Microsoft Office. Из всех программных средств пакета наиболее значимыми считаются текстовый процессор Word, табличный процессор Excel, а также система подготовки презентаций PowerPoint. Тем более, что и основы программирования, и работу с базами данных вполне можно осваивать в рамках табличного процессора Excel [4].

Из упомянутых прикладных программ практически всегда в первую очередь изучается текстовый процессор Word. Именно поэтому общеметодические подходы к обучению пользователей пакета Microsoft Office целесообразно рассмотреть на его примере.

История серьёзного изучения текстовых процессоров в высших учеб-

ных заведениях перевалила за десяток лет. Следовательно, уже пора делать определённые выводы из накопленного опыта, тем более, что итоговых методических материалов в этом направлении явно недостаточно.

Усиление роли учебной литературы. Один из основных выводов состоит в том, что процессу обучения компьютерным технологиям свойственен повышенный уровень самостоятельности и индивидуализации темпов со стороны обучающихся. Это связано, прежде всего, с личностными особенностями и, кроме того, с тем, что в результате обучения предусмотрено получение значительного объёма практических навыков. Если коллективный метод обучения тут и пригоден, то лишь до определённого уровня, по достижении которого он утрачивает эффективность. Именно поэтому следует признать исключительную роль качественной учебной литературы в самостоятельной работе обучающихся, без которой полноценный учебный процесс невозможен.

Уровень качества учебной литературы. Общеизвестно, что в количественном отношении литературы по текстовым процессорам расплодилось более чем достаточно. А вот качество пособий различно. Если проанализировать с методической точки зрения даже лучшие из них, то легко обнаружить, что преобладающая часть содержания в них чаще всего имеет описательный характер. Однако, опыт преподавания в учебных заведениях убеждает в том, что эффективное обучение достигается не в результате чтения описаний. Глубокие знания и уверенные навыки являются следствием выполнения системы специально подобранных и специально ориентированных практических заданий. К сожалению, полноценные книги подобного рода практически отсутствуют. Книга должна быть такой, чтобы её можно было использовать и как пособие для самостоятельной работы, и как учебник во время занятий в учебных заведениях. Иначе говоря, книга должна быть руководством к действию.

Структурирование процесса обучения. Для существующей учебной литературы характерен ещё один недостаток, связанный с отсутствием чёткого разделение учебного материала по уровням функционирования текстового процессора. Эти уровни и их последовательность фактически предопределены логикой преподавания. К ним относятся:

- загрузка текстового процессора;
- настройка текстового процессора;
- создание документа;
- ввод и редактирование текста документа;
- оформление (форматирование) текста документа;
- создание специфических объектов в составе текста документа;
- автоматизация подготовки документа;
- издание документа.

Отметим, что изучение текстового процессора обязательно должно быть ориентировано на стандартную его настройку, что в существующей

учебной литературе, как правило, вообще выпадает из поля зрения.

Связь с операционной средой. Ещё один недостаток существующих учебных книг состоит в том, что в них «забыта» тесная связь текстового процессора с операционной средой. И то, как будет работать текстовый процессор, во многом зависит от умения пользователя сориентироваться в целом ряде настроек операционной системы. Было бы целесообразно исключить абстрактное изучение Windows и рассматривать её возможности в контексте будущего изучения и использования текстового процессора.

Не описания, а пошаговые инструкции. Авторы, как преподаватели, вынуждены ещё раз подчеркнуть, что серьёзным недостатком существующих учебников является их построение по принципу обычного описания тех или иных возможностей и функций текстового процессора. При этом начинающий читатель, чаще всего, попадает в сложную ситуацию. Учащийся растерян, он не знает, что нужно сделать (какие конкретные действия выполнить), чтобы изучить описанную функцию и получить правильный практический результат её выполнения. Кроме того, читать описания, пусть даже очень подробные, скучно и утомительно. Возможно, именно поэтому большинство учащихся имеют лишь поверхностные знания и не используют возможностей текстового процессора даже наполовину.

Таким образом, общий вывод состоит в том, что изучение функций текстового процессора следует вести посредством выполнения специально подобранных отлаженных упражнений в виде пошаговых инструкций от «нуля» и до момента получения результата.

Справедливости ради отметим, что упомянутые пошаговые инструкции нужны не всегда. Во многих случаях возможность опробовать ту или иную функцию оказывается очевидной. Но, вот в критических случаях, когда необходимо строго соблюсти определённую последовательность действий, чтобы добиться положительного результата, упражнения незаменимы.

Дополнительный положительный эффект тут состоит в том, что в процессе работы с упражнением приобретается необходимый навык в работе на клавиатуре.

В качестве примера приведём простейшее упражнение из седьмой главы пособия [5], посвящённой созданию и редактированию таблиц.

Упражнение 7.1. Создание таблицы с помощью кнопок панели инструментов **Стандартная**.

➤ Загрузите текстовый процессор любым известным вам способом. На экране появляется окно пустого документа **Документ1**.

➤ Создадим таблицу размером 5x7 (5 строк и 7 столбцов). Установите текстовый курсор в ту строку документа, начиная с которой необходимо разместить таблицу.

➤ Нажмите на кнопку **Добавить таблицу** панели инструментов **Стандартная**. В результате этого на экране появляется решётчатая панель размером 4x5 с индикатором размера таблицы в нижней части.

➤ Удерживая в нажатом состоянии ЛКМ (левую кнопку мыши) и контролируя значение на индикаторе, протяните указатель мыши вправо и вниз до момента достижения нужных размеров таблицы 5x7.

➤ Отпустите ЛКМ. В результате в текущую строку вставляется пригодная для редактирования таблица заданного размера.

➤ Закройте окно документа, не сохраняя его.

Доброжелательное отношение к читателю. Чрезвычайно серьёзные требования к учебным материалам связаны с необходимостью обеспечить доброжелательное отношение к читателю посредством соответствующего стиля изложения материала. У читателя не должно возникать чувство собственной неполноценности, а со стороны авторов не должно быть проявлений превосходства. Именно поэтому, в частности, учебные материалы должны быть самодостаточными, с тем, чтобы исключить необходимость консультирования у, так называемых, «специалистов» и у прочих «продвинутых чайников», которые позволяют себе пренебрежительное отношение к начинающим.

Уровень предварительной компьютерной подготовки, необходимой для успешной работы с учебной книгой, должен быть минимальным. В этом случае подготовленные читатели уточнят или просто пропустят знакомый им материал, а отсутствие значительных затруднений придаст им уверенности в собственных силах. Главное же, чтобы всё было понятно начинающим.

Серьёзные требования предъявляются также и к уровню педагогической квалификации авторов, которые должны хорошо знать типичные ошибки и затруднения начинающих.

Нужны ли рисунки? Ещё одна интересная особенность учебного процесса. При изучении компьютерных технологий с помощью книги рисунки оказываются ненужными. Действительно, вполне очевидно, что изучение текстового процессора обязательно должно сопровождаться реальной деятельностью на компьютере. Тогда для чего нужны иллюстрации? Главное, что необходимо, это следить за поясняющими надписями, которые появляются на экране, и чётко следовать пунктам выполнения упражнений. Таким образом, можно сделать вывод, что до 40% содержания существующих учебных книг оказывается лишним за счёт рисунков. То, что сейчас представлено на книжном рынке, могло бы быть почти вдвое дешевле.

Преимущественное использование клавиатуры. Важно то, что обучаемых следует ориентировать на преимущественное использование клавиатуры. Разумеется, использование мыши наглядно само по себе, но, к сожалению, замедляет выполнение операций. Бесспорно, что использовать клавиши сложнее (хотя бы потому, что их назначение нужно помнить). Однако, с течением времени начинающий обязательно убедится, насколько существенно ускоряет его работу именно использование клавиатуры.

Свой опыт преподавания авторы учли в изданных пособиях и книгах [5,

б], которые пользуются большой популярностью среди учащихся.

Литература

1. Інформатика : Комп'ютерна техніка. Комп'ютерні технології : Підручник для студентів ВНЗ / За редакцією О.І. Пушкаря. – К.: Видавничий центр “Академія”, 2003. – 704 с., іл.

2. Лабораторний практикум з інформатики та комп'ютерних технологій : Навчальний посібник / За редакцією О.І. Пушкаря. – Харків: Видавничий дім “ІНЖЕК”, 2003. – 424 с., іл.

3. Информатика : Базовый курс : Учебник для вузов / Симонович С.В. и др. – СПб.: Издательство “Питер”, 1999. – 640 с., ил.

4. Кривель І.А., Моргун А.Н. Некоторые аспекты преподавания основ алгоритмизации и программирования // Теорія та методика навчання математики, фізики, інформатики : Збірник наукових праць. Випуск 4. Т. 3. – Кривий Ріг: Видавничий відділ НМетАУ, 2004. – С. 125-129.

5. Моргун А.Н. MS Word. Руководство к действию : Учебно-справочное издание. – М.: “Диалог-МИФИ”, 2004. – 316 с., ил.

6. Частоколенко І.П., Моргун О.М., Моргун Л.О., Акіншин В.Д. Практикум з комп'ютерних технологій (Windows 98, Microsoft Office 2000) : Навчальне видання. – Черкаси: ЧПБ, 2004. – 120 с., іл.

КОМП'ЮТЕРНІ СИСТЕМИ КОНТРОЛЮ ЗНАНЬ У ДИСТАНЦІЙНОМУ НАВЧАННІ

А.П. Кудін^α, Ю.А. Свистун^β

м. Київ, Інститут дистанційного навчання

Національного педагогічного університету імені М.П. Драгоманова

^α kudin@npu.kiev.ua

^β svistun@ukr.net

Використання комп'ютерного тестування дозволяє у короткий час перевірити знання у великій кількості слухачів або учнів, оперативно виявляти труднощі в сприйнятті навчального матеріалу і усувати недоліки при викладанні навчального матеріалу, застосовувати методи математичної статистики для оцінки ступеня засвоєння матеріалу. Часто самі слухачі віддають перевагу тестовим методам контролю знань, вважаючи його більш об'єктивним. Як підтвердження цьому є рекомендації Міністерства освіти і науки України про проведення іспитів у вищих навчальних закладах у формі тестів, з метою боротьби проти зловживань [1].

Найпоширенішою формою тесту є вибір правильної відповіді з декількох запропонованих. Цей тип тестів має як свої переваги, так і недоліки. Основний недолік – наявність сформульованої підказки, яка у випадку, наприклад, іноземної мови, що дає можливість перевірити на звучність різні варіанти фрази і тим самим спрощує пошук правильної відповіді. Однак буде неправильно відкидати його. Цей тип тестів має простий критерій правильності відповіді і простоту організації введення відповіді у програму. І як результат – ця форма достатньо легко програмно реалізується на комп'ютері.

Системи сучасного контролю знань повинні відповідати наступним вимогам:

1. Список питань повинен бути реалізований як у виді тексту, так і у виді графіки, малюнку, аудіо-, відеофрагменту.

2. Повинна бути задана можливість зміни параметрів тестування (часу, кількості балів, кількість запитань і т.д.).

3. При відповіді на запитання повинні реалізовуватись різні форми вибору відповіді “один правильний”, “декілька правильних” і т.д.

4. Якщо суб'єкт тестування використовує декілька спроб для проходження тесту, то порядок питань повинен бути різним.

5. Інтерфейс користувача повинен бути зрозумілим, простим у експлуатації.

6. Система не повинна залежати від специфіки навчального предмета, по якому тестуються слухачі.

7. Система повинна працювати в мережі Internet.

Головна умова:

8. Система повинна бути доступною для викладачів, які не володіють навичками програмування.

Усім цим перерахованим вимогам задовольняє комп'ютерна система контролю знань “Венера – 2004”, яка була розроблена в Інституті дистанційного навчання НПУ ім. М.П. Драгоманова

У системі використовуються 5 типів завдань:

1 тип – вибір однієї правильної відповіді. Цей тип питання призначений для визначення засвоєння навчального матеріалу, тобто перевіряємо конкретне знання.

2 тип – вибір декількох правильних відповідей. Цей тип питання призначений для визначення повноти знань.

3 тип – послідовність. Цей тип тестів використовується для перевірки логічних здібностей.

4 тип – на відповідність. Фактично це розширений варіант 1 типу, який дає змогу одночасно перевірити велику базу знань.

5 тип – це введення правильної відповіді з клавіатури. Цей тип призначений для більш серйозної перевірки знань 1 типу завдань, через відсутність варіантів вибору. Тобто особа, що тестується, або знає точну відповідь, або зовсім не знає (ймовірність вгадати мінімальна).

Архітектурно система складається з декількох оболонок: “Викладач”, “Слухач”, “Адміністратор”. Для викладача існує своя оболонка для створення тестів – конструктор тестів. Вхід здійснюється через спеціальну адресу. Для кожного предмета існує свій логін та пароль (рис. 1).

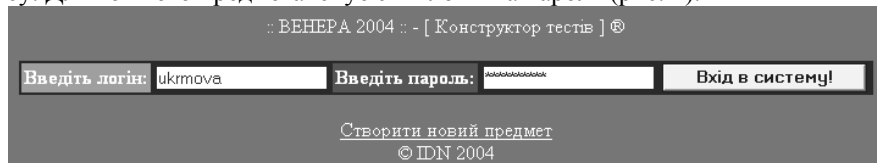


Рис. 1. Перше вікно конструктора тестів

Якщо тест не створений, натискають Створити новий предмет (рис. 1) і одразу з’являється вікно для створення нового предмету (рис. 2). Вводиться назва предмету українською мовою, тривалість тесту у хвилинах, максимальна кількість балів (можна ввести приблизну кількість, а потім змінити).

Система передбачає 3 рівні запитань. Три рівні – це умовне розділення запитань за балами. Тобто, на 1 рівні, наприклад, зібрані 5 запитань по 1 балу, на другому – 10 запитань по 2 бали, а на третьому 15 запитань по 3 бали. У випадку використання двох рівнів, у поля “Кількість запитань” і “Кількість балів” 3 рівня вводиться “0”. Система передбачає зміну ваги кожного рівня (це може бути 3, 5, 7 або 9, 11, 15 і т.д.) Зрозуміло, що основною метою створення рівнів є можливість використання простих і складних запитань по змісту, і відповідне їх оцінювання через вагу. Редагування параметрів уже в самій системі показано на рис. 3.

Створення нового предмету:	
Всі поля обов'язкові для заповнення!	
Назва предмету (українською):	<input type="text"/> Приклад: укромва7а, біологія8б, хімія9г
Тривалість тесту (хв.):	<input type="text"/> бажано в межах 60 хв.
Максимум балів:	<input type="text"/> протестуйтеся і поразуйте самі
Кількість запитань 1 рівня	<input type="text"/> до 100 запитань
Кількість балів за 1 правильну відповідь	<input type="text"/> (1,2,3,4)
Кількість запитань 2 рівня	<input type="text"/> до 100 запитань
Кількість балів за 1 правильну відповідь	<input type="text"/> (1,2,3,4)
Кількість запитань 3 рівня	<input type="text"/> до 100 запитань
Кількість балів за 1 правильну відповідь	<input type="text"/> (1,2,3,4)
Коротка анотація:	
<ul style="list-style-type: none"> Всі ці параметри, крім англійської назви предмету, можна потім змінити. Так що вводьте приблизні параметри предмету. Якщо Ви хочете наприклад, щоб 3 рівень завдань не працював введіть замість кількості запитань і балів 0. 	
До входу	
<input type="button" value="Стерти всі поля!"/>	<input type="button" value="Створити предмет!"/>

© IDN 2003

Рис. 2. Друге вікно конструктора тестів

До входу	До запитань
Якщо ви зробили якісь зміни в полях, то натисніть зберегти щоб зберегти зміни!	
Назва предмету (українською):	<input type="text" value="укрмова"/>
Тривалість тесту (хв.)	<input type="text" value="30"/>
Максимум балів	<input type="text" value="15"/>
Кількість запитань 1 рівня	<input type="text" value="1"/>
Кількість балів за 1 рівень запитань	<input type="text" value="1"/>
Кількість запитань 2 рівня	<input type="text" value="1"/>
Кількість балів за 2 рівень запитань	<input type="text" value="2"/>
Кількість запитань 3 рівня	<input type="text" value="3"/>
Кількість балів за 3 рівень запитань	<input type="text" value="3"/>
<input type="button" value="Повернути як було!"/>	<input type="button" value="Зберегти зміни!"/>

Рис. 3

Система комп'ютерного контролю знань “ВЕНЕРА 2004” використовується для тестування учнів загальноосвітніх шкіл (Український гуманітарний ліцей при Київському національному університеті ім. Т.Г. Шевченка) та слухачів навчально-підготовчого відділення дистанційної форми навчання НПУ ім. М.П. Драгоманова.

Система особливо ефективна у такій формі навчального процесу як модульно-рейтингова [2], а головним надбанням стала її доступність для викладачів-фахівців, які не володіють навичками програмування.

Література:

1. Розпорядження МОН “Щодо поліпшення роботи зі зверненнями громадян на попередження проявів службових зловживань” від 09.01.2004 http://www.mon.gov.ua/laws/MON_rozp_187.doc

2. Болонський процес у фактах і документах (Сорбонна-Болонья-Саламанка-Прага-Берлін) / Упорядники: Степко М.Ф., Болюбаш Я.Я., Шинкарук В.Д., Грубінко В.В., Бабин І.І. – Тернопіль: Вид-во ТДПУ ім. В. Гнатюка, 2003. – 52 с.

ДИСТАНЦІЙНИЙ КУРС З ІНФОРМАТИКИ ДЛЯ СЛАБОЧУЮЧИХ

А.П. Кудін^α, Л.В. Харченко^β

м. Київ, Інститут дистанційного навчання

Національного педагогічного університету імені М.П. Драгоманова

^α kudin@npu.kiev.ua

^β lili_x@ukr.net

В наш час дистанційне навчання стає реальною можливістю навчатися в індивідуальному режимі незалежно від місця і часу, отримати освіту по індивідуальній траєкторії у відповідності з принципами відкритої освіти і призване реалізувати права людини на безперервне навчання і вільне отримання інформації [1–5]. Більш за все в цього потребують люди з обмеженими можливостями у здоров'ї, в тому числі слабочуючі. Існуючі проекти організації дистанційного навчання для слабочуючих людей носять в основному ігровий характер [7, 8], причому на рівні середньої освіти для українських громадян їх практично немає. Відсутність професійного навчання робить цих дітей не потрібними на сучасному ринку праці. Вирішенню цієї проблеми призначений дистанційний курс з інформатики для слабочуючих “ІНФОРМАТИКА” [9], який розроблений на платформі Moodle в рамках міжнародної програми “Deaf Workforce Education in Ukraine” між НПУ імені М.П.Драгоманова і Університетом “Лойола коледж” (США). Цей курс навчить:

- працювати з документами, таблицями, базами даних, редагувати електронні тексти за допомогою комп'ютерних офісних програм і звільнитись від щоденної рутинної праці, яку виконують секретарі, діловоди, економісти;
- швидко і якісно донести потрібну інформацію за допомогою Web-презентації;
- створювати елементи комп'ютерної графіки;
- швидко обмінюватись інформацією в електронному вигляді;
- знаходити потрібні матеріали в Інтернеті та опрацювати їх;
- удосконалить навички роботи за комп'ютером, поповнить знання базової комп'ютерної термінології.

Після успішного проходження курсу слухач буде знати:

1. Основну комп'ютерну термінологію.
2. Принципи роботи в середовищі Windows, її основні можливості.
3. Що таке текстовий редактор, електронний документ, електронні таблиці, бази даних, презентації у PowerPointi.
4. Можливості роботи в графічних редакторах, основні інструменти для створення графіки.
5. Про архіватори і антивірусні програми, і як з ними працювати.







6. Основи побудови мережі Інтернет, принципи роботи в браузері, зокрема в Internet Explorer.

Основні вміння, що набуває слухач успішно закінчивши курс:

1. Визначати об'єм інформації
2. Працювати в середовищі Windows.
3. Створювати електронні документи засобами Word та редагувати їх, електронні таблиці Excel, створювати базу даних засобами Access та працювати з даними, та вміти створювати найпростіші презентації.
4. Створювати та редагувати графіку за допомогою графічних редакторів Paint, CorelDraw, Adobe Fotoshop
5. Архівувати файли та перевіряти файли на комп'ютерні віруси.
6. Вміти шукати інформацію в Інтернеті, зберігати її та редагувати.

У курсі передбачена перевірка знань: контрольні запитання, практичні завдання, тести. **Курс передбачає** 5-бальну систему оцінювання, оскільки для людей з особливими потребами вона є простою. **Мова навчання:** українська. Навчання модульне.













На платформі Moodle структура курсу виглядає так:

-  Форум новин
-  Реєстрація
-  Прочитати ПРОГРАМУ КУРСУ
-  Форум учасників
-  Глосарій
-  ЧАТ загальний










Зміст курсу:

Модуль 1: Основи інформатики

















1.1. Основні поняття

-  Що таке інформатика?
-  Що таке інформація?
-  Кодування інформації
-  Одиниці вимірювання інформації
-  Якою буває інформація та дії з нею
-  Засоби збереження інформації
-  Запитання №1.1.
-  Завдання до пункту 1.1.
-  Самоперевірка 1.1.
-  Форум-консультація 1.1.
-  форум відповідей 1.1.
-  Підсумковий тест 1.1.





1.2. Інформаційна система

-  Система. Системний підхід
-  Властивості систем
-  Програмне забезпечення
-  Файлова система
-  Автоматизовані інформаційні системи
-  Задання №1.2.
-  Форум-консультація 1.2.
-  Форум відповідей 1.2.
-  Підсумковий тест 1.2.

1.3. Апаратне забезпечення

-  Що таке комп'ютер
-  Пристрої комп'ютера
-  Принципи побудови комп'ютера
-  Що таке команда?
-  Виконання команди
-  Архітектура і структура комп'ютера
-  Центральний процесор
-  Пам'ять комп'ютера. Внутрішня пам'ять
-  Зовнішня пам'ять
-  Принтер, плотер, сканер
-  Аудіоадаптер, відеоадаптер
-  Відеосистема комп'ютера
-  Модем
-  Самоперевірка 1.3.
-  Форум-консультація 1.3.
-  Підсумковий тест 1.3.

1.4. Алгоритми

-  Поняття алгоритму і його види
-  Задання №1.4.
-  Форум відповідей 1.4.
-  Підсумковий тест 1.4.

На початку подано рекомендації до вивчення даного модуля:

1. Почніть з розділу 1.1. Ознайомлюйтеся з матеріалами поступово. Матеріали завжди доступні для читання.
2. В кінці розділів стоять уроки для самоконтролю. Максимальна кількість спроб відповіді на запитання на уроці – 10. Після неправильної відпові-

ді відбувається автоматичний перехід на сторінку, де буде надана можливість перечитати ще раз запитання і обрати правильну відповідь.

3. Якщо на уроці багато відповідей дає неправильно, то звідси висновок, що варто краще ознайомитись з матеріалами. Тільки після цього переходьте до виконання завдань.
4. Наприкінці вивчення модуля буде запропоновано пройти підсумковий тест.

Порада: користуйтеся календарем. Завдання на виконання обмежені часом. Виконані завдання пишуть у форум-консультацію відповідного розділу. Підсумковий тест буде ґрунтуватися на матеріалах уроків і завдань. Доступ до підсумкового тесту буде обмежений. Час доступу дивіться в календарі. Тому Вам бажано вчасно здавати завдання. Якщо у Вас виникають запитання, заходьте у форум.

Форма подання теоретичного матеріалу курсу повністю відповідає психологічним особливостям слаббачуючих: образне мислення, простота викладення і тлумачення термінів, не лімітований у часі процес навчання та ін. У цьому році цей дистанційний курс [9] проходить апробацію: більше 100 слаббачуючих дітей з 10 областей України в лютому-березні виявили бажання навчатись за ним дистанційно.

Література:

1. Обобщение и краткий анализ результатов эксперимента в области дистанционного образования / О.М. Карпенко, М.С. Антропов, В.Е. Бочков. // В сб.: Эксперимент в области дистанционного образования: результаты и перспективы. – М.: МГИУ, 2002. – 188 с.

2. Згуровський М.З., Сергієнко І.В. Інформаційні технології у сучасному суспільстві // Вісник НАН України. – 2000. – №12. – С. 9-16.

3. Олійник В. Дистанційне навчання в післядипломній педагогічній освіті. – К.: ЦППО, 2001. – 148 с.

4. Жабеев Г.В., Кудин А.П. Система дистанционного образования Национального педагогического университета имени М.П. Драгоманова // ИТО. Межд. конг. конф., 16-20 нояб. 2003 г. – М.: Просвещение. – С. 70-71.

5. Российский портал открытого образования: образование, опыт, организация / под ред. В.И. Солдаткина. – М.: МГИУ, 2003. – 508 с.

6. XII Международная конференция «Информационные технологии в образовании»: Сб. трудов уч. конф. Ч. III. – М.: Просвещение, 2003. – 248 с.

7. Віртуальна школа з лідерства для учнів шкіл-інтернатів для дітей з вадами слуху. – <http://e-schola.org.ua/>

8. Концептуальні підходи до соціально-педагогічної підтримки осіб з фізичними вадами в Україні на основі експериментальних досліджень, надбань вітчизняного та зарубіжного досвіду. – **Ошибка! Недопустимый объект гиперссылки.**

9. <http://e-schola.org.ua/course/category.php?id=4>.

ИСПОЛЬЗОВАНИЕ СОВРЕМЕННЫХ КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ ПРИ ПОДГОТОВКЕ И ЧТЕНИИ РАЗЛИЧНЫХ РАЗДЕЛОВ КУРСА «ИНФОРМАТИКА»

Г.И. Кулик

г. Днепропетровск, Приднепровская государственная академия
строительства и архитектуры
kulik.galina@mail.ru

Все более популярным становится использование электронных учебников, интерактивных средств обучения. Это обусловлено внедрением информационных технологий в учебный процесс, удобством их использования в аудиторной и внеаудиторной работе.

Во всех предметных областях в высшей школе сейчас прослеживается тенденция к «компьютеризации» учебного процесса, использованию электронных методических разработок, которые позволяют существенно повысить эффективность учебного процесса. В особом положении оказался курс «Информатика» в техническом вузе. При изучении этого курса информационные технологии являются одновременно и целью, и средством обучения.

Преимущества электронных методических разработок состоят в том, что студент получает более широкие возможности для самостоятельной работы. Нет необходимости тратить временные и материальные ресурсы на подбор необходимой литературы в библиотеках, стоимость тиражирования подобных разработок значительно ниже традиционных. Кроме того, постоянная необходимость в модификации программного материала, характерная для курса «Информатика», легче реализуется именно в таком варианте.

Использование информационных технологий позволяет, кроме того, активизировать восприятие изучаемого материала. Особенно это актуально при изучении разделов, традиционно трудных для усвоения. В частности, сложным оказывается раздел «Алгоритмизация вычислительных процессов в инженерных расчетах».

С большими трудностями при изучении этого раздела сталкиваются студенты, которые в школе либо не изучали информатику, либо изучение этого предмета оказывалось формальным и, в результате, навыки алгоритмического мышления не были сформированы. Если основные понятия, определения раздела алгоритмизации студенты усваивают относительно легко, то практическое применение, решение даже типовых задач связано с большими трудностями. И тем более проблематичной оказывается программная реализация задач без правильно разработанного алгоритма.

При изучении раздела алгоритмизации часто останавливаются на графическом способе представления алгоритмов, как наиболее наглядном. Однако, при изложении лекционного материала, применение этого способа связано с рядом проблем. Использование традиционных «инструментов»

преподавателя, таких как мел и доска, является в данном случае трудоемким и приводит к ослаблению контакта с аудиторией. Пока преподаватель занят вычерчиванием графических образов на доске, студенты, в лучшем случае, автоматически перечерчивают их в тетрадь. Использование заранее подготовленных плакатов с рассматриваемыми примерами также не является решением проблемы; так как на плакате представлен конечный результат, а в данном случае особенно важен процесс поиска решения.

Перспективным, по мнению автора, является путь использования возможностей, заложенных в традиционно используемых программных продуктах – приложениях MS Office (Word, Power Point).

Подготовка конспектов лекций преподавателями уже давно ведется в электронном варианте. В том числе готовятся и примеры алгоритмов, которые будут рассмотрены на лекциях. Это обычный вид представления материала, который мы встречаем в различных учебных пособиях. Применяемая автором демонстрация подготовленных материалов с использованием возможностей Power Point, в частности, анимационных эффектов, позволяет управлять процессом восприятия нового материала.

Так, например, возможность демонстрировать заранее подготовленный материал по частям, управлять появлением графических объектов, из которых состоит рассматриваемая блок-схема, на экране, который заменяет доску, позволяет не навязывать студенту уже готовый вариант решения, а рассуждая вместе с ним, прийти к желаемому результату.

Кроме того, анимационные эффекты, заложенные в приложении Power Point, позволяют дополнительно активизировать внимание студентов. Совместная работа зрительных, звуковых анализаторов дают возможность легче воспринимать новый материал и надолго запоминать изложенное.

В свою очередь, для преподавателя очень удобной является возможность акцентировать внимание на различных моментах в процессе изложения, зрительно выделить наиболее существенные пункты, возможность вернуться в любой момент лекции к нужному материалу.

Применение методических разработок, подготовленных в виде презентаций, находит разнообразное применение. Это и иллюстрации к лекционному материалу, и электронные справочники по различным разделам курса. Собранные в одно целое подобные разработки могут быть представлены как электронный учебник. В настоящее время автором подготовлены презентации по разделам алгоритмизации и программирования, по работе с электронными таблицами.

Необходимо отметить, что, чтение лекций с использованием современных компьютерных технологий существенно меняет отношение студентов к предмету. С возрастающим интересом студенты осваивают программный материал, увеличивается количество студентов, которые творчески подходят к выполнению заданий и стремятся глубже изучить курс «Информатика». А именно это и является желаемым результатом для преподавателя.

НЕЙРОННАЯ СЕТЬ КАК ЭЛЕМЕНТ СИСТЕМЫ УПРАВЛЕНИЯ ОБУЧЕНИЕМ

Е.А. Лавров, Н.Л. Барченко
г. Сумы, Сумской национальный аграрный университет
vs@sau.sumy.ua

Развитие информационных технологий в образовании требует нового подхода в проектировании средств обучения. Актуальным является разработка систем, способных адаптироваться к индивидуальным возможностям и потребностям отдельных пользователей.

Под адаптацией подразумевается способность системы приспосабливаться к изменяющимся внутренним и внешним условиям при помощи изменения своей структуры и значений параметров.

Механизм адаптации может быть реализован посредством создания модели обучаемого (МО) и реализации процедур адаптации в системе управления учебной деятельностью. Основными моделями, используемыми в настоящее время в интеллектуальных системах, являются оверлейные модели: векторные и сетевые, иногда встречаются и имитационные модели.

Существуют различные подходы к построению имитационной модели, однако они обладают рядом существенных недостатков при практическом применении. Технология нейронных сетей часто выступает как наиболее эффективный метод решения.

Модель обучаемого. Основная особенность адаптивной системы обучения – оптимизация процесса обучения. В качестве критерия эффективности рассматривается уровень знаний, умений и навыков обучаемого. Учёт в адаптивной системе обучения индивидуальных свойств обучаемого позволяет наиболее эффективно достигнуть поставленной цели обучения [7]. Адаптация к особенностям и потребностям выполняется путём формирования модели обучаемого (МО). Изучение работ [1, 4–7] позволило выделить ряд параметров (см. табл. 1), составляющих ядро модели обучаемого.

Таблица 1

№	Компоненты	Описание
1.	Начальные знания (а)	Общий уровень подготовки
2.	Психофизиологические характеристики (b)	Тип мышления, воспринимаемая форма знаний, способность к обучению
3.	Пользовательские предпочтения (с)	Предпочтения в работе с компьютерными системами
4.	Текущая работа с курсом (d)	Текущий уровень знаний, выполнение заданий, обращение за справочной информацией, время работы с курсом, пройденный материал
5.	Метод обучения (е)	Стратегия и метод обучения

№	Компоненты	Описание
6.	Отношение к системе (f)	Цели, мотивации, ожидание
7.	Преыстория работы (g)	Изученные темы, оценки, время изучения материала

Модель обучаемого обновляется в ходе обучения в соответствии с изменением отражаемых ею характеристик обучаемого.

Система управления учебной деятельностью. Предлагаемая система будет использовать два этапа построения курса обучения (рис. 1)



Рис. 1. Адаптация учебного материала
(МО – модель обучаемого, НС – искусственная нейронная сеть)

На первом этапе из МО извлекаются данные, необходимые для классификации текущей ситуации (например, результат тестирования, время изучения темы, метод решения задачи, количество обращений за справочной информацией). Из базы знаний выбирается соответствующее данной ситуации учебное воздействие (перейти к новому разделу, ознакомиться с дополнительным материалом, выполнить практическое задание, пройти тест и т.д.).

На втором этапе происходит формирование содержания учебного материала с учётом информации об индивидуальных характеристиках, хранящихся в МО (воспринимаемая форма представления знаний, текущий уровень знаний по предмету и т.д.). Такая процедура позволит повысить значение выбранного критерия эффективности, максимально быстро и полно донести до конкретного обучаемого основные принципы, понятия и правила изучаемой предметной области.

В качестве математического аппарата, решающего задачу классификации, выбрана многоуровневая нейронная сеть (НС) [2]. Основным преимуществом нейронных сетей является самообучение на предлагаемых приме-

рах и настройка в процессе работы [3]. Первоначально НС обучается на примерах, выбранными экспертами в области обучения (педагогами).

В основе построения НС лежит элементарный преобразователь, называемый искусственным нейроном (рис. 2) или просто нейроном по аналогии с его биологическим прототипом.

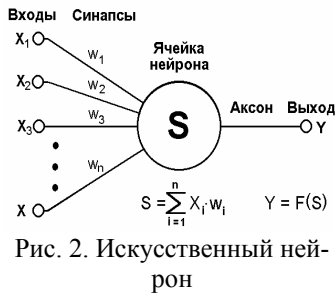


Рис. 2. Искусственный нейрон

Каждый синапс характеризуется величиной синаптической связи или ее весом w_i . Текущее состояние нейрона определяется, как взвешенная сумма его входов:

$$s = \sum_{i=1}^n x_i \cdot w_i \quad (1)$$

Выход нейрона есть функция его состояния:

$$y = f(s) \quad (2),$$

где f – активационная функция [2].

Проектирование НС. Для проведения

этой работы мы использовали Neural Analyzer, входящий в пакет Deductor 4.0. Структуру нейросети можно описать следующим образом. Нейросеть состоит из нескольких слоев: входной, внутренние (скрытые) и выходной слою. Входной слой реализует связь с входными данными, выходной – с выходными. Внутренних слоев может быть от одного и больше. В каждом слое содержится несколько нейронов.

Нейросеть способна имитировать какой-либо процесс. Любое изменение входов нейросети ведет к изменению ее выходов, причем выходы нейросети однозначно зависят от ее входов.

Рассмотрим НС в применении к задаче адаптивного обучения.

Смоделируем первый этап (рис. 2) – классификация текущей ситуации.

Пусть характеристики текущей работы с курсом представлены вектором

$$X(x_1, x_2, x_3, x_4) \quad (3),$$

где x_1 – время работы с курсом;

x_2 – количество пройденного теоретического материала;

x_3 – количество пройденного практического материала;

x_4 – результат тестирования.

На вход НС подадим набор параметров МО (3), на выходе получим код Y , определяющий режим обучения, соответствующую текущим значениям параметров МО (рис. 3).

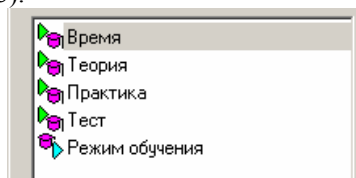


Рис. 3. Входы и выход сети

Работу сети можно записать в векторной форме:

$$Y=F(XW) \quad (4),$$

где Y – выход сети, X – вектор входов, W – вектор весов, $F(XW)$ – передаточная функция, преобразующая вход X в отклик сети Y .

Структура сети представлена на рис. 4. Во входном слое 4 нейрона, т.е. по одному на один вход (в обучающей выборке 4 столбца). Сделаем один скрытый слой с двумя нейронами. В выходном слое будет один нейрон, на выходе которого будет решение о выборе режима обучения.

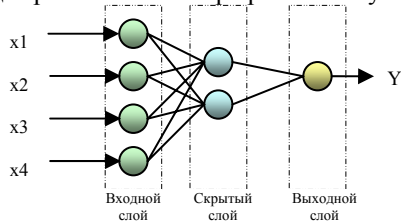


Рис. 4. Структура НС

Выберем алгоритм обучения сети – Resilent Propagation (эластичное распространение) с настройками по умолчанию. Обученная нейронная сеть позволяет в ходе эксперимента выяснить наиболее эффективное учебное воздействие. Это можно сделать, используя анализ «Что-Если» (рис. 5).

Поле	Тип	Значение	Минимум	Максимум	Колво
Входные					
Время	9.0	1		0	1
Теория	9.0	0,5		0,1	1
Практика	9.0	0,2		0,2	1
Тест	9.0	1		1	5
Выходные					
Режим обучения	9.0	4,50539586599342			

Рис. 5. Анализ «Что-Если»

После изменения в этой таблице входных полей система сама принимает решение об оптимальном режиме обучения.

По диаграмме, на которой отображается зависимость выходного поля от одного из входных полей при фиксированных значения остальных полей, можно определить на какой режим обучения наиболее вероятный для обучаемого с определёнными характеристиками. Например, по диаграмме на рис. 6 можно определить, что чем хуже результаты тестирования, тем больше вероятность начать обучение заново.

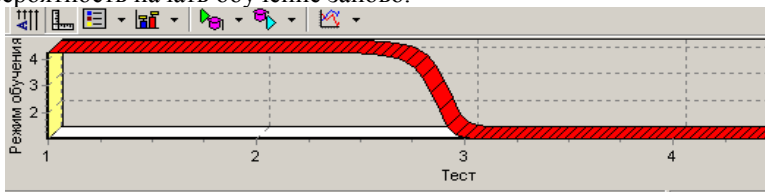


Рис. 6. Зависимость режима обучения от результатов тестирования

Качество построенной модели можно определить по таблице сопряженности, которая является одним из визуализаторов. Чем больше правильно классифицированных записей, тем лучше построенная модель.

Результаты и перспективы. Построенная нейросетевая модель студента позволяет решить задачу выбора оптимального учебного воздействия в адаптивной обучающей системе. Нарращивание параметров МО с учетом таблицы 1 и интеграция с информационной базой программного комплекса «Учебный процесс» позволит существенно улучшить процесс управления качеством обучения.

Таким образом, адаптивная система обучения предоставит возможность:

- получить прогноз, какие педагогические результаты наиболее вероятны (а какие практически невозможны) для учащегося с данной предысторией, с данным актуальным состоянием при воздействии на него имеющимися в распоряжении образовательными (педагогическими) технологиями;
- разработать рекомендации о том, какие предыстория, актуальная картина и педагогические воздействия должны быть, чтобы можно было рассчитывать с определенной уверенностью на заданный уровень учебных достижений и заданный педагогический эффект.

Литература:

1. Атанов Г.А. Предметное моделирование обучаемого // Актуальные проблемы педагогики и психологии. – Вып. 3. – Днепропетровск: Навчальна книга, 2000. – С.5-14.
2. Горбань А.Н. Нейронные сети на персональном компьютере. – Новосибирск: Наука. Сибирская издательская фирма РАН, 1996. – 276 с.
3. Лавров Є.А., Барченко Н.Л. Интеллектуальные обучающие системы на основе нейронных сетей // “Информатизация освіти України: стан, проблеми, перспективи”: Матер. другої міжнарод. наук.-практич. конф., 3-5 вересня 2003 р. – Херсон, 2003. – С. 80-82.
4. Растринин Л.А., Эрнштейн М.Х. Адаптивное обучение с моделью обучаемого. – Рига: Зинатне, 1988. – 160 с.
5. Тихомиров В.П., Солдаткин В.И., Лобачев С.Л. Виртуальная образовательная среда: предпосылки, принципы, организация / Международная Академия Открытого Образования. – М.: Издательство МЭСИ, 1999. – С. 164.
6. Brusilovsky, P.: Adaptive educational systems on the World Wide Web. In: Ayala, G. (ed.) Proc. of Workshop "Current Trends and Applications of Artificial Intelligence in Education" at the 4th World Congress on Expert Systems, Mexico City, Mexico, ITESM (1998) 9-16.
7. Brusilovsky, P.: Methods and techniques of adaptive hypermedia. User Modeling and User-Adapted Interaction 6, 2-3 (1996) 87-129.

МОДУЛЬНА ТЕХНОЛОГІЯ ВИВЧЕННЯ МЕТОДІВ ОБЧИСЛЕНЬ

О.П. Ліннік

м. Кривий Ріг, Криворізький коледж Національного авіаційного
університету
aplinnik@mail.com

У багатьох галузях людської діяльності широко використовуються математичні методи для вирішення найрізноманітніших задач науки, техніки й економіки. Значення цих методів істотно зросло в зв'язку з масовим застосуванням у всіх галузях народного господарства електронних обчислювальних машин (ЕОМ).

У статті запропоновано модульну технологію вивчення методів обчислень курсу вищої математики та інформатики для студентів заочного та дистанційного навчання інженерних спеціальностей.

Матеріал курсу поділяється на шість модулів:

- елементи теорії похибок;
- наближення (апроксимація) функцій;
- методи розв'язання систем лінійних рівнянь;
- чисельні методи розв'язання нелінійних рівнянь;
- методи мінімізації функцій;
- чисельне диференціювання;
- чисельне інтегрування;
- чисельні методи розв'язання диференціальних рівнянь.

Модуль являє собою логічно завершений розділ навчального матеріалу. Модуль об'єднує кілька мікромодулів (один мікромодуль – як правило, одне практичне заняття) залежно від обсягу матеріалу.

Кожен модуль починається з загальних положень, в яких формулюються теми розділу, базисні поняття, основні задачі, а також вимоги до теоретичних та практичних знань і вмінь студентів, якими вони повинні володіти після вивчення даного модуля.

Мікромодуль містить:

- теоретичну частину;
- практичну частину;
- тестові завдання.

В теоретичній частині викладено у стислій формі необхідний матеріал для опитування з розглядуваної теми (конспект лекції). До всіх тем подано посилання на літературу, що дасть можливість студентам у разі необхідності більш детально і ґрунтовно опанувати теоретичний матеріал.

Практична частина містить приклади розв'язання типових задач, які ілюструють теоретичний матеріал, а також вправи з відповідями для аудиторної і самостійної роботи студентів [1, 7].

Наприкінці мікромодуля вміщено індивідуальні тестові завдання для

контролю засвоєння студентами матеріалу даного розділу. Кожне завдання, як правило, складається з двох-трьох задач. На кожному практичному занятті студент здає індивідуальне завдання попереднього мікромодуля, виконане у письмовій формі.

Після вивчення модуля відбувається захист виконаних робіт.

Лекційний матеріал викладається традиційним чином.

На практичних заняттях студенти оволодівають основними методами та способами розв'язування математичних задач, отримують роз'яснення теоретичних положень курсу чисельних методів [2]. Тематика практичних занять пов'язана зі змістом лекцій.

Практичні заняття проводяться за схемою:

- перевірка теоретичного матеріалу (згідно з планом);
- перевірка домашніх завдань, однакових для всіх студентів; пояснення найбільш складних моментів;
- розв'язання типових задач за темою модуля;
- самостійна робота (згідно з планом – останнє заняття модуля).

Враховуючи різну кількість годин, які відводяться за планом для вивчення чисельних методів студентам різних спеціальностей, викладач (лектор) може коригувати вміст модулів, кількість тестових завдань, які студент повинен виконати протягом семестру, а також форми семестрового контролю знань студентів (самостійні та контрольну роботу, колоквиум, тощо). Про ці заходи викладач повідомляє студентів на початку семестру.

Відзначимо, що при належній успішності протягом семестру підсумкова рейтингова оцінка може бути зарахована як залікова, тому маємо надію, що це буде спонукати студентів до підвищення успішності протягом усього навчання.

Програма курсу «Чисельні методи» складена в обсязі, необхідному для вивчення загальнонаукових, загальноінженерних і спеціальних дисциплін і розвитку навичок, необхідних для застосування математичних методів у практиці роботи інженера.

Робота студента над навчальним матеріалом складається з таких елементів: вивчення матеріалу по підручниках, розв'язування задач, відповіді на питання для самоперевірки, виконання лабораторних робіт, слухання лекцій та реалізації методів на персональному комп'ютері, здача усних заліків [3].

Студенти повинні набути уміння застосовувати основні методи дослідження та розв'язання інженерних задач, заміни їх простішими реалізаціями на електронно-обчислювальних машинах, будувати математичні моделі, ставити математичні задачі, вибирати відповідні математичні методи та алгоритми для розв'язування задач, застосовувати якісні методи дослідження, виробляти практичні рекомендації, самостійно поповнювати математичні знання.

Процес розв'язання при цьому має такі етапи:

- постановка задачі;

- побудова математичної моделі;
- вибір чисельного методу розв'язання та визначення початкових даних;
- складання алгоритму та його запис мовою програмування;
- виконання програми на ЕОМ;
- аналіз отриманих результатів.

Література:

1. Бахвалов Н.С. Численные методы. – М.: Наука, 1975. – 630 с.
2. Васильев Ф.П. Лекции по методам решения оптимальных задач. – М.: Изд-во Моск. ун-та, 1974. – 374 с.
3. Волков Е.А. Численные методы. – М.: Наука, 1982. – 254 с.
4. Демидович Б.П., Марон И.А. Основы вычислительной математики. – М.: Наука, 1966. – 364 с.
5. Калиткин Н.Н. Численные методы. – М.: Наука, 1978. – 512 с.
6. Хемминг Р.В. Численные методы. – М.: Наука, 1968.
7. Ракитник В.И., Первушин В.Е. Практическое руководство по методам вычислений. – М.: Высшая школа, 1998.

‘HELLO WORLD’ ЯК ДИДАКТИЧНИЙ АТАВІЗМ

Р.В. Ліхачов

м. Кривий Ріг, Криворізький металургійний факультет
Національної металургійної академії України

Органи, що втратили свої основні функції внаслідок процесу еволюції, але не відмерли повністю, називають атавізмами. Звичайно ж, мова йтиме не про апендикс і медично-етичний аспект його видалення.

Так склалося історично, що більшість підручників, методичних вказівок чи інших посібників, пов'язаних з вивченням мов програмування, починаються з канонічного ‘Hello World’.

На перший погляд здається, що проблем немає, що цей звичайний приклад сприймається лише як перший ступінь пізнання складного і цікавого шляху майбутнього програміста. Проте чи це так насправді?

Переглянувши велику кількість підручників з різних мов програмування та оцінивши загальний підхід авторів до викладення матеріалу, легко впевнитись, що проблема, про яку йде мова, не висмоктана з пальця, а є нагальною і потребує вирішення.

Звісно ж, ніхто не закликає спалити старі підручники чи заборонити використання стійкого іноземного фразеологізму ‘Hello World’ у навчальній літературі. Адже проблема не у фразі, з якої починаються всі перші приклади у книжках (до речі, це не лише привітання світові, а й ‘Hello from console’, ‘Welcome to C++’, ‘My first string’ тощо), проблема у використанні примітивних, нічого не пояснюючих прикладів.

Давайте подивимось, що демонструє згадана вище конструкція. Частіше за все – це просте використання консольного виводу за допомогою вбогого набору найпростіших функцій. Весь приклад часто складається з трьох-чотирьох строк, в залежності від мови програмування, що несуть в собі: коментар, головний клас або оператори початку й кінця програми, функцію видачі на екран рядка символів (іноді з використанням змінної, що несе в собі основний текст) та функцію затримки чи очікування натискання будь-якої клавіші.

Зустрічаються видання, де коментарям присвячено окрему главу замість кількох абзаців, чого буває досить для нормальної людини. Інколи складається враження, що деякі автори захищали дисертацію, присвячену темі повсякденного використання коментарів у коді програм і намагаються якнайкраще познайомити читача з усіма аспектами коментування коду.

На перший погляд, згадана вище проблема не виглядає такою вже страшною, адже, опрацюовуючи новий матеріал, людина працює не лише з друкованими виданнями, а й з викладачем, що може звернути її увагу на більш важливі речі, ніж необхідність щоразу вітати світ. Так, дійсно викладачі повинні допомогти починаючому програмісту зрозуміти, що є важли-

вим, а що другорядним, та мають надати нормальні приклади використання мов програмування. Та хіба ж так відбувається насправді?

Нажаль, у повсякденному житті ситуація завжди одна – приклади програм, що їх дають викладачі як у конспектах лекцій, так і в методичних вказівках до виконання практичних робіт, дуже рідко серйозно відрізняються від ‘Hello World’.

Давайте поглянемо, що пропонують своїм слухачам викладачі мов програмування. По-перше, це робота зі строками (прийом даних зовні, конкатенація, підрахунок кількості символів чи слів, видача на екран), по-друге, прості математичні дії (обчислення декількох формул з подальшим друком результатів), по-третє, робота з масивами даних (заповнення одно- чи двовимірної матриці випадковими числами, сортування кількома простими методами, перестановка рядків або стовпчиків в залежності від якихось умов), далі – робота з файлами, частіше з текстовими (відкриття та закриття, запис та зчитування даних, зміна розташування файлу чи зміна атрибутів), а наприкінці – основи роботи із графікою (примітивні графіки функцій чи різнокольорові геометричні фігури).

Хіба можна вважати, що для ґрунтовного вивчення матеріалу достатньо таких простих задач? Вважати можна, проте їх все одно недостатньо.

Хвала тим вищим навчальним закладам, де програмуванню і мовам програмування навчають окремо. А саме: формують мислення майбутнього програміста, займаючись математикою та алгоритмізацією, та дають можливість застосувати своє мислення в різний спосіб – вивчаючи три-чотири мови програмування одночасно. Такий підхід вже близький до ідеального, проте не до кінця.

Коли я зустрічаю викладачів, що намагаються дати студентам якомога важкіші задачі, часто комплексні, такі, що вимагають не лише філігранного володіння мовою, як інструментом, а й оригінального підходу до вирішення поставленої проблеми, я вклоняюся їм, бо вони *майже* досягли істини! Так – МАЙЖЕ!

Де ж знаходиться те маленьке *щось*, що перетворює МАЙЖЕ в НАСПРАВДІ?

Згадаємо лише, як виглядають програми, написані студентами під час практичних занять (це те, що викладачі бачать щодня, проте чомусь не помічають). Ось приклад: потрібно отримати від користувача два числа, провести над ними прості математичні дії та продемонструвати результат. Задача проста, а тому й не цікава зовсім. Що ж ми отримуємо на виході?

Випадок перший (*клінічний*): чорний екран, на ньому куцо промигує курсор, що натякає користувачу розпочати активні дії, проте які не – натякає. Лише досвідчений знавець студентського середовища, професіонал з програмування та психології, а головне – автор методичного посібника, в якому згадується подібна задача, одразу зрозуміє (напевно, інтуїтивно), що потрібно ввести якесь число, а можливо, й не одне. Результат шокуючий,

після вводу першого числа програма підмигує курсором, запрошуючи ще одну порцію вхідних даних, певна річ ми їх надаємо... І, О ЧУДО, вона (якщо не мигнула екраном і завершила роботу, повернувшись до консолі чи редактору коду) показує нам якесь число!

Можна не посміхатись, адже у 99% випадків це саме та картина, що її бачать викладачі на практичних заняттях з мов програмування.

Випадок другий (*неймовірний*): чорний екран, на ньому напис “Введіть число”, після вводу – інший “Введіть ще”, вінчає обстановку напис “Відповідь” та число, що найчастіше буває впритул приліплене до нього (знову ж згадаємо можливість миттєвого завершення програми, проте з ймовірністю 50%). Сльози радості у викладачів та студентів, оплески, залік *автоматом*.

Такий випадок можливий десь у 20 відсотках. І нарешті щось, що мені вдалося зустріти лише один раз.

Випадок третій (*сюрреалістичний*): чорний екран, на ньому напис “Введіть ціле число у діапазоні від 0 до 65535”, після вводу – “Введіть ще одне число з того ж діапазону, що б воно не дорівнювало 0”, наприкінці на екрані з’являється не лише результат, а й формула за якою відбувався підрахунок, а ще напис “Натисніть клавішу ‘ESC’ для завершення роботи програми, чи ‘ENTER’ для повторного використання”. Немає мови про те, що числа обов’язково перевіряються після їх вводу, та в разі помилки користувачу надається можливість повторно задати вхідні дані.

Мабуть, є над чим замислитись. Прості задачі, такі, як у наведеному прикладі, особливо ті, що розраховані на вивчення лише математичного апарату різноманітних мов програмування, потрібно раз і на завжди викреслити з усіх методик навчання, та викинути на сміття методичні посібники, де розглядаються подібні задачі. Вони не цікаві, а це означає, що студенти ними займатись не будуть, або займатимуться ними під тиском. Адже вони розуміють, що, почавши вивчати мову програмування високого рівня, не асемблер, а щось набагато розвиненіше, можна розраховувати на наявність нормальних засобів роботи як зі строковими даними, так і з різноманітними математичними функціями.

Старий Сірко у п’єсі “За двома зайцями” казав: “Простота – не гріх!”, з чим у даному випадку можна дуже посперечатись. Прості задачі ведуть до спрощення мислення, до втрати цікавості, до *шедеврів*, змальованих у перших двох прикладах вирішення студентами занадто легких завдань.

Прості однотипні задачі – це навіть не верхівка айсбергу, з яким ми стикаємось при навчанні (до того ж не лише мов програмування), це лише одна сніжинка, на самій вищій його точці. Проблема заховалась у відсутності естетичного виховання у вищій школі.

Можна заперечувати, що таке виховання має давати школа, батьки чи навіть дитячий садочок, а вищий навчальний заклад не має необхідної кількості годин навіть для викладення у нормальному обсязі профільних дисциплін. Так, але мова не йде про навчання студентів гри на скрипці, чи баль-

ним танцям. Проте ніхто не заперечуватиме, що музику Баха, Брамса, Вівальді, Гріга, Моцарта, Прокоф'єва, Хачатуряна або Чайковського потрібно хоча б раз у житті послухати не лише у виконанні мобільного телефону!

В стародавній Японії вважали за норму, коли людина розпізнає понад декілька мільйонів кольорів, складає вірші, володіє музичними інструментами, розуміється на філософії, малює чи займається каліграфією, та плаче від насолоди, коли спостерігає, як квітне сакура.

Давайте повернемося до того, як студенти виконують практичні завдання, як виглядають їх програми, як вони працюють. А може ні, краще поглянемо на себе, які ми робимо програми, як вони виглядають, як працюють. За прикладами далеко ходити не потрібно – достатньо уважно передивитись матеріал, представлений на компакт-диску, який додається до матеріалів конференції.

Тож як можна знайти вихід із ситуації, що склалася? Треба починати з себе, просто необхідно зробити ті важкі, але потрібні кроки до виправлення власних помилок, до перемоги над своєю лінію.

Перше, що має бути зробленим – це відмова від примітивних прикладів і завдань. Звісно ж, не всі цим грішать, проте практика показує, що більшість.

Завдання з курсу мов програмування повинно обов'язково складатись з декількох частин: це мають бути задачі, що вимагають знання базових алгоритмів, завдання, що охоплювали б сферу інтересів студента чи були б пов'язані з профілем ВНЗ, а краще і те і інше одночасно. Якщо ж завдання розраховано на виконання кількома студентами, тобто є колективним, то воно взагалі безцінне! Адже, повертаючись до теми коментарів у програмному кодї, лише коли проект великий і ведуть його кілька осіб – виникає потреба використання не лише коментарів, а й нормального стилю написання, що робить код більш зрозумілим.

А найголовніше, чому потрібно навчити майбутнього програміста – це нормальний підхід до виконання роботи і зручний якісний інтерфейс.

Знову ж таки можна мені заперечити, мовляв над програмами у справжньому житті завжди працюють кілька не лише людей, а кілька різних груп програмістів, кожна з яких відповідає за свою конкретну частину роботи. І навчати студентів розробляти інтерфейс до своїх програм чи взагалі замислюватись над цілісністю проекту – це марнотратство. Так, дійсно, розподіл обов'язків серед програмістів, та розділення груп по напрямкам відбувається лише в достатньо великих фірмах, що спеціалізуються на розробці комерційного програмного забезпечення. А чи не вважаєте ви, що всі наші студенти підуть працювати саме в такі організації, яких до речі не так вже й багато, а конкуренція доволі сильна?

Ні, більшість піде штатним програмістом в організацію, що займається приватним бізнесом і не пов'язана з випуском програм, або на підприємстві, де потрібна людина, що зможе налагодити дві чи три програми, які ви-

користовуються там постійно. А можливо і доведеться написати кілька програм, що полегшать роботу того чи іншого відділку, чи автоматизують власне робоче місце.

Саме так виглядає реальність майбутнього програміста, а тому вміння правильно розробити і довести до кінцевого користувача власний проект – одне з найважливіших. А якщо програмою, що буде народжена цим проектом, зможуть нормально і зручно користуватись люди, то можна вважати, що навчання не пройшло задарма.

Будь-яка робота над майбутньою програмою повинна починатись з чистого аркушу паперу, з всебічного розгляду необхідних функцій та елементів інтерфейсу, з відсікання циклічних дій чи непотрібних частин, і все це повинно бути замальовано та прораховано з точністю виготовлення космічного човна.

Розробка інтерфейсу користувача – це не непотріб, це необхідність! Якщо правильно навчити студента формуванню якісного інтерфейсу, пояснити, яким законам підпорядковані його елементи, як вони сприймаються кінцевим користувачем, то з'явиться нарешті цікавість в очах, програма почне грати новими фарбами, а у майбутнього програміста суттєво виросте самоповага та бажання *творити*.

Творчий підхід до виконання роботи, створення естетичного образу майбутнього продукту надає не лише натхнення для наступних проектів, а й формує свідомість, заохочує працювати з цікавістю.

Зрозуміло, що мова йде не про революцію в системі навчання, просто крапля творчого підходу до рутинної справи може зробити неймовірні перетворення, коли нецікаве і нудне одразу виявиться захоплюючим і веселим. Якщо відноситись до майбутнього програміста, як до митця, якщо дати йому цікаві задачі – можна перевернути гори!

З точки зору практичної психології, майбутні програмісти – це дуже складні в повсякденному спілкуванні люди, це індивідууми, що не виносять, коли на них тиснуть чи намагаються заохотити до нецікавої справи. Часто відлюдкуваті молодики із дуже специфічним, навіть інколи цинічним світосприйняттям.

Дуже складним є аспект керування групою програмістів, виконуючих спільну роботу: занадто мала кількість природжених керівників творчих груп (в середній групі з 15 чоловіків, з задатками до керування, може виявитись іноді лише один такий лідер). Звісно ж, керувати складною групою можна навчитись, постійно практикуючи та вдосконалюючи техніку роботи. Проблема полягає лише в тому, що людина зі здатністю до програмування, сама часто буває непоганим лідером, тому робота з групою програмістів ускладнюється саме необхідністю бути лідером серед лідерів.

І ось саме тут в нагоді нам стане нетлінний образ нашого першого радянського космонавта! Звісно ж, не доведеться розвішувати в комп'ютерних класах його фотографії – просто доведеться ним стати для своїх студентів.

Потрібно бути людиною, яка побувала *за межами* звичайного, за кордоном рутинного. Потрібно бути тим світлим образом, зрівнятись з яким хотітиме і початківець-першокурсник, і свідомий випускник.

Скажете це неможливо? Можливо, якщо зацікавлювати людей не програмами обчислення суми двох змінних, а завданнями, що на перший погляд не мають рішення, шокувати оригінальним підходом і знанням історії розвитку питання, що розглядається. Потрібно вчити студентів мислити самостійно, знаходити помилки і шляхи подолання труднощів при вивченні мов програмування.

Навчити студентів замислюватись над виглядом своїх програм, на власному прикладі показати, як розплутувати важкі ситуації, надати цікаві завдання та запропонувати чистий аркуш паперу – ось і все, що потрібно, щоб ‘Hello World’ перестав бути “прикладом номер один у світі”!

ОРГАНІЗАЦІЯ САМОКОНТРОЛЮ ПРИ НАВЧАННІ ІНФОРМАТИКИ

І.М. Лукаш

м. Чернігів, Чернігівський державний педагогічний університет
імені Т.Г. Шевченка
lukash@cg.ukrtel.net

Контроль та самоконтроль знань, умінь та навичок завжди займали важливе місце у навчальному процесі, але співвідношення між контролем та самоконтролем змінювалось протягом часу і залежало від дидактичної системи викладання, що застосовувалась у певний період. Утвердження парадигми масового навчання передбачало застосування книжково-фронтального методу навчання, жорстку регламентацію (навчальний розклад, обов'язковий контроль з боку вчителя тощо), що призвело до втрати індивідуального підходу в освіті та її варіативності, ігноруванню індивідуальних здібностей. В більшості випадків оцінювання учнів в ході педагогічного контролю здійснювалося за інерцією, виходячи з попередніх досягнень або невдач учня. Замість формування інтересу до знань в учнів формувався інтерес до оцінок. Вважається, що саме стимулювання самоконтролю, самоперевірки з боку учня сприяє зростанню впевненості у своїх можливостях, стимулюванню активізації діяльності на уроці, вихованню вибагливості до себе, формуванню інтересу до знань. А це, безперечно, відповідає вимогам до навчання з точки зору сучасної гуманістичної парадигми.

Серед вимог до організації процесу навчання, які висуває втілення гуманістично-орієнтованого підходу в сучасну освіту, і які сприяли б реалізації самоконтролю учнів, доцільно виділити діагностичність, комунікативність та активність засобів навчання. Певною мірою цим вимогам задовольняють такі засоби навчання, як електронні підручники та контролюючі програми. Електронні підручники продвинутого, визначного та перспективно-дослідницького рівнів [1] передбачають, окрім мультимедійних засобів подання знань, наявність оберненого зв'язку, який забезпечує навчальний діалог та моніторинг успішності навчання учнів. Якщо електронні підручники дозволяють організувати самоконтроль учнів на етапі вивчення нового навчального матеріалу, а контролюючі програми – на етапі контролю та оцінювання навчальних досягнень учнів, то на тренувальному етапі навчання, у процесі якого відбувається формування певних умінь, доцільним є застосування робочих зошитів. Реалізації самоконтролю при роботі з електронним зошитом в більшості випадків, як і при роботі із друкованим зошитом, забезпечується можливістю порівняння результату виконання завдання зі зразком або правильною відповіддю.

В своїй практиці вчителі все частіше замість друкованих робочих зошитів застосовують електронні. Так, наприклад, при вивченні Excel в [1]

рекомендується розміщувати умови тренувальних задач в середовищі їх розв'язування. Але передбачається, що вчитель повинен “перевірити” зошит, а учень – “виправити” помилки. При роботі з таким електронним робочим зошитом повідомлення від програми спонукають учня до контролю за своїми поточними діями, до пошуку помилки. Сам програмний засіб може надавати повідомлення про наявність синтаксичних помилок, відсутність зв'язків між певними об'єктами тощо. Вчитель певною мірою може спланувати виникнення таких повідомлень завдяки формулюванню умови задачі, в якій “упущені” або наведені зайві дані. Наприклад, в умові завдання, пов'язаного зі створенням бази даних та виконанням запитів, навмисне випустити назву поля, за яким повинен здійснюватися зв'язок між таблицями.

Приклад 1. Створити базу даних, що складається з двох таблиць наведеної структури. Структура таблиці 1 (робітники): табельний номер, код підприємства, професія, посада, заробітна плата, дата прийняття на роботу. Структура таблиці 2 (підприємства): прізвище та ініціали директора, назва підприємства, телефон, адреса. До бази даних виконати запит: отримати список всіх програмістів, що працюють на підприємстві Bel-IBM.

Якщо учень не виявить помилки відсутності необхідного ключового поля “код підприємства” в структурі другої таблиці в ході створення таблиць бази даних, то реалізація наведеного запиту буде вимагати наявності зв'язку між двома таблицями через додаткове поле, про що будуть повідомляти певні повідомлення від СКБД.

Окрім цього, за допомогою VBA – вбудованого в MS Office засобу програмування – надається можливість як для автоматичної перевірки результату розв'язування задачі, так і для формулювання з боку вчителя певних повідомлень стосовно можливих логічних помилок учня. Реалізацію цього підходу можна здійснити, наприклад, в Excel, через введення до електронного робочого зошиту управляючої кнопки з написом “Дізнайся оцінку” або “Перевір розв'язок”, на яку підключений написаний вчителем на Visual Basic певний контрольний макрос.

Приклад 2. Умова. Існує мережа із 6 магазинів, в кожному з яких продають однакові товари восьми найменувань. Кількість товарів в кожному з магазинів наведена в таблиці 1. Вартість кожного товару наводиться в таблиці 2. Завдання. За даними таблиці 1 та таблиці 2 заповнити таблицю 3, в якій підрахувати, на яку суму є товару в кожному з магазинів. Підрахувати, на яку суму є товару в усіх шести магазинах. Результат підрахунку записати до комірки F30.

Приклад оформлення фрагменту електронного зошита для розв'язування задачі прикладу 2 наводиться на рис. 1. Фрагмент спрощеного програмного коду, що приєднується до кнопки, наводиться нижче.

```
Sub notation()  
If Range("F30") = 0 Then  
    Range("J30").Value = "Розв'яжуйте..."  
Else
```

```

If Range("F30") <> 206250 Then
    Range("J30").Value = "Помилка"
Else
    If Range("F30") = 206250 Then
        Range("J30").Value = "Правильно"
    End If
End Sub

```

	A	B	C	D	E	F	G	H	I	J	K
14			тов.1	тов.2	тов.3	тов.4	тов.5	тов.6	тов.7	тов.8	
15	Табл.1	магазин 1	45	55	65	75	85	95	105	115	
16		магазин 2	35	40	45	50	55	60	65	70	
17		магазин 3	25	25	25	25	25	25	25	25	
18		магазин 4	15	20	25	30	35	40	45	50	
19		магазин 5	5	15	25	35	45	55	65	75	
20		магазин 6	10	20	30	40	50	60	70	80	
21											
22		Табл. 2			Табл. 3						
23			ціна								
24		тов. 1	10								
25		тов. 2	20								
26		тов. 3	30								
27		тов. 4	40								
28		тов. 5	50								
29		тов. 6	60								
30		тов. 7	70	Сума:				Оцінка:			
31		тов. 8	80								

Рис. 1

Отже, запропоноване в роботі доповнення електронного зошита елементами управління та програмними модулями дозволяє в повнішій мірі реалізувати методи самоконтролю учнів, що підвищує доцільність застосування електронних зошитів при навчанні інформатики.

Література:

1. Кузьмінська О.Г., Власенко Т.І. Електронний зошит для роботи з Excel // Комп'ютер у школі та сім'ї. – 2002. – №6. – С. 21-22.
2. Прохоренко І.Ф., Биков В.Ю., Раков С.А. До питання інформатизації вищих педагогічних навчальних закладів // Комп'ютер у школі та сім'ї. – 2002. – №4. – С. 8-13.

НАВЧАЛЬНІ ПРОЕКТИ ДЛЯ МАЙБУТНІХ ВЧИТЕЛІВ

І.В. Лупан, В.В. Копотій

м. Кіровоград, Кіровоградський державний педагогічний університет
імені Володимира Винниченка
ilupan@kspu.kr.ua

Дослідження, присвячені питанням удосконалення процесу навчання, пошуку нових, більш ефективних форм та методів навчання, розробці нових освітніх парадигм тощо, ніколи не втрачають актуальності. Тема даного повідомлення обумовлюється проблемами, що виникають при вивченні інформатики на неспеціальних факультетах педагогічного вузу. На наш погляд, при вивченні інформатики доречно використовувати матеріали профілюючих для відповідної спеціальності дисциплін. Водночас не можна від кожного викладача інформатики вимагати обізнаності з усіх дисциплін, щоб використовувати на кожному факультеті спеціальний набір завдань. Розв'язати дане протиріччя можна застосувавши елементи технології навчальних проектів. Дана технологія в останній час набуває усе більшої популярності: її покладено в основу програми “Intel®. Навчання для майбутнього” [1] (розпочато співробітництво МОН України з цією програмою), проводяться тренінги та дистанційні курси для вчителів, є публікації з даної тематики у періодичних виданнях, методичні рекомендації та посібники [2, 3, 5]. Але здебільшого згадані матеріали стосуються навчання у шкільних закладах. Наш досвід показує, що, принаймні, елементи технології навчальних проектів можна застосовувати також на молодших курсах вузів. Особливо це доцільно при підготовці майбутніх вчителів: ще на студентській лаві вони можуть ознайомитися з методикою організації та впровадження методу навчальних проектів, побачити їх так би мовити “із середини”.

Під *навчальним проектом* будемо розуміти організаційну форму роботи, яка орієнтована на засвоєння навчальної теми або навчального розділу і становить частину стандартного навчального предмета або кількох предметів. У вузі навчальний проект можна розглядати як спільну навчально-пізнавальну, дослідницьку або творчу діяльність студентів (індивідуальну, парну, групову), що має спільну мету, однакові методи і способи діяльності, спрямовані на досягнення спільного реального результату, потрібного для вирішення деякої вагомості проблеми [2].

Метод навчальних проектів завжди передбачає розв'язування певної значущої для студентів проблеми, яка потребує, з одного боку, використання різноманітних методів, засобів навчання, а з іншого – інтегрування знань, умінь із різних галузей науки, техніки, технології та творчих галузей. Результати виконаних навчальних проектів повинні бути “відчутними”, тобто, якщо це теоретична проблема, то слід отримати конкретне її вирішення, якщо практична – конкретний результат, готовий до впровадження [5]. До

того ж специфіка навчальних проектів полягає у тому, що вони за своєю суттю є міжпредметними. Навіть якщо проект виконується в рамках однієї навчальної дисципліни, робота над ним вимагає застосування знань з інших галузей та зв'язку з реальним життям.

Відзначимо, що існує низка ознак для класифікації навчальних проектів. Це домінуючий в проекті метод (дослідницький, творчий, рольово-ігровий, ознайомлювально-орієнтувальний тощо), характер координації проекту (безпосередній, прихований), характер контактів (серед учнів одного класу, школи, міста, регіону, країни, різних країн світу), тривалість проекту та кількість учасників [6].

Експериментальне навчання здійснювалося нами на спеціальності “Фізика” фізико-математичного факультету, але його легко адаптувати до особливостей будь-якої педагогічної спеціальності.

Студентам пропонувалося розробити навчальний проект “Фізика сьогодні”. Найдоцільнішим, на нашу думку, застосування навчальних проектів є при вивченні тем “Комп’ютерні мережі”, “Мова HTML”, “Системи Multimedia” (Power Point). Стосовно інформатики цей матеріал є для студентів цікавим і відносно легким (адже тут не доводиться розв’язувати задач фізичного чи математичного змісту). Але, можливо саме через уявну легкість, інколи студенти приділяють даним темам менше уваги, і в результаті їхні знання не досягають бажаної глибини, а уміння – досконалості. Застосування методу навчальних проектів кардинально змінює ситуацію: виконуючи проект, студенти опановують пропоновані засоби майже на рівні професіоналів, значно підвищується мотивація навчання, самостійність і критичне ставлення до своїх знань та умінь.

Метою проекту у нашому випадку було формування у студентів теоретичних знань та практичних навичок роботи у глобальній комп’ютерній мережі Internet, у середовищі програми Power Point, опанування ними мови HTML. Студенти також повинні були навчитися створювати веб-сайти та мультимедійні презентації. В цілому, робота над проектом передбачала залучення знань студентів з двох навчальних дисциплін: інформатики та фізики.

Постановка цілей та завдань проекту і контроль за його реалізацією покладається на викладача інформатики (а, якщо можливо, то й на викладача фізики). Тобто координація здійснюється безпосередньо. Навчання під час роботи над проектом відбувається звичайним порядком у вигляді лекційних та лабораторних занять згідно робочих програм та розкладу, однак на початку вивчення даних тем студенти отримують завдання для самостійного виконання, яке по суті є програмою індивідуального довготривалого (місяць-півтора) навчального проекту ознайомлювально-орієнтувального або інформаційного характеру. Особливістю інформаційного проекту є те, що він, як зазначає Є.С. Полат, спрямований на пошук інформації про об’єкт або явище з наступним ознайомленням широкої аудиторії з цією інформаці-

сю [6].

Структура подібного проекту може бути наступною:

- визначення мети пошукової діяльності та постановка проблеми (виконується викладачем);
- пошук інформації за різноманітними джерелами: літературні, засоби ЗМІ, бази даних, включаючи електронні, інтерв'ю, анкетування та ін., залежно від специфіки факультету (цей та наступні етапи передбачає самостійну діяльність студентів);
- обробка інформації (аналіз, співставлення з відомими фактами, синтез, оцінювання, аргументовані висновки) та формулювання можливих способів вирішення поставленої проблеми;
- підготовка звіту (у формі веб-публікації та презентації) та обговорення можливого застосування отриманих результатів;
- захист проекту.

Кожний студент отримує таке завдання:

1. Обрати тему, за якою буде здійснюватися пошукова діяльність.

Тематику може запропонувати викладач або самі студенти. Як правило, сильні, добре мотивовані до навчання студенти мають певні інтереси у предметній галузі і здатні визначитися із темою самостійно. Слабкі ж студенти потребують сторонньої допомоги. Єдиним обмеженням на тематику проекту є основна фахова дисципліна (у нашому випадку це була фізика).

2. Здійснити пошук інформації за різноманітними джерелами:

- вивчити вибрану тему за доступною спеціальною літературою (підручниками, науково-популярними журналами тощо);
 - ознайомитися з термінологією: з'ясувати, до якого розділу базової дисципліни дана тема має безпосереднє відношення, хто з вчених досліджував дану тематику; сформулювати запити для пошуку матеріалів в Internet;
 - за допомогою термінологічного або електронного словника, наприклад, LINGVO, отримати англomовні еквіваленти термінів для можливого пошуку на англomовних сайтах. Побудувати запити англійською мовою.
3. У мережі Internet знайти сайти, присвячені обраній тематиці, оформити звіт про них із вказанням URL-адрес та коротких анотацій. Скопіювати або скачати матеріали, необхідні для виконання наступних пунктів завдання.
4. Проаналізувати зібрану інформацію, оцінити, зробити висновки.
5. Підготувати звіт:
- розробити власний веб-сайт ієрархічної структури, присвячений обраній тематиці, з використанням матеріалів з Internet;
 - розробити презентацію нелінійної структури.

Отже на першому, підготовчому, етапі студентам оголошується обсяг робіт, які слід виконати для отримання заліку за перелічені вище теми курсу інформатики.

На лабораторних заняттях студенти знайомляться з інструментарієм відповідних прикладних програмних засобів, опановують прийоми роботи з ними, відповідають на контрольні запитання, виконують завдання, вказані у лабораторній роботі і т. п. – усе, як і завжди. Але тепер вони роблять це свідомо і не тільки для отримання оцінки за лабораторну роботу, а й для виконання власного проекту. Поступово творча атмосфера охоплює усіх.

Спочатку згідно навчальної програми вивчаються прийоми роботи у мережі. Тематика лабораторних робіт така: “Електронна пошта” – 2 години, “Internet: адреси ресурсів, способи збереження даних” – 2 години, “Пошукові системи” – 4 години. Пошук в Internet здійснювався за тематикою студентських проектів. Для завершення пошуку у позаурочний час додатково виділялося від 2 до 6 годин.

У проекті “Фізика сьогодні” діапазон обраних студентами тем виявився досить широким. Частина студентів зупинилася на темах із шкільного курсу фізики: “Закон Архімеда”, “Планети Сонячної системи” і тому подібне. Вони знайшли багато цікавого, починаючи від портретів вчених, біографічних відомостей про них, шкільних дистанційних курсів, електронних підручників, анімаційних дослідів, фото галерей NASA тощо, але потім були розчаровані тим, що знайдені матеріали, безперечно важливі у викладанні, містять для них самих мало нового.

Інша ж частина студентів зрозуміла поставлене завдання більш широко і не обмежилася переліком тем шкільного курсу. Вони представили теми: “Торсійні поля”, “Телескопи”, “Вітрова енергетика” тощо. Працюючи над цими проектами, студенти, крім відомих фактів, знайшли також нові матеріали, опубліковані та оприлюднені лише недавно.

З вищезазначеного ясно, що робота над проектом дозволяє кожному студенту працювати з матеріалом, який відповідає його власному рівню підготовки з фахової дисципліни і водночас є цікавим та зрозумілим. Але це не знижує успішності виконання завдання з інформатики. Тобто цілком органічно, начебто самі собою, запроваджуються індивідуалізація та диференціація навчання.

На вивчення мови HTML та програми Power Point нами було відведено по 3 лабораторні заняття. Перші два – навчальні: на них розглядаються окремі прийоми роботи, створюються зразки лінійних та нелінійних веб-сайтів та презентацій, опановуються прийоми роботи з гіперпосиланнями та кнопками управління. Під час виконання лабораторних робіт студенти розробляють сценарії веб-сайтів та презентацій за тематикою власних проектів. Кожний сценарій – індивідуальний, але він повинен відповідати деяким обов’язковим технічним вимогам, які є переліком засвоєваних умінь з інформатики. Це, по-перше, нелінійність, тобто сайт повинен складатися із декількох веб-сторінок, а презентація – із декількох слайдів, порядок експонування яких можна регулювати кнопками або гіперпосиланнями. По-друге, обов’язковим є використання графіки та анімаційних ефектів. У веб-

сторінці обов'язкове використання списків та біжучих рядків.

На третьому занятті проводився захист проектів шляхом демонстрації веб-сторінок та презентацій. Тут дуже продуктивною виявилася ідея, скопійована нами в однієї з колег, – запропонувати оцінювання робіт одногрупників самим студентам. Причому оцінки потрібно було обґрунтувати – виділити критерії, наприклад, оцінювати зміст, глибину подання матеріалу, естетичний вигляд, відповідність технічним вимогам. “Експерту”, який усі або майже усі роботи оцінив однаково, знижувалася оцінка за оцінювання. Кінцева оцінка визначалася за середнім арифметичним оцінок “експертів”. “Штрафи” додали критичності тим, у кого її бракувало. А перспектива бути оціненим товаришами підвищила відповідальність при виконанні завдання.

У результаті організації навчально-пізнавальної діяльності на заняттях за допомогою метода навчальних проектів студенти не тільки набувають визначених програмою знань, умінь та навичок з інформатики, а ще починають розуміти зв'язки між отриманими знаннями з різних навчальних дисциплін. Залучення методу навчальних проектів розвиває вміння самостійно вирішувати різноманітні проблеми; формувати навички критичного та творчого мислення у практичній діяльності, наближеній до реального життя.

Література:

1. <http://www.iteach.com.ua> / Програма “Intel®. Навчання для майбутнього”.
2. Intel®. Навчання для майбутнього. – К.: Видавнича група ВНУ, 2004. – 416 с.
3. Дементієвська Н.П. Інтернет і телекомунікаційні проекти мережі шкіл України // Комп'ютер у школі та сім'ї. – 2003. – №6. – С. 31-35.
4. Морзе Н.В. Методика навчання інформатики: Навч. посіб.: Ч. I. / За ред. акад. М.І. Жалдака. – К.: Навчальна книга, 2003. – 254 с.
5. Морзе Н.В., Дементієвська Н.П. Телекомунікаційні проекти: стан та перспективи // Комп'ютер у школі та сім'ї. – 1999. – №4.
6. Новые педагогические и информационные технологии в системе образования / Под ред. Е.С. Полат. – М.: Издательский центр “Академия”, 2003. – 272 с.

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ЯК СКЛADOVA ФОРМУВАННЯ ІНФОРМАЦІЙНОЇ КУЛЬТУРИ

Т.Л. Мазурок, О.А. Логіна
м. Одеса, Південноукраїнський педагогічний університет
імені К.Д. Ушинського
log_lena@ukr.net

Перехід до профільного навчання у старшій школі, визначений Концепцією загальної середньої освіти (12-річна школа), Державним стандартом базової і повної середньої освіти, передбачає докорінну перебудову змісту освіти, зокрема змісту навчання інформатики, яка як окрема навчальна дисципліна входить до галузі “Технології”.

Інформатизація освіти, заснована на використанні засобів нових інформаційних технологій, направлена на інтенсифікацію процесу навчання, реалізацію ідей розвиваючого навчання, вдосконалення форм і методів організації учбового процесу.

Однією з суттєвих ознак готовності до повноцінного життя в інформатизованому суспільстві є здатність особи оперативно обробляти інформацію та приймати на її основі обґрунтовані рішення. Ця здатність може бути сформованою за умов належного рівня розвитку мислення людини. Розвиток мислення є об’єктом формування багатьох шкільних предметів, але розвиток системного мислення може формуватися в сучасному курсі інформатики за рахунок організації системи міжпредметних зв’язків.

Одним з ефективних методів пізнання навколишньої дійсності є метод моделювання. Можливості моделювання різноманітні як по використовуваним формальним моделям, так і по засобам реалізації методів моделювання. Істотне значення має і технологічний розвиток систем моделювання, які на сьогоднішній день є могутнім аналітичним засобом, що увібрав в себе весь арсенал новітніх інформаційних технологій.

Сучасні тенденції розвитку інформаційних технологій висувають вивчення питань, пов’язаних з моделюванням і формалізацією, в число основних задач курсу інформатики. Універсальність технології інформаційного моделювання дозволяє розглядати її як основний компонент формування уявлень про наукові основи інформаційних технологій в шкільному курсі інформатики.

Із вищесказаного можна зробити висновок, що саме моделювання є одним із сучасних інструментів системного аналізу. За останній час у шкільному курсі інформатики відбулися значні зміни, суть яких полягає в зміні пріоритетів від “користувацького” ухилу до інформаційного підходу пізнання навколишнього світу. Про це свідчать численні методичні розробки з навчання учнів моделюванню та формалізації з використанням сучасних

інформаційно-комунікативних технологій, включення до змісту освіти основної школи в розділі 5 “Людина та інформаційна діяльність” теми “Моделювання та інформаційні моделі”.

Згідно з тенденціями подальшого розвитку шкільного курсу інформатики, є включення у зміст навчання, як логічне продовження понять системи, елементів системного аналізу, методів прийняття рішень. У відповідності до дидактичного принципу науковості, інформаційна технологія прийняття рішень має знайти своє відображення в шкільному курсі інформатики. Це відповідає основним змістовним лініям навчання інформатики у старшій школі, реалізує принцип дидактичної спіралі.

Методична база формування вмінь по прийняттю рішень потребує теоретичного обґрунтування, дослідження, проведення педагогічних експериментів. Аналіз методичних основ формування вмінь використання інформаційної технології прийняття рішень вказує на протиріччя між соціальним замовленням на сучасну підготовку учнів до повноцінного життя і діяльності в інформатизованому суспільстві в умовах широкого використання засобів інформаційно-комунікаційних технологій, зокрема, технологій підтримки прийняття оптимальних рішень, і методичною базою здійснення цієї підготовки в практиці навчання інформатики.

Також, мислення людини формується і розвивається в процесі рішення розумових задач як основного виду розумової діяльності. У практиці шкільного навчання рішення учнями різноманітних задач виступає однією з головних умов і засобів оволодіння знаннями і уміннями, розвитку розумових здібностей і особових якостей.

Одним із засобів реалізації загальноосвітнього потенціалу інформатики є метод конструювання навчання з використанням засобів інформатизації освіти, що передбачає систематичне використання при розв’язуванні задач нових комп’ютерних технологій. При цьому власне інформаційний компонент технології включає визначення доцільної форми пред’явлення, переробки і подання інформації тим або іншим технічним засобом.

Застосування засобів нових інформаційних технологій дозволяє підсилити мотивацію навчання завдяки не тільки новизні роботи з комп’ютером, яка сама по собі нерідко сприяє підвищенню інтересу до навчання, але і можливості регулювати пред’явлення задач по трудності, заохочуючи правильні рішення, не вдаючись при цьому до моралей і осуду. Працюючи на комп’ютері, учню надається можливість довести рішення будь-якої учбової задачі до кінця, оскільки йому виявляється необхідна допомога, а якщо використовуються найефективніші навчальні системи, то йому пояснюється рішення, він може обговорити його оптимальність і тупикові ходи. Комп’ютер може впливати на мотивацію учнів, розкриваючи практичну значущість матеріалу, що вивчається, надаючи їм можливість проявити оригінальність, поставивши цікаву задачу, поставити будь-які питання і запропонувати знайти можливі рішення [2].

Як найважливіші цілі навчання інформатиці, на сьогоднішній день, наживаються орієнтація на оволодіння методологічними знаннями і засобами продуктивної діяльності, формування уміння вирішувати змістовні задачі, прививання готовності до ухвалення рішень на основі адекватної оцінки одержаної інформації, знайомство з принципами формалізації і моделювання.

Орієнтація на особу учня, на активне освоєння суб'єктом способів пізнавальної діяльності визначає необхідність аналізу процесу рішення задач з використанням комп'ютерів, дослідження узагальненої стратегії рішення деякого класу задач.

Процес комп'ютеризації навчання, пізнавальна діяльність в системі "людина-комп'ютер" вимагають обґрунтованих рекомендацій для ефективної організації інтелектуальних рішень взаємодіючої з комп'ютером людини. Комп'ютер як засіб інтелектуальної діяльності пред'являє особливі вимоги до організації мислення [3].

Широке упровадження комп'ютерів в практику освіти не тільки висуває вищі вимоги до організації розумової діяльності, але і створює якісно нові умови для розвитку мислення учнів.

Використання людиною комп'ютерів для підвищення ефективності своєї професійної діяльності за рахунок вирішення ним задач на комп'ютері створює нові умови для культивування у себе різних типів рефлексії. Це пов'язано з тим, що залучення комп'ютера для вирішення задач людиною – користувачем ЕОМ, дозволяє фіксувати різні етапи і проміжні результати пошуку рішення задач, забезпечує як би експансію на дисплеї особливостей протікання розумового процесу, що відображується в символічно-комп'ютерній формі.

Людина в своїй діяльності стикається з ситуаціями, в яких їй доводиться здійснювати вибір.

Власне вибір (або прийняття) рішень полягає у вказівці серед всіх можливих такого рішення, яке буде якнайкращим.

Процес прийняття рішення здійснюється за допомогою людини – це особа, що приймає рішення (ЛПР). При цьому вона одержує і вибирає найбільш оптимальну альтернативу з урахуванням прорахунку всіх наслідків.

При виборі альтернатив треба вибирати ту, яка якнайповніше відповідає поставленій меті, але при цьому доводиться враховувати велику кількість суперечливих вимог і, отже, оцінювати вибраний варіант рішення по багатьох критеріях.

З давніх часів людство, використовуючи метод проб і помилок, інтуїцію і досвід, виробляло підходи до ухвалення найкращих рішень в самих різних областях своєї діяльності.

Прийняття рішень в реальних задачах – проблема досить складна, перш за все, через різноманітні альтернативні рішення і обмежені можливості тих, хто узявся за їх пошук.

Випробуваний метод проб і помилок в наші дні часто втрачає свою універсальність, оскільки помилки можуть виявитися катастрофічними і для проб відпущено дуже мало часу.

Застосування математичного апарату і комп'ютерного моделювання дозволяє підготувати учнів ще в період навчання до розуміння важливості і необхідності пошуку рішень одним з можливих методів.

Саме той метод, який учні засвоять на уроці, обов'язково буде використаний пізніше при вирішенні самих різноманітних “дорослих” життєвих та професійних задач. Можливо, це проявиться неусвідомлено і через багато років після закінчення шкільного курсу інформатики.

Системна діяльність має на увазі використання сукупності взаємопов'язаних логічних і аналітичних процедур. Вони полегшують людині аналіз реальних життєвих ситуацій, що передують розумному вибору мети своєї діяльності, способів її досягнення і організаційних форм реалізації, дозволяють об'єктивно протиставити бажане з фактично досягнутим. На будь-якому етапі системна діяльність, хочемо ми того або не хочемо, нав'язує нам процедуру вибору того або іншого рішення. Найуразливіша ланка дій – вибір мети, оскільки помилка у виборі мети зводить нанівець всю решту роботи.

Щоб вирішувати – треба вибирати; щоб вибирати – треба перебирати варіанти; щоб перебирати варіанти – треба моделювати. Вибір – один з основних актів розумово-логічної діяльності людини. У цей момент учень намагається співвідносити свої бажання і свої ресурси. У більшості випадків особа, що ухвалює рішення, йде на компроміс і крім власних інтересів враховує ще і інтереси тих, на кому відобразиться майбутнє рішення. Проблема вибору особливо гостра в нестандартних ситуаціях. Системна діяльність по самому своєму визначенню є ефективний і надійний метод організації знань про реальні об'єкти і їх поведінку, своєрідний спосіб для обігу знань в уміння жити. Смак до системної діяльності в нашому складному реальному світі слід було б привити з малих років. Поняття про складні і прості системи існує стосовно до моделей, що пояснюють і прогнозують їх поведінку. Модель, яка, дозволяє з достатньою точністю передбачити майбутнє системи, вже корисна, а система проста, хороша і зрозуміла [4].

Значний внесок у розвиток творчої компоненти учня, як відомо, вносить правильно організований процес побудови моделі. Комп'ютер може моделювати будь-який процес або послідовність подій. Це дозволяє учню робити самостійні висновки з приводу факторів, що впливають на перебіг процесу або події. При здійсненні цієї роботи учень може приймати рішення і перевіряти їх наслідки. Велика перевага комп'ютерного моделювання полягає також в можливості неодноразового повторення імітації, часто через короткі часові інтервали, до тих пір, поки не буде досягнутий суб'єктивно бажаний результат. Це дає можливість учням глибше засвоїти механізм дії і взаємодії різних факторів, а також придбати і осмислити уявлення про дося-

гнуते, що теж важливе.

Становлення шкільного курсу інформатики зв'язується не з модою на комп'ютери, не з престижністю комп'ютеризованого учбового закладу, не з широким поширенням обчислювальної техніки, а з концептуальним запасом інформатики як природничонаукової дисципліни, здатної сформувати уміння і навички, здатність мислити. Задача формування мислення у покоління молодих людей трактується як найважливіша мета комп'ютеризації освіти.

Отже, становлення курсу інформатики в школі треба розглядати як позитивну і конструктивну реакцію системи освіти на соціальне замовлення сучасного інформаційного суспільства.

Рівень розвитку методів і засобів сучасних інформаційних технологій створює реальні можливості для їх використання в системі освіти з метою вдосконалення творчих здібностей людини в процесі його навчання [1].

У зв'язку з цим дослідження доцільності включення до змісту профільного навчання методів прийняття рішень, розробка методичних основ формування вмінь використання технологій прийняття рішень, підвищення практичної значущості навчання в умовах становлення інформаційного суспільства є актуальною проблемою.

Література:

1. Раскина И.И. Изучение моделирования и формализации в школьном курсе информатике как компонентов научных основ информационных технологий // Информатика и образование. – 2004. – №7.
2. Босова Л.Л. Компьютерные уроки в начальной школе // Информатика и образование. – 2000. – №1.
3. Скибицкий Э.Г., Шкабура О.В. Стиль мышления как стратегия решения задач с использованием компьютера // Информатика и образование. – 2000. – №10.
4. Перегудов Ф. Системная деятельность и образование // Информатика и образование. – 1990. – №6.

ИСПОЛЬЗОВАНИЕ МОБИЛЬНЫХ КОММУНИКАЦИОННЫХ УСТРОЙСТВ В ОБРАЗОВАТЕЛЬНЫХ ЦЕЛЯХ

И.Е. Мазурок¹, Т.Л. Мазурок²

¹ г. Одесса, Одесский национальный университет им. И.И. Мечникова

² г. Одесса, Южно-украинский педагогический университет

им. К.Д. Ушинского

im@im.tenet.odessa.ua

В настоящее время мы наблюдаем бурный рост возможностей технических средств мобильной коммуникации. Мобильные телефоны оснащаются все более мощными процессорами, все более качественными экранами и превращаются в так называемые SmartPhone. Портативные компьютеры дополняются радиочастотным блоком и превращаются в персональные коммуникаторы. Стандарт GPRS позволил этим небольшим компьютерным устройствам иметь постоянный канал передачи цифровых данных, т.е. находиться в сети Интернет, обмениваться мультимедийными сообщениями (MMS), содержащими звук, текст и видео.

Распространенность мобильных коммуникационных устройств на сегодняшний день составляет для Украины около 10% населения. Охват территории (т.н. «покрытие») зонами действия мобильных операторов также весьма значителен и по некоторым оценкам составляет половину площади страны.

Таким образом, по техническим возможностям и по распространенности среди населения мобильные персональные телекоммуникационные (устройства) терминалы значительно превышают аналогичные характеристики персональных компьютеров на момент начала их использования в педагогических целях. Естественно, что это новое и мощное техническое средство, прочно вошедшее в арсенал современных компьютерных информационных технологий, не может оставаться за пределами внимания педагогов, разработчиков обучающих и развивающих компьютерных программ, инновационных образовательных технологий.

Применения этих устройств в сфере образования расширяет возможности дистанционного образования и область применимости дистанционных технологий. Использование коммуникационных возможностей мобильных устройств позволяет по-новому использовать технику совместных Интернет-проектов [1, 2].

Выделим и детально рассмотрим категории персональных мобильных телекоммуникационных терминалов в зависимости от их характеристик применительно к использованию в современных образовательных технологиях. При разбиении на категории будем учитывать и возможности разработки педагогического (учебного) программного обеспечения.

Категория А. К этой категории отнесем простейшие мобильные тер-

миналы бытового назначения. Возможности программирования этих устройств весьма ограничены. Возможности учебного использования таких устройств не слишком велики.

Во-первых, это прием и отправка коротких (несколько строк) текстовых сообщений (SMS). Т.е. пользователь мобильного терминала может принимать короткие текстовые сообщения в любой момент времени и в любом месте. Сообщения могут рассылаться с другого терминала, либо средствами электронной почты (через e-mail). При помощи этой техники можно организовать следующие педагогические проекты.

- Совместные образовательные проекты учеников из разных городов в области культуры, краеведения, наблюдения за погодными явлениями (по курсу природоведения), реализуемые обменом моментальными сообщениями. В проектах участвуют пары учеников из разных городов, совместно выполняющие общее задание.
- Периодическая рассылка сообщений по подписке (через электронную почту или средствами широкоэмитательных сообщений). Таким образом, группе учеников могут рассылаться сообщения, являющиеся самостоятельными фактами или правилами-напоминаниями из какого-либо раздела учебной дисциплины.
- Рассылка сообщений, которые содержат задания, требующие SMS-ответа (пропущенное слово, результаты вычислений и т.п.).
- SMS-игры. Рассылка сообщений, которые требуют реальных действий получателя. При помощи этого приема можно проводить, например, игры-соревнования по краеведению. В качестве задания указывается (в завуалированной форме) необходимость собраться у определенной достопримечательности города.
- SMS-пейджинг. Система мобильного оповещения о протокольных событиях в рамках структуры дистанционного образования. Этот вид сервиса предоставляется в сочетании с другими, информационно емкими средствами коммуникаций. Таким образом, например, сообщается о готовности нового дистанционного курса, или поступлении e-mail ответа преподавателя на запрос обучаемого.

Второй важной возможностью таких устройств является программируемость реакции абонента на нажатие клавиш абонентом. Это позволяет создавать интерактивные голосовые справочники, управляемые действиями обучаемого. Практически это выглядит, как некоторое речевое меню, разделы которого пользователь выбирает нажатием соответствующей клавиши на своем терминале. Именно таким образом операторы услуг мобильной связи организуют внутреннюю справочную службу для своих клиентов. Подобным образом можно организовать электронный справочник материалов по тем учебным дисциплинам, для которых речевая форма предоставления информации является приемлемой. Особенности клавиатуры терминала и речевая форма вывода результатов запроса позволяют использовать этот

сервис в проектах для людей с дефектами зрения.

Категория В. К этой категории отнесем более сложные мобильные терминалы. Для этих устройств характерна возможность подключения к персональному компьютеру через инфракрасный порт или непосредственно при помощи специального кабеля к последовательному или USB-порту компьютера. Как правило, эти устройства позволяют загружать и выполнять простые Java-приложения (MIDlets – мидлеты). Такие терминалы (мобильные телефоны) на сегодняшний день являются самым распространенным классом мобильных устройств из-за своей низкой цены.

Мобильные терминалы этой группы оснащены черно-белыми или цветными мониторами (4096 цветов) с небольшой разрешающей способностью не более 120x120. Обычный объем оперативной памяти менее 512 килобайт и 16 или 32-разрядный процессор. Для программирования устройств этого класса (CLDC), фирмой SUN Microsystem разработана облегченная версия виртуальной Java-машины – KVM. Используя Java 2 Micro Edition, можно вести кроссплатформенную разработку педагогического программного обеспечения. Т.е. на персональном компьютере устанавливается специальное программное обеспечение, моделирующее мобильные устройства данной платформы. После написания и отладки программ на персональном компьютере они загружаются в память телефона.

Рассмотрим дополнительные возможности устройств этого класса.

Во-первых, подключение к персональному компьютеру позволяет организовать образовательный вариант мобильного офиса. Сочетание портативного компьютера (ноутбук) и мобильного телефона предоставляет все возможности Интернет-связи в «полевых» условиях. Наиболее целесообразно использование такого тандема в распределенных учебных проектах экологического мониторинга, оперативных социологических исследованиях.

Во-вторых, возможности загрузки программного обеспечения позволяют разрабатывать и загружать любое учебное программное обеспечение. При этом следует учитывать ограниченность памяти и невысокие графические возможности терминалов. Т.е. возможны практически любые обучающие программы, не требующие большого количества графики фотографического качества. В этот разряд попадают программы моделирования физического эксперимента, логические тренажеры, программы проекта «Кумир» [3] и многие другие, предусмотренные отечественной образовательной программой [4]. Однако, при разработке учебного программного обеспечения для этого класса устройств следует учитывать их основную специфику – возможность предоставления цифровых каналов связи в «полевых» условиях.

Третья важная возможность – предоставление доступа к WAP-сайтам в сети Интернет. Это открывает web-доступ для обладателя такого устройства. При помощи специально разработанных WAP-сайтов можно построить

распределенные мобильные справочники по учебным дисциплинам. On-line тестирование также легко реализуется уже для этой категории устройств.

Полезным в учебном плане является и развитие возможности коротких текстовых сообщений – MMS-сообщения, сопровождаемые звуком, музыкой и графическими изображениями или короткими фрагментами видео. Эта техника используется в тех же учебных проектах, что и SMS-сообщения.

Категория С. К этой категории относятся CDC-мобильные телефоны, оснащенные мощными процессорами (до 64 бит) и имеющие большой объем (от 1 мегабайта и более), иногда называемые SmartPhone. Как правило, такие телефоны оснащают цифровой видеокамерой (1 мегапиксел и более) и мониторами с повышенной разрешающей способностью (от 176x208 до 352x416) при 64К цветов. Для этих устройств также характерно наличие устройства высокоскоростного автоматического сопряжения компьютерных устройств (Blue Tooth) и системы быстрого Интернет по стандарту GPRS.

Мощность процессора и большой объем памяти позволяет реализовывать сложные учебные программы. Кроме дальнейшего развития возможностей программирования и сетевых средств (поддержка стандарта MIDP – Mobile Information Device Profile версии 2.0), совершенно новые возможности предоставляет наличие цифровой видеокамеры. Возможность оперативной фото и видеосъемки с передачей данных по сети предоставляет возможность развивать совместные Интернет-проекты на новом качественном уровне.

Примером реализации учебного проекта реализуемого для этой категории устройств могут служить совместно разрабатываемые учениками различных городов мобильные проекты «Восходы Украины». Участники проекта в условленный день фотографируют восход солнца в их местности. Фотографии в сопровождении информации о точном времени и координатах точки наблюдения передается по сети. Аналогичными по форме, но различными по реализации являются астрономические проекты по лунным затмениям, метеорным потокам, проекты наблюдения за погодой и др.

Общей особенностью всех описанных категорий устройств, значительно ограничивающих их образовательные возможности, является неудобство ввода текстовой информации. Клавиатура телефона позволяет легко набирать числа, но плохо приспособлена для ввода текста. Этого недостатка лишены устройства, которые мы отнесли к категории D.

Категория D – персональные коммуникаторы. Кроме собственно коммуникаторов, в эту категорию отнесены также «наладонные» компьютеры и PDA различных модификаций, оснащенные радиочастотным модулем. Вычислительные возможности этих устройств практически аналогичны устройствам категории С, однако они отличаются большими размерами экрана и более развитыми средствами ввода информации. Последняя проблема решается либо за счет полнофункциональной клавиатуры, либо путем исполь-

зования чувствительного (сенсорного) экрана и рукописного ввода.

Мобильные устройства категории D предоставляют практически неограниченные возможности для построения обучающих и развивающих сетевых программ и организации учебных проектов путем взаимодействия с полнофункциональными Интернет-серверами поддержки проекта (игры). Наличие полнофункциональной системы ввода текстов позволяет реализовать учебные языковые проекты или полноценные справочные и тестирующие системы [5]. Возможности устройств этого класса приближаются к т.н. мобильному офису.

Однако, следует отметить, что высокая стоимость устройств этой категории пока еще сильно ограничивает их распространение. В тоже время, вспоминая заявления многочисленных скептиков по поводу педагогического использования персональных компьютеров, сделанные на заре компьютеризации образования, хочется надеяться, что они вновь окажутся неправы. Следует отметить и неуклонное снижение цен на высокотехнологичные устройства категорий C и D. Ценовая политика многих компаний на этом рынке определяется не техническими характеристиками предлагаемых устройств, а вопросами моды и рекламы, что делает возможным приобретение и использование недорогих, но мощных и полнофункциональных мобильных устройств.

Описанные категории устройств и их возможности в области коммуникаций, простота разработки кроссплатформенного программного обеспечения на языке Java при помощи многочисленных инструментальных средств (Nokia Developer's Suite, SUN Net Beans Mobility, Java Studio Mobility) дает основание надеяться на широкое распространение их в сфере информатизации и компьютеризации образования.

Литература:

1. Барыкова Н.А. Метод проектов в преподавании информатики в системе общего среднего образования. // Сб. трудов "X международная конференция "Информационные технологии в образовании, часть II". – М., МИФИ, 2000. – С. 60–61.

2. Пак Н.И. О нелинейных технологиях обучения // Информатика и образование. – 1997. – №5.

3. Кушниренко А.Г., Лебедев Г.Е. 12 лекций о том, для чего нужен школьный курс информатики и как его преподавать // Информатика. – 1999. – №1, 3, 5, 6, 8, 10, 11, 14.

4. Программа для общеобразовательных учебных заведений «Основы информатики и вычислительной техники» // Жалдак М.И., Морзе Н.В., Науменко Г.Г. – К.: Шкільний світ, 2001.

5. Атаманчук П.С. Призначення та особливості оперативного контролю готовності студентів до засвоєння навчального матеріалу // Проблеми освіти: Наук.-метод. зб. – К., 2001. – Вип. 24. – С. 159-165.

МУЛЬТИМЕДИЙНЫЙ АППАРАТНО-ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ ОТОБРАЖЕНИЯ УЧЕБНОЙ ИНФОРМАЦИИ НА ОСНОВЕ БЫТОВЫХ ТЕЛЕВИЗИОННЫХ ПРИЕМНИКОВ

Г.Г. Маклакова, Г.Ю. Маклаков, Е.А. Кожаев

г. Севастополь, Севастопольский национальный технический университет

Одним из путей повышения эффективности учебного процесса является использование мультимедийных технологий. Внедрение таких прогрессивных технологий обучения во многом сдерживается из-за необходимости использовать специализированные устройства для их реализации. Тривиальным решением является применение стационарных компьютеров, однако из-за их дороговизны использовать их исключительно для мультимедийного отображения информации является непозволительной роскошью, как в средней, так и высшей школе. Портативные компьютеры класса ноутбук (NoteBook) также достаточно дороги, и поэтому применяются в учебном процессе ограниченно не только в нашей стране, но и за рубежом. Следует заметить, что дороговизна современных компьютеров определяется в основном их универсальностью – они изначально ориентировались на решение самого широкого класса задач. В связи с этим представляется целесообразным разработку специализированных устройств, ориентированных только на отображение мультимедийной учебной информации. Для эффективного использования в учебном процессе современных устройств отображения информации они должны иметь высокие эксплуатационные характеристики при минимальной стоимости и высокой надежности.

Одним из подходов к разработке специализированных устройств мультимедийного отображения информации является создание портативных устройств на основе микроконтроллеров с выводом информации на жидкокристаллический индикатор (ЖКИ) [1–5].

Не отрицая преимуществ применения ЖКИ, следует все же отметить, что они достаточно дороги и их стоимость по сути дела определяет стоимость всего устройства (порядка 400–450 грн. для устройства с монохромным дисплеем с размером экрана 148x75 мм). При использовании цветных ЖКИ стоимость всего устройства возрастает в 7–9 раз, что ставит под сомнение широкое внедрение таких систем.

В данной работе рассматривается класс альтернативных устройств отображения мультимедийной информации. В качестве дисплея в них используется бытовой телевизор. Представляется целесообразным практически реализовать данное направление в разработке устройств со следующими параметрами:

- 1 – вывод информации только в виде текста и графики (минимальная стоимость);
- 2 – вывод информации в виде текста и графики с использованием про-

стейшей анимации (на основе Flash-технологий);

3 – вывод информации в виде звука, текста, графики с использованием простейшей анимации (на основе Flash-технологий);

4 – вывод информации в виде звука, текста, графики с использованием полной анимации;

5 – вывод информации с максимальным использованием возможностей мультимедийных технологий (включая просмотр учебных фильмов в цифровом формате).

Устройства, начиная со 2 подкласса, имеют развитый интерфейс регулировки формы представления информации.

Техническую реализацию разработки устройств такого класса целесообразно проводить в рамках принятой нами ранее концепции создания аппаратно-программных комплексов для мультимедийного отображения учебной информации [1–3]. Согласно этой концепции устройство должно удовлетворять следующим принципам:

- для хранения информации должна использоваться энергонезависимая память;

- для управления устройством необходимо использовать специализированный микропроцессор;

- устройство должно обеспечивать простоту обращения с ним (рассчитано на неквалифицированного пользователя).

Проведенный анализ существующих подходов к построению аналогичных устройств позволил выбрать соответствующую элементную базу. Наиболее рациональным решением для аппаратно-программных комплексов выбранного направления является Flash-память. Обзор существующих микросхем Flash-памяти показал, что для устройства данного класса предпочтительно использовать CMOS SEEPROM семейства AT24 разработки фирмы Atmel. В качестве протокола обмена выбрали шину с последовательным способом передачи данных I2C («Inter-Integrated Circuit»), которая обеспечивает обмен данными на скорости до 100 Кбит/сек. В качестве специализированного процессора целесообразно использовать PIC-микроконтроллер. Эти микроконтроллеры объединили в себе все передовые технологии, применяемые в производстве микроконтроллеров: развитую RISC-архитектуру, малое энергопотребление, высокое быстродействие, ППЗУ, программируемое пользователем [4, 5]. Они имеют хорошие эксплуатационные характеристики при минимальной стоимости, могут гармонично сочетаться с универсальными вычислительными комплексами на базе процесса класса Pentium, дополняя их в тех случаях, когда использование дорогостоящих устройств экономически не выгодно.

Исходя из принятого подхода, была разработана типовая структурная схема устройств мультимедийного отображения информации с выводом информации на телевизионный приемник (см. рис.1).

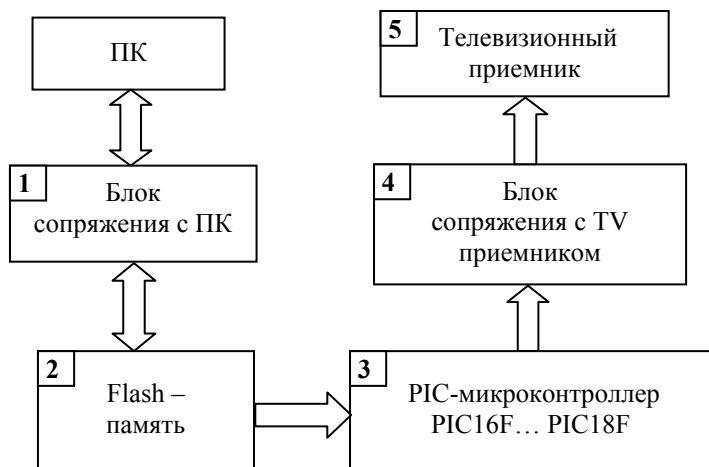


Рис. 1. Типовая структурная схема устройств мультимедийного отображения информации с выводом информации на телевизионный приемник

Блок 1 – блок сопряжения аппаратно-программного комплекса с параллельным портом персонального компьютера. Блок предназначен для подготовки и перезаписи мультимедийной информации из базового компьютера во внешнюю память. В качестве базовой ЭВМ используется персональный компьютер стандартной комплектации, функционирующий в операционной системе Windows 9*/XP.

Блок 2 представляет собой энергонезависимую Flash-память с последовательным интерфейсом I2C. Блок служит для хранения мультимедийной информации. Анализ задачи позволяет предположить, что для проектируемого комплекса 1-го подкласса можно ограничиться объемом памяти в 512 К. Исходя из вышесказанного, выбираем в качестве Flash-памяти микросхему AT24C512.

Блок 3 – управляющий процессор на базе PIC-микрoкoнтрoллера. Основное назначение блока – организовать работу всего комплекса и обеспечить удобный интерфейс работы системы с пользователем. Для комплекса 1-го подкласса достаточен микрoкoнтрoллер PIC16F628A. Для старших моделей необходимо использовать микрoкoнтрoллер не ниже PIC18F452.

Блок 4 – блок сопряжения. Обеспечивает вывод информации на телевизионный индикатор. Для комплексов 1-го и 2-го подклассов возможна его реализация на микрoкoнтрoллере SED1336. Для старших моделей необходимо специальное устройство генерации полного видеосигнала.

Блок 5 – устройство отображения. В качестве устройства отображения, как уже указывалось, используется телевизионный приемник.

В соответствии с постановкой задачи аппаратно-программный комплекс должен состоять из двух отдельных устройств: стационарного и переносного, и блока внешней памяти (EEPROM). Переносное устройство реализует функции мультимедийного отображения учебной информации, считываемой из EEPROM памяти. Стационарное устройство представляет собой аппаратно-программный комплекс, служащий для записи учебной информации (текстовой, графической и звуковой) во внешнюю память. Аппаратный блок представляет собой программатор, заносающий информацию, поступающую из COM-порта, в EEPROM память. Программная часть кодирует информацию из текстового, графического и звукового файлов в 16-ричный формат, необходимый для последующей записи, и управляет работой программатора. Управляющая программа стационарного устройства функционирует в ПК стандартной конфигурации (ОС – Windows 9*, XP; процессор Pentium III, память ОЗУ 256 Мб, НЖМД 40 Гб).

Файл сценария может быть создан в стационарном устройстве с помощью соответствующего редактора. Также предусмотрена возможность импорта ранее созданного файла, который при необходимости может быть передан в редактор сценария, если в него требуется внести изменения. Текст и графика (анимация), а также меню, необходимое для управления устройством, отображаются на экране телевизионного приемника, который выполняет функцию монитора.

Рассмотренный класс альтернативных устройств отображения мультимедийной информации позволяет резко сокращается стоимость всего устройства (примерно на порядок) при удовлетворительных эксплуатационных характеристиках. Учитывая тот факт, что современные телевизионные приемники нашли широкое распространение в учебных заведениях и в быту выбранный нами подход может способствовать внедрению прогрессивных информационных технологий в учебный процесс.

Литература:

1. Маклаков Г.Ю., Кожаев Е.А., Маклакова Г.Г. Перспективы использования современных микроконтроллеров для создания портативных устройств мультимедийного отображения учебной информации. В кн.: Теорія та методика навчання математики, фізики, інформатики: Збірник наукових праць. Випуск 4: В 3-х томах. – Кривий Ріг: Видавничий відділ НМетАУ, 2004. – Т.3: Теорія та методика навчання інформатики. – С. 176-179.
2. Кожаев Е.А., Маклакова Г.Г. Специализированный аппаратно-программный комплекс для мультимедийного представления учебной информации. В кн.: Информационные технологии и информационная безопасность в науке, технике и образовании «ИНФОТЕХ-2004». Материалы международной научно-практической конференции, 20–25 сентября 2004, Севастополь. – Киев-Севастополь: НТО РЭС Украины,

2004. – С. 178-182.

3. Маклакова Г.Г., Маклаков Г.Ю. Особенности архитектуры аппаратно-программного комплекса для мультимедийного представления учебной информации. В кн.: Системний аналіз та інформаційні технології: Тези доповідей учасників VI Між нар. наук. практичн. конф. студ., асп. та молодих вчених (1-3 липня 2004 р., м. Київ). – К.: НТУУ «КПІ», 2004. – С. 167-168.
4. Тавернье К. ПС-микроконтроллеры. Практика применения. – М.: ДМК Пресс, 2003. – 272 с.
5. Яценко В.С. Микроконтроллеры Microchip. Практическое руководство. – М.: Горячая линия-Телеком, 2002. – 296 с.

ОБРАЗНАЯ СИСТЕМА АЛГОРИТМИЧЕСКОГО ЯЗЫКА

В.Л. Малорян

г. Одесса, Южно-Украинский государственный педагогический университет
им. К.Д. Ушинского
mvadim@mail.od.ua

В контексте задач формирования алгоритмической культуры важнейшим представляется вопрос о месте этого весьма специального аспекта культуры в общем ее потоке. Первоначальное значение понятия «алгоритм» для культуры в ее историческом контексте приносит жесткую внутреннюю связь в само понятие «алгоритмическая культура» [2]. Однако традиционное противопоставление понятий «алгоритм» и «творчество» в их концептуальном плане до сих пор сдерживало возможности раскрытия гуманитарного потенциала алгоритмики.

Преодоление этого противопоставления представляется возможным на пути выделения базисных образов-архетипов современного алгоритмического языка с целью их соотнесения с живыми образами традиционного искусства.

Установим это соотнесение, взяв в качестве рабочего материала конкретизации операторы языка программирования Оберон, разработанного Н. Виртом как естественное и гармоничное развитие Паскаля. Рассматриваемые операторы относятся к классам операторов цикла, выбора, присваивания, вызова, детализации, выхода.

Для каждого из операторов предложены графический и поэтический образы, цвет, звук и музыкальный аккорд. Типичными примерами предлагаемых визуальных образов являются: кольцо, спираль, змея, часы – для операторов цикла; река, дорога – для операторов выбора; лифт, тор – для вызова и рекурсии; пирамида – для детализации; лестница – для следования; гвоздь, буква «зэт» – для выхода; арка, яйцо – для начала модуля. Цвета и звучание операторов выбирались в соответствии с принятыми в искусстве и эзотерике символическими смысловыми значениями цветов и звуков, взятыми нами из [1]. Например, красный цвет соответствует ноте «до». Основные образы алгоритмического языка императивного типа (Паскаль, Оберон) сведены в таблицу. Визуальная подсистема системы образов может быть также оформлена в виде набора карт.

Таблица образов операторов алгоритмического языка

<i>Оператор</i>	<i>Цвет</i>	<i>Визуальный образ</i>	<i>Звук</i>	<i>Аккорд</i>
REPEAT	Красный	Змея с рубиновым хвостом	До	C6,4
FOR	Оранжевый	Будильник	Ре	D7
LOOP	Желтый	Кольцо	Ми	E6
WHILE	Зеленый	Змея с изумрудным глазком	Фа	F

<i>Оператор</i>	<i>Цвет</i>	<i>Визуальный образ</i>	<i>Звук</i>	<i>Аккорд</i>
IF	Голубой	Река, дорога	Соль	Gm
CASE	Синий	Развилка	Ля	Am
CALL	Фиолетовый	Лифт	Си	Hm6

Совместное использование образов способно резко повысить степень восприятия и понимания соответствующей алгоритмической конструкции при ее изучении. Кроме того, разработанная система образов создает основу для построения системы отображения (трансляции) текста программы в мультимедийные образы. Использование подобной системы позволит решать не только задачу обучения основам алгоритмической культуры, но и может существенно облегчить процесс понимания программы за счет задействования образотворческих правополушарных механизмов мозга обучаемого, а также облегчить процесс нахождения ошибок в программе.

Разработанная система образов практически без изменений переносится на конструкции других языков императивной парадигмы – Паскаль, Си и т.п. В будущем, после получения подтверждений её практической полезности, планируется распространение системы и на языки других парадигм.

Рассмотрим возможные области применения описанной системы образов в современном педвузе.

Процесс овладения основами алгоритмической культуры в педагогическом вузе должен учитывать фундаментальный и стабильно действующий фактор преимущественно гуманитарной ориентации большинства студентов, даже на специальностях физико-математического и информатического направлений. Это порождает целый букет проблем, подходы к решению которых простираются от принятия специальных мер для сглаживания различий в начальном уровне подготовки студентов до организации процесса непрерывной адаптации учебных курсов по информатике к возможностям и уровню конкретных групп и потоков студентов.

Представляется весьма перспективным предлагаемый подход к адаптации существующих учебных курсов, включающих алгоритмические компоненты, путем использования в них мультимедийных образов как средства активизации и переключения внимания обучаемых на другие уровни восприятия алгоритмических текстов. В качестве основы для реализации взята предложенная система образов алгоритмического языка.

Для реализации этого подхода в настоящее время под руководством автора на кафедре прикладной математики и информатики ЮУГПУ им. К.Д. Ушинского разрабатывается мультимедийная студия – среда системы программирования на языке Оберон, включающая палитры образов, словари языка и программы, автоматический генератор текста по последовательности образов, транслятор текста в последовательность образов, а также, подсистему проигрывания программы [3]. По замыслу, студия станет надстройкой над системой программирования Блэк Бокс, образовательная вер-

сия которой получена нами от разработчика – фирмы Оберон Майкросистемс (Швейцария).

С одной стороны, использование разрабатываемой системы может облегчить процесс разработки программы на Обероне, удовлетворяя потребность программиста в визуальном макросредстве. С другой стороны, она сможет облегчить и «одухотворить» процесс поиска ошибок в программе, сводя его к мгновенному отслеживанию диссонансов в звучании программы. Для этого обучаемый, конечно, должен предварительно приобрести некоторый опыт прослушивания правильных программ.

Далее, система способна стать для преподавателя удобным средством отслеживания индивидуальных особенностей стиля программирования студентов, а затем и средством обнаружения плагиата в программах.

Наконец, создавая новый слой – уровень виртуальной программной реальности – система становится инструментом своеобразного «оживления» мертвого текста программ.

Получаемые и просматриваемые в режиме реального времени мультипликация обладают своеобразной странной красотой, и поэтому могут стать частью практикума к спецкурсу «Красота в программировании».

Однако нельзя не отметить наличие серьезных препятствий и проблем на описанном пути. Важнейшей из них является проблема возможного неконтролируемого воздействия потока образов на подсознание обучаемых. Поэтому практическому внедрению будущей системы должны предшествовать тщательные психологические исследования.

Литература:

1. Энциклопедия символов, знаков, эмблем. – М.: МИФ, 2002. – 556 с.
2. Малорян В.Л. Компоненты алгоритмической культуры и задача ее формирования ВУЗом // Нові інформаційні технології навчання в учбових закладах України, ч.1. Зб. статей з проблем десятої Української наук.-метод. конф. 10-12 вересня 2002 р. / Під редакцією І.І. Мархеля. – Одеса, 2003. – С. 39-46.
3. Малорян В.Л., Романенко А.Б. Трансляция в мультимедийные образы как средство гуманитаризации процесса овладения основами алгоритмической культуры. // В сб. «Информационные технологии в учебном процессе». – Одесса: ЮУГПУ, 2003. – С. 155-157.

ПЕРЕВАГИ XML ПРИ РОЗРОБЦІ ОСВІТНЬОГО ПОРТАЛУ

Ю.С. Матвієнко

м. Полтава, Полтавський державний педагогічний університет

імені В.Г. Короленка

uuri@rambler.ru

В Інтернеті термін “портал” спочатку використовувався для назви сайтів, таких як Excite, Yahoo, MSN, Netscape, Netcenter, Rambler, Яндекс, що забезпечують користувачам “централізований вхід” та спеціальні засоби для зручного переміщення по мережі. Кінцевим етапом еволюції таких сайтів стало створення так званих горизонтальних порталів.

Горизонтальний портал – це портал, зорієнтований, перш за все, на максимально широке коло інтересів своїх користувачів [1].

З ростом аудиторії з’явилися тематичні спільноти, що об’єднують групи користувачів мережі за визначеними інтересами. Точками входження таких користувачів стали вертикальні або профільні портали. *Вертикальні портали*, на відміну від горизонтальних, зорієнтовані на повне охоплення певної аудиторії, тематики або сфери діяльності людини [2].

Загальними рисами порталів усіх типів є можливість інтеграції та агрегації великого об’єму неоднорідних даних, існування розвинених механізмів пошуку та існування засобів персоналізації для окремого користувача.

Інформатизація освіти та розвиток єдиного інформаційно-освітнього простору України передбачає створення системи вертикальних освітніх порталів.

Освітній портал – це віртуальне середовище, яке містить інформаційні дані освітньої галузі та надає користувачам можливість їх повного використання для полегшення процесу навчання.

Проблема створення таких порталів досить актуальна на даному етапі інформатизації освіти.

В наш час розширені мови розмітки (Extensible Markup Language, XML) стали стандартом «де факто» для обміну метаданими та структурованою інформацією в сучасних освітніх порталах. XML є універсальним, гнучким та виразним засобом. Крім того, XML дозволяє відокремити сутність документу, його логічну структуру від графічного представлення документу, його зовнішнього вигляду. XML дозволяє «втягти» на синтаксичному рівні сутнісну структуру документу. Такий підхід має далекі наслідки. При конструюванні освітніх порталів для нас найбільш важливими є:

Незалежний формат даних. XML надає універсальні засоби структуровання документу на логічному рівні, залишаючись при цьому відкритим стандартом. Незалежно від того, в якій системі ця розмітка була проведена, її будуть розуміти й інші системи. Використання XML передбачає відокремлення змістовної складової документу від форми його представлення, що

дуже важливо для електронних навчальних матеріалів. Життєвий цикл навчального курсу, оформленого у вигляді xml-документу, буде залежати тільки від його значення для освітнього процесу. Навіть якщо комп'ютерна система, в якій він був підготовлений, «загине», навчальний матеріал може бути використаний в інших системах. Те ж саме стосується й будь-яких інших видів xml-документів – наукових статей, книг, різноманітних каталогів.

Простота структури XML. Конструкції XML прості та зрозумілі. Вони дозволяють дуже легко настроїти мову на відповідну предметну область. Це значить, що XML може використовуватися людьми, далекими від комп'ютерних технологій. Розробник документа, чи то професор, автор книги, студент, чи вчитель, створюючи текст, працює в термінах близької йому предметної області. Звісно, напряду XML використовувати не дуже зручно. Але можливе створення спеціальних редакторів та інтерфейсів, дуже простих та зрозумілих в поведінці, на яких може працювати практично будь-яка людина, що не боїться підійти до комп'ютера.

XML-документ – загальне джерело для декількох представлень документа. Ще одна суттєва перевага XML – різке підвищення ефективності роботи з документами. Документи готуються швидко, але це не все. Створюючи xml-документ, ми завчасно не обмежуємо варіанти його зовнішніх графічних форм. Тому, маючи в якості джерела xml-документ, ми можемо за допомогою спеціальних програм автоматично конвертувати це джерело в різноманітні графічні представлення, наприклад, в HTML формат, формат редактора Microsoft Word, pdf-формат, автоматично перетворювати книги на сайти та навпаки. Одного разу підготовлений xml-матеріал може стати джерелом цілого спектру представлень документа, що звільняє розробника від рутинного поліграфічного оформлення. Не псуючи документ зайвою поліграфічною інформацією, ми даємо можливість для більш повного, гнучкого та вільного використання документа в різних форматах та програмах.

Доступність даних. Дуже велика кількість проблем при розробці освітніх порталів виникають через використання різноманітних програмних засобів, що працюють з несумісними стандартами. Практика вказує на те, що витрати, які пішли на подолання несумісності стандартів, найчастіше перевищують витрати на основну роботу. XML, що має відкритий стандарт, робить підготовлені в ньому документи загальнодоступними.

Технологічність. XML дозволяє будувати інформаційні системи, в повній мірі використовуючи переваги об'єктно-орієнтованого підходу. Зв'язка XML+Java забезпечує незалежність інформаційних систем від конкретних платформ [3].

«Інтелектуальний» пошук. В xml-форматі змістовна (логічна) структура документа чітко описана. Структурування документів на змістовному рівні дозволяє забезпечити систему пошуку багатьма можливостями. Наприклад, ми можемо зобов'язати таку систему знайти всі документи, в анотації до яких міститься слово «інтеграл», знайти документи конкретного

автора, які містять теорему Лопітала та ін. Ці можливості ефективно доповнюються механізмами динамічної каталогізації – система може представити не тільки посилання на документ, що задовольняє умовам пошуку, але й обробити його у відповідності із запитом користувача, наприклад, зробити «вибірку» з документа. Припустимо, ми хочемо переглянути вміст документів, що містять інформацію про теорему Лопітала. Тоді можна зробити наступний запит: знайти всі документи, що містять поняття «теорема Лопітала», сформувані динамічний каталог цих документів, що міститиме для кожного документа назву, авторів, анотацію та вміст. Ці можливості роблять пошук по xml-документам більш потужним засобом, аніж пошук за ключовими словами, що використовується зараз у добре відомих пошукових системах Інтернету [4].

Коротко опишемо конструкції мов розмітки, що використовуються для структурування даних. Ідея XML полягає у використанні спеціальних «дужок», які розставляються в тексті. Крім того, що «дужка» виділяє деякий сегмент тексту, вона містить внутрішню інформацію, що характеризує виділений текст. Наприклад,

```
<анотація ім'я="an1">  
    В роботі розглядається технологія створення та  
    обробки документів, що заснована на змістовній  
    (логічній) розмітці текстів.  
</анотація>
```

Для кращого розуміння теги спеціально подані в україномовному вигляді.

В цьому прикладі за допомогою засобів XML розмічена анотація. Відкриваючі та закриваючі «дужки» розмітки (<анотація...> та </анотація>) називаються тегами. Ознакою закриваючого тега є слеш «/», що ставиться перед іменем тега.

Конструкція тегів споріднює XML та HTML, який також є мовою розмітки. Але між ними є і дуже суттєва відмінність. По-перше, в HTML використовується фіксований набір імен тегів – div, frame, img, object і так далі. XML – це спектр мов, що настраює предметну область. Під кожен предметну область можна налагодити свій варіант мови, який буде найбільш зручним для опису логічної структури документів з цієї області. По-друге, HTML в основному орієнтований на зовнішнє представлення документа та використовує конструкції для форматування тексту та його графічне представлення. XML орієнтований на змістовну, логічну структуру документа. І врешті-решт HTML та XML мають низку синтаксичних відмінностей. Наприклад, в HTML можна використовувати теги, що відкриваються, без тегів, що закриваються, не використовувати в атрибутах лапки та користуватися низкою інших послаблень, що неможливі в XML.

Отже, сам процес створення XML-документа дуже легкий та вимагає від нас лише базових знань HTML та розуміння тих завдань, які ми бажаємо виконати, використовуючи XML в якості мови розмітки. Таким чином, у

розробників виникає унікальна можливість визначати власні команди, що дозволятимуть їм найбільше ефективно визначати дані, які містяться в документі. Автор документу створює його структуру, будує необхідні зв'язки між елементами, використовуючи ті команди, які задовольняють його вимогам та досягають того типу розмітки, який необхідний для виконання операцій перегляду, пошуку, аналізу документу.

Описані вище переваги та можливості XML дуже важливі в освітній та науковій сферах, де пошук інформації, реферування, каталогізація відіграють величезну роль. Якщо документи, що обробляються, зберігаються в xml-форматі, то левову частку цією роботи можна проводити автоматично.

Достатньо гнучкий для того, щоб використовуватися в різних сферах, в тому числі й в освітній, що має просту структуру, XML залишає прозорим зміст документа, а завдяки розмітці, дозволяє легку обробку та використання документа. Зважаючи на все вище зазначене, XML стає першим кандидатом на базовий стандарт освітнього порталу.

Література:

1. Матвієнко Ю.С., Литвиновський Є.Ю. Освітні портали: визначення, проблеми організації / Зб. наук. праць «Військова освіта». – Вип. №2(14). – К., 2004. – С. 151-159.
2. Бабинский А.З., Букатов А.А., Шапиро В.А., Шаройко О.В. Определение базовых сервисов, разработка методики наполнения и методов реализации образовательных порталов / Сб. научн. ст. “Интернет-порталы: содержание и технологии”. Вып. 1. ГНИИ ИТТ “Информика”. – М.: Просвещение, 2003. – С. 329-364.
3. Дакота М., Саганич А. XML и Java 2. – С-Пб.: Питер, 2001. – 377 с.
4. Сайт комітету стандартів навчальних технологій IEEE (Learning Technology Standars Commettee, Institute of Electrical and Electronics Engineers). <http://ltsc.ieee.org/>

ТЕСТИРУЮЩИЙ ПРОГРАММНЫЙ КОМПЛЕКС «MATEST»

А.Ю. Мельников

г. Краматорск, Донбасская государственная машиностроительная академия
alexandr.melnikov@dgma.donetsk.ua

Тестовый контроль знаний (стартовых, текущих, итоговых) традиционно является неотъемлемой частью как аудиторной, так и самостоятельной работы студентов. В настоящее время разработано множество как универсальных, так и специализированных программ для проведения тестирования, каждая из которых имеет свои преимущества и недостатки. В Донбасской государственной машиностроительной академии на протяжении почти 10 лет используется специальный программный комплекс, который включает в себя:

- тестирующую оболочку для среды ОС Windows (разработано недавно);
- тестирующую оболочку для среды MS-DOS;
- библиотеку тестов;
- транслятор тестов, записанных на специальном языке;
- программу для работы с библиотеками тестов.

Работа с комплексом осуществляется, как правило, в такой последовательности:

1. Составляются вопросы (ответы) и записываются в специальном формате.
2. Запускается транслятор тестов `tscomp`, который или добавляет откомпилированный тест в библиотеку, или (в случае обнаружения ошибок) выдает соответствующее сообщение; протокол работы транслятора выводится в специальный файл.
3. Осуществляется собственно тестирование. Независимо от рабочей среды (MS-DOS или Windows) тестируемый (или преподаватель, если тестирование осуществляется под непосредственным контролем) может определить число задаваемых вопросов, установить время (в секундах), выделяемое на тест, или отключить фактор случайности.
4. При необходимости можно работать с библиотекой тестов с помощью вспомогательной программы `tsbibl`.

Совершенствование программ проводится постоянно, неизменным остается только формат входных данных, по сути, представляющим собой специальный язык [1]. Подробное его описание, как и схема работы транслятора, выходит за границы данной статьи; ниже приведен упрощенный пример – часть теста «Основы языка С». В примере предполагается, что выбор осуществляется цифрами, а каждый вопрос предполагает только один правильный ответ, оцениваемый в 1 балл.


```
name Основы языка C
define "1","2","3"
part 1
score 1,0,0
ask
```

Подключение библиотеки производится командой:

- 1) #include <name.h>
- 2) #include (name.h)
- 3) #define <name.h>

```
endask
score 0,0,1
ask
```

Главная программа задается так:

- 1) void main():integer
- 2) integer main()
- 3) void main()

```
endask
define "1","2"
score 0,1
ask
```

В цикле "for" в качестве индекса может быть использована:

- 1) только целочисленная переменная
- 2) переменная любого типа

```
endask
answer
range 0 0
ans
```

Вы правильно ответили на 0 вопросов.

```
endans
range 1 1
ans
```

Вы правильно ответили на 1 вопрос.

```
endans
range 2 2
ans
```

Вы правильно ответили на 2 вопроса.

```
endans
range 3 3
ans
```

Вы правильно ответили на 3 вопроса.

```
endans
```

Рассмотрим работу тестирующего модуля MatestW (“Matest for Windows”). Сначала загружается библиотека («Загрузка данных / Библиотека» или Ctrl+B), потом выбирается тест («Загрузка данных / Тест» или Ctrl+O) – см. рис. 1. Возможность изменения параметров проиллюстрирована на рис. 2, сам процесс тестирования – на рис. 3, а результаты – на рис. 4.

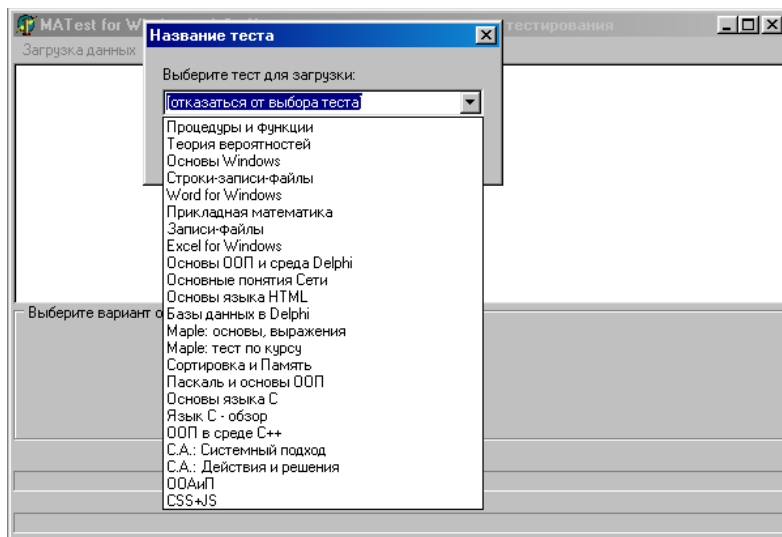


Рис. 1. Выбор теста

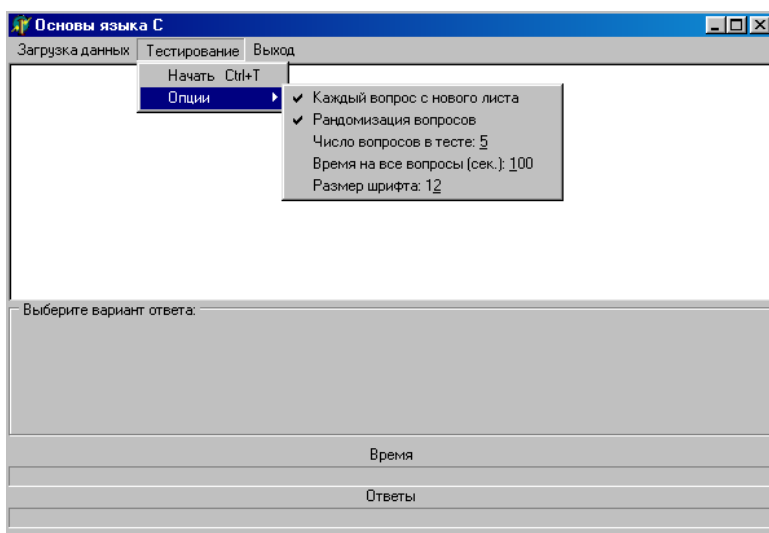


Рис. 2. Выбор параметров

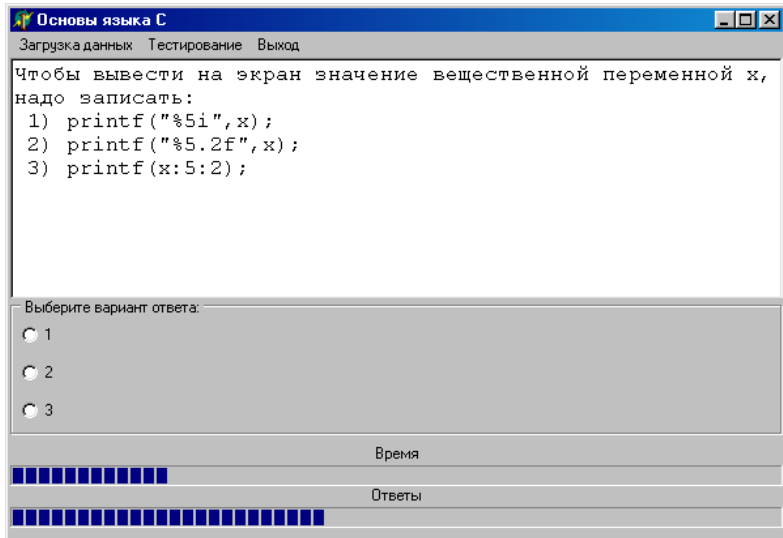


Рис. 3. Тестирование

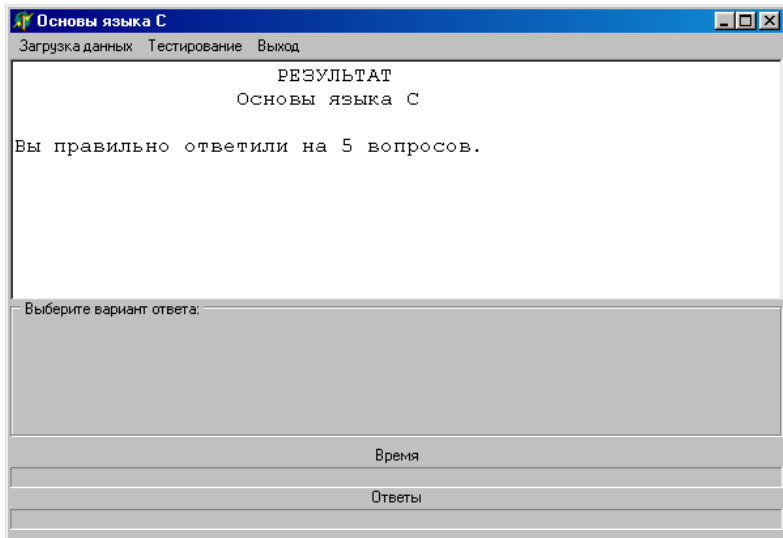


Рис. 4. Результаты

Литература:

1. Рейуорд-Смит В.Дж. Теория формальных языков. Вводный курс: Пер. с англ. – М.: Радио и связь, 1988. – 128 с.

САМОСТІЙНА РОБОТА СТУДЕНТІВ У ПРОЦЕСІ ВИВЧЕННЯ КУРСУ “ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ЕОМ”

О.І. Миронова

м. Луцьк, Волинський державний університет імені Лесі Українки
elena@univer.lutsk.ua

В час активного розвитку науки, суспільного життя підвищуються вимоги до спеціалістів будь-яких спеціальностей. Це призводить до змін у вищій школі, яка від елітарної установи, в якій готують спеціалістів з “конкретними” знаннями, переходить у організацію, що допомагає особистості навчитись набувати знання, мобілізуватись в умовах змін та нових досягнень суспільства. Тому питання самостійної роботи студентів є надзвичайно актуальним та важливим.

У Положенні про організацію навчального процесу у вищих навчальних закладах визначається, що “навчальний час, відведений для самостійної роботи студента, ... повинен становити не менше 1/3 та не більше 2/3 загального обсягу навчального часу студента, відведеного для вивчення конкретної дисципліни”.

Але не дивлячись на це, як зазначає А.М. Алексюк [1], проблема оптимального співвідношення між обов’язковими аудиторними заняттями та часом на самостійну роботу залишається відкритою.

Спробуємо визначити дане співвідношення для одного з основних курсів, визначених у освітньо-професійній програмі вищої освіти за професійним спрямуванням “Прикладна математика” спеціальності “Інформатика” освітньо-кваліфікаційного рівня – бакалавр (програми, згідно якої відбувається підготовка спеціалістів зазначеного рівня, спеціальності та напряму, у Волинському державному університеті імені Лесі Українки), – “Програмне забезпечення ЕОМ”, який входить до циклу професійно-орієнтованих навчальних дисциплін. Врахувавши при цьому, що дане співвідношення визначається з урахуванням специфіки та змісту конкретної навчальної дисципліни, її місця, значення і дидактичної мети в реалізації освітньо-професійної програми [3].

З запланованих 1566 годин на вивчення предметів професійно-орієнтованого циклу, даному курсу відводиться найбільша (270) кількість годин. Його пропонується вивчати після дисциплін “Дискретна математика” та “Програмування” та визначають як такий, що на ньому ґрунтуються не менш важливі дисципліни: “Системне програмування”, “Бази даних та інформаційні системи”, “Мережі ЕОМ”, “Комп’ютерна графіка”, “Операційні системи”, “Автоматизовані системи управління”.

На основі отриманих знань, а саме: структури сучасного програмного забезпечення, поняття баз даних, застосування професійних систем обробки даних, ґрунтується матеріал, що подається в курсах “Бази даних та інфор-

маційні системи”, “Комп’ютерна графіка”; розгляд систем програмування, їх режимів роботи, поняття технологій програмування, асемблерів, трансляторів, методів інтерпретації, вміння розробляти нескладні інтерпретатори лежать в основі курсу “Системне програмування”; знання основ операційних систем (ОС), ресурсів обчислювальної системи, якими керує ОС, принципів роботи, функцій, класифікації ОС є основою навчального матеріалу курсу “Операційні системи”.

Отже, на сучасному етапі вивчення дисциплін інформатики курс “Програмне забезпечення ЕОМ” виступає як інтегруючий та базовий майже для всіх професійно-орієнтованих курсів.

Освітньо-професійні програми постійно змінюються відповідно до все зростаючих потреб сучасності, тому, на нашу думку, з появою нових курсів постає проблема вивільнення годин, визначення основних курсів, а залежно від конкретного навчального закладу, спеціалізації та потреб кожного студента включення у навчальні плани додаткових факультативних розділів [5], ситуація зміниться наступним чином:

- 1) відбудеться підвищення рівня інформаційної культури, яке буде забезпечено переходом на якісно новий рівень викладання інформатики у школах, технічних забезпеченням, збільшенням кількості годин у навчальних планах тощо, з чим буде пов’язана відсутність потреби пояснювати у вузі та знову зупинятися на базових моментах курсу: основних принципах побудови ЕОМ, основах операційних систем, складу пакету офісних програм і т.д.;
- 2) при забезпеченні першої умови частина тем дисципліни “Програмне забезпечення ЕОМ” у вищому навчальному закладі буде вивчена у школі, частину можна буде запропонувати вивчати на спецкурсах, наприклад, курс “Програмне забезпечення комп’ютерних навчаючих систем” (забезпечить знання принципів побудови автоматизованих систем навчання) або внести ці теми у основні курси, такі як “Системне програмування”, “Архітектура ЕОМ”, “Операційні системи” і т.д. (які саме інформаційні компоненти можна внести у відповідні курси зазначалось вище), або нові, наприклад, курс “Програмна інженерія” забезпечить студентів знаннями етапів розробки програм, стандартів на програмну документацію, вміннями проектувати архітектуру програмного забезпечення, розробляти специфікації компонент програмного комплексу, виготовляти програмну та технічну документацію програмного комплексу.

У такій ситуації курс “Програмне забезпечення ЕОМ” або в цілому трансформується у допоміжні елементи інших дисциплін, або решта курсу залишається у такій кількості, що недоцільно відводити для неї цілий курс і можна, наприклад, об’єднати її з деякою іншою дисципліною, або пропонується розширити теми, які охоплює зазначений курс.

Перший варіант у ситуації стрімкого розвитку інформатики як науки та

тем, які повинен охопити курс “Програмне забезпечення ЕОМ”, вбачається більш доцільним.

Зараз вже робляться спроби зміни курсу. Так, наприклад, згідно проекту галузевого стандарту вищої освіти освітньо-професійної програми підготовки бакалавра за напрямом “Прикладна математика” на цикл професійної підготовки виділяється 1836 годин, з них 108 – на курс “Програмне забезпечення ЕОМ” (тобто кількість годин зменшена).

На даному ж етапі, коли вже постала проблема економії годин, але ще не забезпечена умова високого рівня інформаційної культури, пропонується підвищити ефективність вивчення дисципліни “Програмне забезпечення ЕОМ”, виділивши у курсі додаткові компоненти, що можна запропонувати для самостійного вивчення. У такому випадку буде реалізована наступна порада: “від навчального предмета, переповненого конкретною інформацією та все ж таки такого, що лише конспективно викладає науку, повинен відбутися перехід до навчального предмету з надзвичайно узагальненим викладом основ науки. Навчальний предмет стає скоріш “путівником”, ніж “представником” конкретної галузі знань” [2].

Визначимо основні переваги вивчення деяких компонент дисципліни під час самостійної роботи. Вважаємо, що найбільш доцільним є самостійне вивчення програмних продуктів, які складають частину курсу “Програмне забезпечення ЕОМ” та основу багатьох інших дисциплін (наприклад, “Бази даних та інформаційні системи”, “Комп’ютерна графіка” і т.д.).

Перш за все, вивчення програмних систем саме під час самостійної роботи пов’язано з тим, що студенти мають різний рівень (від рівня, коли студент незнайомий з програмою, до професійного рівня) навичок роботи з програмою, що вивчається.

По-друге, така організація вивчення курсу дозволить під час аудиторних занять спинятись на питаннях практичного застосування програмних продуктів, а не на їх функціональних можливостях та взаємодії з користувачем, тобто з предмету вивчення програмний продукт перейде у засіб розв’язання певного класу задач.

По-третє, такий підхід дозволить розглянути більше представників певних класів програмних систем (наприклад, представників текстових редакторів, електронних таблиць, не обмежуючись лише пакетом Microsoft Office) та більш повно ознайомитись з ними.

Важливим моментом є й те, що під час самостійної роботи з певним програмним продуктом формуються практичні навички самостійного вивчення програмних систем взагалі, що є актуальним у зв’язку із високими темпами розвитку програмної індустрії та активним процесом оновлення та виникнення нових програмних продуктів. Неможливо передбачити, які саме системи будуть використовуватись студентом-інформатиком у професійній діяльності, тому особливо важливо навчити його самостійно оволодівати навичками роботи з будь-яким програмним засобом.

Згідно сформульованої гіпотези про те, що програмні продукти доцільно вивчати під час самостійної роботи, виділимо компоненти програми підготовки бакалавра з нормативного курсу “Програмне забезпечення ЕОМ”, які варто запропонувати повністю або частково, або розширюючи тему, на самостійне опрацювання: Програми роботи з файлами; Допоміжні програми (утиліти); Пакети програм, що розширюють можливості ОС; Сервісні системи; Оболонки ОС; Текстові та графічні редактори; Електронні таблиці; Системи управління базами даних; Видавничі системи та їх програмне забезпечення; Програмне забезпечення мереж обчислювальних систем; Пакети програм для математичних розрахунків; Пакети програм для моделювання динамічних процесів; Експертні системи.

Таким чином, ми вважаємо, що згідно програми курсу “Програмне забезпечення ЕОМ”, сформованої в кожному вузі, частина, що припадатиме на вивчення програмних продуктів, може бути перенесена на самостійне опрацювання. Відповідні кафедри самостійно визначатимуть даний компонент, тому не можна визначити числового показника співвідношення між обов’язковими аудиторними заняттями та часом на самостійну роботу. Але згідно аналізу програми курсу “Програмне забезпечення ЕОМ”, можна зробити висновок, що навчальний час, відведений для самостійної роботи студента з даної дисципліни, буде складати близько 2/3 загального обсягу навчального часу, відведеного для вивчення зазначеної дисципліни.

Курс “Програмне забезпечення ЕОМ” є основною дисципліною, під час вивчення якої формуються основні практичні навички роботи з комп’ютером, а саме: робота з ОС та вивчення основних принципів роботи у програмних середовищах. Тому пропонується для вивчення програмних продуктів (які ми пропонуємо вивчати під час самостійної роботи) використовувати навчаючі системи, які забезпечать умову системної інтеграції, що передбачає цілеспрямоване об’єднання існуючих та створюваних інформаційних проектів у цілісну систему [4] та підвищать швидкість набуття навичок роботи з програмними продуктами, так як вони будуть формуватись і під час роботи у середовищі, і під час роботи з навчаючою програмою.

Навчаючі системи дозволять не лише подати теоретичний матеріал (як паперові посібники), а, що є дуже важливим моментом при вивченні програмних продуктів, продемонструвати роботу системи, відразу вийти у середовище, що розглядається, та здійснити контроль за взаємодією користувача з програмним продуктом, своєчасно скорегувавши його дії, що є необхідним компонентом ефективної самостійної роботи.

Таким чином, визначено місце, перспективи розвитку курсу “Програмне забезпечення ЕОМ”, на основі чого зроблені висновки щодо співвідношення самостійної роботи та аудиторних занять та запропоновано як варіант підвищення ефективності вивчення зазначеного курсу та досягнення якісного рівня самостійної роботи використання навчаючих систем.

Література.

1. Алексюк А.М. Педагогіка вищої освіти України. – К.: Либідь, 1998. – 560 с.
2. Беспалько В.П. Основы теории педагогических систем. (Проблемы и методы психолого-педагогического обеспечения технических обучающих систем). – Воронеж: Изд-во Воронежского ун-та, 1977. – 302 с.
3. Болюбаш Я.Я. Організація навчального процесу у вищих закладах освіти: Навчальний посібник для слухачів закладів підвищення кваліфікації системи вищої освіти. – К.: ВВП “Компас”, 1997. – 64 с.
4. Миронова О.І. Самостійна робота студентів в умовах системної інтеграції інформаційних технологій. // Теорія та методика навчання математики, фізики, інформатики: Збірник наукових праць. Випуск 4: В 3-х томах. – Кривий Ріг: Видавничий відділ НметАУ, 2004. – Т.3: Теорія та методика навчання інформатики. – С. 193-199.
5. Рекомендации по преподаванию информатики в университетах: Пер. с англ. – СПб., 2002. – 372 с.

БЛОКИРОВКА СПАМА СРЕДСТВАМИ ТРАНСПОРТНОГО АГЕНТА SENDMAIL ПОД УПРАВЛЕНИЕМ UNIX

А.А. Мясищев, В.М. Полозова
г. Хмельницкий, Хмельницкий национальный университет
alex@tup.km.ua

В настоящее время на почтовые сервера организаций ежедневно приходит много нежелательных писем, которые многим пользователям мешают найти нужную корреспонденцию, а некоторая из них становится ими даже не замеченной. Это можно представить себе в случае, когда за ночь в почтовом ящике накапливается около 100 писем. Такая нежелательная корреспонденция и называется спамом (spam). Кроме того, спам занимает значительную часть внешнего трафика сети, за который организация платит немалые деньги. Более интенсивно почтовые ящики заполняются у тех пользователей, которые пишут свой служебный e-mail адрес на каждом форуме или web-страничках организаций. Рассмотрим способы, позволяющие сократить поток нежелательной корреспонденции на примере Хмельницкого национального университета. Представленный пример позволяет фильтровать ориентировочно до 95% спама, т.е. если раньше, например пользователь alex@tup.km.ua в течении суток получал до 100 спамовских писем, то в настоящее время ему приходит не более 4-5 нежелательных писем. В настоящее время описанный здесь способ фильтрации спама используется в качестве лекционного материала для проведения учебной дисциплины “Ресурсы Интернет”.

Рассмотрим принимаемые действия для решения этой задачи.

1. Блокировка приема почты с серверов – открытых релейев.
2. Блокировка приема почты с серверов, которые не имеют в DNS записей обратных преобразований.
3. Фильтрация почта по именам серверов – пересыльщиков почты.
4. Фильтрация почты по содержанию заголовка (Subject, To).

1. В настоящее время в мире находится большое количество серверов, производящих как рассылку, так и пересылку почтовых сообщений. Те сервера, которые рассылают спам, целесообразно помещать в некоторую базу данных, которая будет использоваться впоследствии для блокировки входящей от них почты. Эту базу данных можно формировать локально на почтовом сервере организации а можно формировать на специальных серверах, разбросанных по всему миру. Первый вариант является менее предпочтительным, т.к. требует большого объема работы по выявлению спамовских серверов и формированию базы данных из них сотрудникам организации. Второй способ втягивает в процесс формирования базы данных большой коллектив специалистов, что оказывается более эффективным в борьбе

со спамом.

Начиная с версии программы sendmail 8.9, которая является транспортным агентом электронной почты в системе Unix, введена поддержка списков DNSBL (DNS-based blacklists – «черный список»). DNSBL – это база данных (список) «открытых релейев» – почтовых серверов, с помощью которых спамеры могут свободно рассылать миллионы писем. Сервера, использующие «черные списки» для блокирования спама, поддерживаются различными организациями как на коммерческой основе, так и в свободном доступе. Первая система такого рода была создана Полом Вики и в настоящее время преобразована в коммерческую службу. Существуют другие, некоммерческие организации, предоставляющие схожий сервис. Рассмотрим использование подобных сервисов на примере системы SPAMHAUS (sbl.spamhaus.org). В системах основанных на DNSBL ip-адреса открытых релейев хранятся в базе DNS сервера в виде записей типа A (Address) в обратном порядке. Например, если сервер имеет ip-адрес 63.211.237.252, то в базе SPAMHAUS будет такая запись:

```
252.237.211.63.sbl.spamhaus.org      IN      A      127.0.0.2
```

В данном случае наличие подобной записи указывает на то, что сервер 63.211.237.252 допускает свободную пересылку почты кем угодно и почта с него должна быть заблокирована. Адрес 127.0.0.2 используется лишь в качестве заполнителя (т.к. поле адреса в DNS не может быть пустым). Если запустить программу nslookup, то можно убедиться в том, что сервер 63.211.237.252 является спамовским:

```
gamma# nslookup
Default Server: gamma.tup.km.ua
Address: 195.46.37.161
```

```
> 252.237.211.63.sbl.spamhaus.org
Server: gamma.tup.km.ua
Address: 195.46.37.161
```

```
Name: 252.237.211.63.sbl.spamhaus.org
Address: 127.0.0.2
```

```
> exit
gamma#
```

Для того, чтобы наш сервер Sendmail блокировал почту с узлов, ip-адреса которых содержатся в базе данных sbl.spamhaus.org, необходимо добавить следующие строки в файл конфигурации sendmail.cf (sendmail 8.12 и выше) сразу после последнего правила набора правил Basic_check_relay:

```
# DNS based IP address spam list sbl.spamhaus.org
R$*          $: ${client_addr}
R$-.$-.$-.$- $: <?> $(dnsbl $4.$3.$2.$1.sbl.spamhaus.org. $: OK $)
R<?>OK      $: OKSOFAR
```

```
R<?>$+<TMP>      $: ТМПОК
R<?>$+            $#error $@ 5.7.1 $: 550 I not recive SPAM!!!-5
```

А в начале файла `sendmail.cf`, там, где идет определения классов (строки, начинающиеся на С) следует добавить в любое место строку – определение файла ключей:

```
Kdnsbl dns -R A -T<TMP>
```

Представленное выше работает следующим образом. Первая строка правила R, получая любое рабочее пространство, заменяет его на текущий ip-адрес SMTP клиента, который подсоединился к нашему серверу в настоящий момент времени. Если ip-адрес представлен например в виде 63.211.237.252, то срабатывает второе правило, согласно которому ip-адрес меняется результатом работы правой части правила. Здесь вызывается преобразование `dnsbl`, которое обращается к классу `dns` (см. строку – определение файла ключей). Класс `dns` с аргументом A по имени компьютера в системе DNS отыскивает его адрес. Имя компьютера с помощью аргумента `$4.$3.$2.$1.sbl.spamhaus.org`. передается классу `dns`. Если ip-адрес определен как 63.211.237.252, то аргумент будет выглядеть так: `252.237.211.63.sbl.spamhaus.org`. Если сервер с этим адресом зарегистрирован в базе, то класс `dns` вернет `127.0.0.2`, второе правила сделает подстановку `<?> 127.0.0.2`, и будет выполнено 5-е правило, согласно которому спамер получит обратно свое письмо с сообщением “550 I not recive SPAM!!!-5”. Если ip-адрес не зарегистрирован в базе данных, то преобразование `dnsbl` ничего не вернет и по умолчанию вторым правилом будет выполнена подстановка `<?>OK`, что сообщает о нормальном завершении набора правил `Basic_check_relay` сообщением `OKSOFAR` и, следовательно, получении нашим сервером почтового сообщения.

Однако один сервер не сможет содержать в своей базе данных перечень всех спамовских серверов, поэтому целесообразно добавит к представленным выше правилам аналогичные правила, которые последовательно проверяли бы на отсутствие в базах данных других серверов ip-адреса открытых релеев. Таким образом, если после представленных выше правил добавить правила

```
# DNS based IP address spam list dnsbl.ahbl.org
R$*          $: ${client_addr}
R$-.$-.$-.$- $: <?> $(dnsbl $4.$3.$2.$1.dnsbl.ahbl.org. $: OK $)
R<?>OK       $: OKSOFAR
R<?>$+<TMP>  $: ТМПОК
R<?>$+      $#error $@ 5.7.1 $: 550 I not recive SPAM!!!-6
```

```
# DNS based IP address spam list psbl.surriel.com
R$*          $: ${client_addr}
R$-.$-.$-.$- $: <?> $(dnsbl $4.$3.$2.$1.psbl.surriel.com. $: OK $)
R<?>OK       $: OKSOFAR
R<?>$+<TMP>  $: ТМПОК
R<?>$+      $#error $@ 5.7.1 $: 550 I not recive SPAM!!!-8
```

то если открытых релеев не будет обнаружено на sbl.spamhaus.org произойдет их проверка на dnsbl.ahbl.org **Ошибка! Недопустимый объект гиперссылки.** в случае отсутствия – на bl.spamcop.net. Обращаясь к некоторому числу таких DNSBL серверов, можно резко сократить получение спамовской корреспонденции. В Хмельницком национальном университете используется обращение к DNSBL серверам, расположенным в следующей последовательности:

```
blackholes.five-ten-sg.com  
cbl.abuseat.org  
t1.dnsbl.net.au  
sbl.spamhaus.org  
dnsbl.ahbl.org  
bl.spamcop.net  
psbl.surriel.com  
bl.csma.biz  
dynablock.njabl.org  
list.dsbl.org  
multihop.dsbl.org  
unconfirmed.dsbl.org  
relays.ordb.org  
dnsbl-2.uceprotect.net
```

Эти сервера были выбраны с сайта <http://www.declude.com> как наиболее активные в блокировании спамовских серверов. Следует заметить, что t1.dnsbl.net.au является некоммерческим для первых 1000 обращений в течение суток, поэтому несмотря на свою достаточно объемную базу, отстоит третьим в списке серверов и на него приходится до 100 обращений в сутки.

2. Опыт работы с почтой показывает, что многие сервера-пересылщики не имеют обратной записи в DNS (записи PTR). Как правило, это спамовские сервера и они распространяют около 35% спамовской корреспонденции, поэтому целесообразно проверку на запись PTR поставить перед обращением к серверам DNSBL.

Добавляем в файл `sendmail.cf` строку – определение файла ключей там же, где для серверов DNSBL:

```
Kdnsname dns -R PTR
```

Сразу после последнего правила набора правил `Basic_check_relay` перед первой проверкой на сервере DNSBL открытых релеев вставляем следующие правила:

```
# DNS based (DNS-name - PTR-record)  
R$* $: ${client_addr}  
R$-.$-.$-.$- $: <?> ${dnsname $4.$3.$2.$1.in-addr.arpa. $: OK $}  
R$* $: ${storage {sav1}} $@ $1 $) $1  
R<?>OK $#error $@ 5.7.1 $: 550 I not recive SPAM!!!-DNS (PTR)  
R<?>$+ $: OKSOFAR
```

Они работают так. Первое правило любое рабочее пространство заменяет на текущий ip-адрес подсоединившегося SMTP клиента. По второму правилу полученный ip-адрес заменяется на DNS имя SMTP клиента, если оно есть или на ОК, если его не существует. В обоих случаях впереди дописывается <?>. Третье правило выполняется для отладки. Оно сохраняет в макросе {sav1} результат второго правила. Для его работы необходимо в sendmail.cf дописать строку

```
Kstorage macro
```

Четвертое правило возвращает спамеру сообщение “550 I not recive SPAM!!!-DNS(PTR)” вместе с его письмом в случае отсутствия обратного DNS имени. Если обратное имя существует, выполняется пятое правило, соответствующее нормальному завершению набора Basic_check_relay.

3. Часто почта посылается с SMTP серверов (до 40%), которые имеют имена типа adsl-68-93-114-237.dsl.rcsntx.swbell.net. Обычно нормальные письма с таких серверов не посылаются. Для блокировки писем с серверов с таким DNS именем можно записать в строке определения файла ключей регулярные выражения, в которые попадают такие имена и отфильтровывать с них прием почты. Добавляем в файл sendmail.cf строку – определение файла ключей

```
Kfrrmail regex -aFRRRMAIL ^(*.*-.*-.*-.*|.*adsl.*|.*dhcp.*)$
```

После последнего правила раздела правил # DNS based (DNS-name – PTR-record) делаем вставку:

```
# DNS based (DNS-name - *-**-*-*)
R$*      $: ${client_addr}
R$-.*-.*-.*-.*-      $: <?> $(dnsname $4.$3.$2.$1.in-addr.arpa. $: OK
$)
R$*      $: $(frrmail $1 $)
R$*      $: $(storage {sav2} $@ $1 $) $1
R$*FRRRMAIL $#error $: "550 I not recive SPAM!!!-From_mail"
R<?>+$+  $: OKSOFAR
```

Здесь после определения имени подсоединившегося SMTP клиента по его ip-адресу (2-е правило), вызывается класс regex и, если выражения найдены по регулярному выражению (3-е правило), то на вход 4-го правила подается строка FRRRMAIL, которая запоминается в макросе {sav2} и передается дальше на вход 5-го правила. Если строка FRRRMAIL существует, письмо отсылается обратно с сообщением "550 I not recive SPAM!!!-From_mail".

В противном случае рассмотренный набор правил заканчивается успешно с получением почтового сообщения.

4. Часто в теме писем (Subject) фигурируют слова, по которым с большой степенью вероятности можно отнести их к спаму. Иногда поле Subject может быть пустым или этот заголовок может полностью отсутствовать. Такие письма также считаем спамовскими. Для выявления “специфических”

слов в заголовке можно воспользоваться регулярными выражениями, записав например в одной строке (без переносов)

```
KSPAMSUBJECT regex -aCONTAINSSHIT love|sex|!!!|penis|SEXUALLY|she|games|money|porn|Ladies|price|Dollar|Music
```

и включить в конец файла `sendmail.cf` набор правил, которые будут проверять заголовок `Subject`:

```
HSubject: $>CheckSubject
```

```
SCheckSubject
```

```
R$* $: $(storage {SubjeckCheck} $@ OK $) $1
```

```
R$* $: $(SPAMSUBJECT $1 $)
```

```
R$*CONTAINSSHIT $#error $: "550 5.7.1 Subject denied"
```

```
R$* $@ OK
```

```
Scheck_eoh
```

```
# Check the macro
```

```
R$* $: < ${SubjeckCheck} >
```

```
# Clear the macro for the next message
```

```
R$* $: $(storage {SubjeckCheck} $) $1
```

```
# Has a Subject: header
```

```
R< $+ > $@ OK
```

```
R$* $#error $: "550 5.7.1 Subject null"
```

Данный набор работает следующим образом. Каждая строчка, которая начинается на `Subject:` попадает на команду `HSubject:`, а от нее на набор правил `CheckSubject`. Первое правило набора запоминает в макросе `{SubjeckCheck}` слово `OK` и передает на второе правило рабочее пространство без изменения (оператор `$1`). Второе правило вызывает регулярное выражение через `SPAMSUBJECT` и если будет найдено в теме письма любое из слов, перечисленных в `SPAMSUBJECT`, письмо будет отброшено с сообщением `"550 5.7.1 Subject denied"`. В другом случае четвертое правило заканчивает проверку набора правил `SCheckSubject`. Рассмотрим набор правил `check_eoh`. Оно вызывается после того, когда будут прочитаны все заголовки. Первое правило заменяет любое рабочее пространство на значение макроса, взятое в угловые скобки. Макрос `{SubjeckCheck}` принимает значение `OK`, если заголовок `"Subject:"` существует и содержит набор символов кроме пробелов. Второе правило обнуляет макрос `{SubjeckCheck}` но передает результат работы первого правила на 3-е правило (оператор `$1`). Если на вход 3-го правила подано `< один_элемент >` (иначе `<OK>`), то произойдет нормальный выход из набора `check_eoh`. В противном случае будет выполнено 4-е правило, согласно которому почта будет отброшена с сообщением `"550 5.7.1 Subject null"`.

Для проверки заголовка `"To:"` в конец файла можно добавить следующие команды и набор правил:

```
HTo: $>CheckTo
```

```
SCheckTo
```

R\$*recipient\$* \$#error \$: "554 Unspecified Mailbox"
R\$*undisclosed\$* \$#error \$: "554 Unspecified Mailbox"

Здесь идет запрет на прием писем “recipient” и “undisclosed”. Обычно это спамовские письма.

Окончательно конфигурационный файл sendmail.cf был сформирован так, что спам блокировался в следующей последовательности (она также справедлива для табл. 1):

1. Отсутствие в DNS записи PTR.
2. Имя подсоединившегося SMTP клиента соответствует регулярному выражению $\wedge(\.*\.*\.*\.*|\.*adsl\.*|\.*dhcp\.*)\$$
3. Сканирование DNSBL серверов в последовательности, представленной выше (всего 14 серверов).
4. Проверка заголовков (Subject: и To:).

В табл. 1 представлены результаты анализа фильтрации почты описанными в работе способами. Данные снимались с 3-х публичных почтовых серверов в течении около 30 часов непрерывной работы.

Таблица 1.

Способ блокировки	сервер-1	сервер-2	сервер-3	сумма по серверам
1. DNS-PTR	373	1292	911	2576
2. Имя SMTP	392	1236	733	2361
3. DNSBL:				
<i>blackholes.five-ten-sg.com</i>	140	382	363	885
<i>cbl.abuseat.org</i>	62	122	78	262
<i>tl.dnsbl.net.au</i>	35	77	59	171
<i>sbl.spamhaus.org</i>	15	2	6	23
<i>dnsbl.ahbl.org</i>	0	1	0	1
<i>bl.spamcop.net</i>	7	23	1	31
<i>psbl.surriel.com</i>	10	6	8	24
<i>bl.csma.biz</i>	0	0	0	0
<i>dynablock.njabl.org</i>	0	0	0	0
<i>list.dsbl.org</i>	1	4	0	5
<i>multihop.dsbl.org</i>	1	0	2	3
<i>unconfirmed.dsbl.org</i>	0	0	0	0
<i>relays.ordb.org</i>	1	0	0	1
<i>dnsbl-2.uceprotect.net</i>	0	1	2	3
4. Заголовки:				
<i>Subject null</i>	2	0	21	23
<i>Unspecified Mailbox</i>	0	0	0	0
<i>Subject denied</i>	0	4	0	4
Всего блокировано	1039	3150	2184	6373
Из них серверами DNSBL	270	618	519	1409

Из табл. 1 видно, что только 1-й и 2-й способы блокировки спама дают возможность его уменьшить примерно на 80%. Поэтому применение серверов DNSBL не приводит к существенному сокращению спама, если до них использованы первые два способа фильтрации.

Литература:

1. Sendmail installation and operation guide. Eric Allman, Gregory Neil Shapiro, Claus Assmann. Sendmail, Inc. For Sendmail Version 8.12. <http://www.sendmail.org/~ca/email/doc8.12/op.html>

2. Михайлов А. Sendmail vs. spam. http://www.i2r.ru/static/486/out_15740.shtml

3. Максим Мамаев. Электронная почта. Технологии Интернет. Лабораторный практикум. <http://athena.vvsu.ru/inetcourse/labs/email.html>

УСТАНОВКА ПОЧТОВОГО СЕРВЕРА ПОД UNIX FREEBSD С ИСПОЛЬЗОВАНИЕМ ТРАНСПОРТНОГО АГЕНТА – ПРОГРАММЫ SENDMAIL

А.А. Мясищев, В.М. Полозова
г. Хмельницкий, Хмельницкий национальный университет
alex@tup.km.ua

Для передачи, получения и обработки почтовых сообщений по сети Интернет используют программы трех типов:

- 1) транспортные агенты электронной почты (MTA – Mail Transport Agent);
- 2) агенты доставки (MDA – Mail Delivery Agent);
- 3) пользовательские агенты.

Транспортный агент обычно работает на почтовом сервере. Для операционной системы Unix в качестве транспортного агента широко используют программу sendmail, которая идет в комплекте поставки с Unix. Рассмотрим установку новой версии sendmail (sendmail 8.13.2) на почтовом сервере организации с DNS именем gamma.tup.km.ua, учитывая следующие условия:

1. Сервер должен принимать почту для доменов tup.km.ua, gamma.tup.km.ua, sens.tup.km.ua, причем для одних и тех же пользователей и пересылать почту на mailhub.tup.km.ua для заданных пользователей.
2. Сервер должен блокировать почту, приходящую с «открытых релей» (антиспамовская защита).
3. База данных доступа sendmail должна обеспечивать доступ к серверу с локальных узлов (с доменами tup и tup.km.ua) и отбрасывать с остальных.
4. Почтовый сервер должен работать с протоколами POP3 и SMTP, т.е. чтобы с ним могли работать пользовательские агенты The Bat, Outlook, установленные на клиентских компьютерах.

В качестве Unix будем в дальнейшем использовать ее версию Unix FreeBSD 4.5. В каталог /usr/home/alex/sendmail8_13 сбрасываем с сервера ftp://ftp.sendmail.org/pub/sendmail/ файл sendmail.8.13.2.tar.gz. Разархивируем его командой

```
%tar xfz sendmail.8.13.2.tar.gz
```

и, перейдя в созданный программой tar подкаталог sendmail-8.13.2, откомпилируем файлы дистрибутива sendmail:

```
%sh Build
%su
Password:
#sh Build install
#cd ./makemap
#sh Build install
#cd ..
```

Далее приступаем к конфигурированию. Для этого рассмотрим сле-

дующие файлы:

1. aliases
2. access
3. local-host-names
4. freebsd.mc

1. Файл `aliases` описывает пользовательские псевдонимы, используемые `sendmail`. Файл расположен в каталоге дистрибутива `/usr/home/alex/sendmail8_13/sendmail-8.13.2/sendmail`, из которого его необходимо скопировать в каталог `/etc/mail`. Он представляет собой набор строк вида

```
имя addr_1, addr_2, addr_3, ...
```

Если в конец файла добавить строки

```
alex: alex, alex@mailhub.tup.km.ua
```

```
all: all, alex@mailhub.tup.km.ua
```

то почта для `alex` и `all` будет скапливаться не только на машине `gamma.tup.km.ua`, но и пересылаться на сервер `mailhub.tup.km.ua` в почтовый ящик пользователя `alex`. После изменения файла `aliases` необходимо выполнить команду

```
#newaliases
```

2. В файле `access` указываются имена компьютеров или их IP адресов, для которых сервер `gamma.tup.km.ua` может принимать и пересылать почту. Он также должен находиться в каталоге `/etc/mail`. Например, компьютеры могут быть перечислены как `REJECT` и `RELAY`. Все почтовые соединения от компьютеров, перечисленных в файле `access` с `REJECT` – отбрасываются, а с `RELAY` – пересылаются. Вместо имен компьютеров могут стоять имена доменов или IP адреса. Если необходимо, чтобы сервер `gamma.tup.km.ua` принимал и пересылал почту с компьютеров, принадлежащих домену `tup.km.ua` или `tup`, а также с IP адресами, начинающимися с `172.20`, файл `access` должен содержать строки

```
tup.km.ua RELAY
```

```
tup RELAY
```

```
172.20 RELAY
```

Для активизации созданного или измененного файла `access` необходимо выполнить команду

```
#makemap hash /etc/mail/access.db < /etc/mail/access
```

3. В файле `local-host-names`, который расположен в каталоге `/etc/mail`, указывается список имен компьютеров, для которых программа `sendmail` на сервере `gamma.tup.km.ua` будет принимать почту.

```
gamma.tup.km.ua
```

```
sens.tup.km.ua
```

```
tup.km.ua
```

Здесь сервер `gamma.tup.km.ua` будет для самого себя, домена `tup.km.ua` и машины `sens.tup.km.ua` принимать почту при условии, что одноименный пользователь существует на компьютере `sens.tup.km.ua`, а в файле зоны DNS сервера существуют записи:

tup.km.ua.	IN	MX	10 gamma.tup.km.ua.
sens.tup.km.ua.	IN	A	195.46.37.168
sens.tup.km.ua.	IN	MX	10 gamma.tup.km.ua.

Из записей видно, что компьютер sens.tup.km.ua имеет свой ip-адрес, но почту для него будет получать gamma.tup.km.ua.

4. Центральное место в настройке sendmail принадлежит конфигурационному файлу sendmail.cf. Он также должен быть расположен в каталоге /etc/mail. Для упрощения его конфигурирования используют файл freebsd.mc, который обрабатывают с помощью препроцессора m4. Файл freebsd.mc значительно проще sendmail.cf и может содержать всего несколько строк. В дистрибутиве Unix FreeBSD файл freebsd.mc находится в каталоге /etc/mail для соответствующей реализации FreeBSD, а в дистрибутиве sendmail файлы препроцессора m4 для различных Unix – систем расположены в каталоге /usr/home/alex/sendmail8_13/sendmail-8.13.2/cf/cf. Скопируем файл freebsd.mc из каталога /etc/mail в каталог /usr/home/alex/sendmail8_13/sendmail-8.13.2/cf/cf. Для сервера gamma.tup.km.ua он имеет вид

```

1 divert(0)dnl
2 VERSIONID(`$FreeBSD: src/etc/sendmail/freebsd.mc,v 1.10.2.11
2001/07/14 18:07:27 gshapiro Exp $')
3 OSTYPE(freebsd4)dnl
4 DOMAIN(generic)dnl
5 FEATURE(access_db, hash -T<TMPF> /etc/mail/access)dnl
6 FEATURE(blacklist_recipients)dnl
7 FEATURE(mailertable, `hash -o /etc/mail/mailertable')dnl
8 FEATURE(local_lmtp)dnl
9 FEATURE(dnsbl, `blackholes.five-ten-sg.com', `550 I not receive
SPAM!!!-1')dnl
10 FEATURE(dnsbl, `cbl.abuseat.org', `550 I not receive SPAM!!!-
2')dnl
11 FEATURE(`dnsbl', `t1.dnsbl.net.au', `550 I not receive SPAM!!!-
3')dnl
12 FEATURE(always_add_domain)dnl
13 define(`confMAX_MESSAGE_SIZE',100000)dnl
14 define(`confDOMAIN_NAME', `gamma.tup.km.ua')dnl
15 define(`confBIND_OPTS', `WorkAroundBrokenAAAA')dnl
16 define(`confNO_RCPT_ACTION', `add-to-undisclosed')dnl
17 define(`confPRIVACY_FLAGS', `authwarnings,noexpn,novrfy')dnl
18 MAILER(local)dnl
19 MAILER(smtp)dnl

```

Здесь каждая строка пронумерована для пояснения, т.к. для распечатки использовалась команда cat -n freebsd.mc. Опишем некоторые команды в соответствии с их номерами.

1. divert(0) – строки, стоящие после, являются командами, divert(1) – комментариями. dnl – удаляет все символы до символа перевода строки включительно.

2. Макрос VERSIONID автоматически встраивает информацию о версии файлов m4.

3. OSTYPE указывает имя файла в каталоге `./ostype`, который содержит макросы, соответствующие данной архитектуре UNIX-системы. Например, местоположение и аргументы локального агента доставки почты.

4. DOMAIN указывает на имя файла, который находится в каталоге `./domain` и содержит общесистемную базовую информацию для домена. Например, возможность использования файла `local-host-names` и средства `redirect`, указание пути к файлу `.forward`, установление максимальной длины заголовка письма.

5. Это макроопределение m4 включает использование базы данных доступа. Благодаря этой функции, можно в базе данных `access` разрешать или запрещать прием почты из определенных доменов.

6. Включает возможность блокирования входящей почты от определенных отправителей, компьютеров и адресов. Описание соответствующих блокировок выполняется в файле `access`.

7. Это макроопределение включает возможность использования "mailtable" (базы данных выбора нового агента доставки). `mailtable` совместно с файлом `/etc/mail/mailtable.db` (базой данных) связывает имена (например `beta.tup.km.ua`) со специальными агентами доставки. Благодаря этой возможности, почта может доставляться специфическими агентами доставки к новым доменным именам. Обычно, эта возможность используется только на центральном почтовом концентраторе. Например, если машина `beta.tup.km.ua` имеет локальный ip-адрес, не имеет записи A в DNS, но имеет запись MX

```
beta.tup.km.ua.           IN      MX      10 gamma.tup.km.ua.
```

то для пересылки почты от `gamma.tup.km.ua` до `beta.tup.km.ua` по локальной сети необходимо создать файл `/etc/mail/mailtable` со следующими строками:

```
.tup.km.ua      smtp:[beta.tup.km.ua]
```

и выполнить команду

```
#makemap hash /etc/mail/mailtable.db < /etc/mail/mailtable
```

8. Это макроопределение определяет, что в качестве локального агента доставки используется программа `mail.local`, расположенная по умолчанию в каталоге `/usr/libexec`.

9, 10, 11. Макроопределения разрешает `sendmail` отклонять почту от любых сайтов, входящих в базу данных «открытых релеев» (DNS-based blacklists). В рассматриваемом конфигурационном файле `freebsd.mc` выполняется последовательное обращение к трем DNSBL серверам – `blackholes.five-ten-sg.com`, `cbl.abuseat.org`, `t1.dnsbl.net.au`. Если в базе данных любого из них будет обнаружен ip-адрес SMTP клиента, который подключился к серверу `gamma.tup.km.ua`, письмо будет заблокировано для приема и отправлено обратно с сообщением "550 I not receive SPAM!!!-2".

12. Включает использование "always_add_domain", т.е. добавляет ло-

кальный домен в локальной почте. Благодаря этой возможности все локально доставляемые адреса будут полностью квалифицированными (содержать также и домен).

13. Макроопределение определяет максимальный размер сообщения в байтах, которое будет принято sendmail (1000000 байт).

14. Определяет полное доменное имя машины – gamma.tup.km.ua.

15. Программа sendmail вначале просматривает записи AAAA сервера имен IPv6, а затем записи A. Все сервера имен должны вернуть NODATA, если хост найден, но нет записи AAAA. Некоторые сервера имен могут работать некорректно и когда выполняется запрос записи AAAA, а ее там нет, то выдается сообщение SERVFAIL и sendmail почтовое сообщение устанавливает в очередь для последующей обработки. Таким образом, почта может быть вообще не отправлена. Для исключения этого добавляется аргумент WorkAroundBrokenAAAA в опцию ResolverOptions, что для препроцессора m4 представляется макроопределением `define(`confBIND_OPTS', `WorkAroundBrokenAAAA')`.

16. Если на gamma.tup.km.ua было послано почтовое сообщение, в котором отсутствует заголовок “To:”, то согласно макроопределению он будет сформирован в виде “To: undisclosed-recipients:;”.

17. Увеличивает безопасность демона sendmail. В нашем случае запрещает SMTP команды EXPN и VRFY, разрешает заголовок “X-Authentication-Warning:”

18, 19. Макроопределение включает агенты доставки local и smtp для программы sendmail.

После формирования файла freebsd.mc обрабатываем его препроцессором m4 для получения конфигурационного файла sendmail.cf. Для этого необходимо находиться в каталоге /usr/home/alex/sendmail8_13/sendmail-8.13.2/cf/cf и выполнить команду:

```
#m4 ../m4/cf.m4 freebsd.mc > /etc/mail/sendmail.cf
```

Окончательно, запуск программы sendmail выполняется командой

```
#sendmail -bd -q30m
```

Здесь -bd – работа в режиме демона, -q30m – каждые 30 минут обрабатываются все стоящие в очереди сообщения.

Однако перед запуском sendmail с новым конфигурационным файлом целесообразно провести тестирование с помощью команды

```
#sendmail -bt -C /etc/mail/sendmail.cf
```

sendmail запускается в диалоговом тестовом режиме. Ожидаемый ввод:

```
>номер_набора рабочее пространство
```

Программа произведет разбор *рабочего пространства* с помощью указанного *набора правил*. Например проверим, находится ли ip-адрес 222.116.94.72 подключившегося SMTP клиента в базе данных «открытых релеев»

```
# sendmail -bt -C /etc/mail/sendmail.cf
```

```
ADDRESS TEST MODE (ruleset 3 NOT automatically invoked)
```

```

Enter <ruleset> <address>
> .D{client_addr}222.116.94.72
> Basic_check_relay <>
Basic_check_rela input: < >
Basic_check_rela returns: $# error $# 5 . 7 . 1 $: 550 I not receive
SPAM !!! -1
>

```

Проверка показала, что клиент SMTP с адресом 222.116.94.72 в момент написания статьи зарегистрирован в базе данных сервера “blackholes.five-ten-sg.com”

Рассмотрим установку сервера POP3 на машине gamma.tup.km.ua. Для этого копируем с сайта ftp://mirror.aarnet.edu.au/pub/eudora/servers/unix/popper/ дистрибутив qpopper4.0.5.tar.gz. Разархивируем его командой

```
#tar xzf qpopper4.0.5.tar.gz
```

И перейдя в созданный программой tar подкаталог /usr/home/alex/pop/qpopper4.0.5, откомпилируем файлы дистрибутива qpopper:

```

#./configure
#make
#make install

```

После этого необходимо зайти в файл /etc/inetd.conf и отредактировать строку, чтобы она имела вид:

```
pop3 stream tcp nowait root /usr/local/sbin/popper popper
```

и перезапустить демон inetd

```

#ps ax | grep inetd
170 ?? Is 0:00.01 /usr/sbin/inetd -wW
#kill -HUP 170

```

Для проверки установки выполним команду

```

# telnet gamma.tup.km.ua 110
Trying 195.46.37.161...
Connected to gamma.tup.km.ua.
Escape character is '^]'.
+OK Qpopper (version 4.0.5) at gamma.tup.km.ua starting.
quit
+OK Pop server at gamma.tup.km.ua signing off.
Connection closed by foreign host.
#

```

Данный листинг свидетельствует о правильности установки POP-сервера. Затем настраивается почтовая программа (The Bat или Outlook) на клиентской машине. В качестве POP3-сервера и SMTP-сервера устанавливаем машину 195.46.37.161 (gamma.tup.km.ua). На этом установка почтового сервера организации завершается.

Литература:

1. Sendmail installation and operation guide. Eric Allman, Gregory Neil Shapiro, Claus Assmann. Sendmail, Inc. For Sendmail Version 8.12.

<http://www.sendmail.org/~ca/email/doc8.12/op.html>

2. Михайлов А. Sendmail vs. spam. http://www.i2r.ru/static/486/out_15740.shtml

ВИКОРИСТАННЯ ІННОВАЦІЙНИХ ТЕХНОЛОГІЙ НА ПЕРШИХ УРОКАХ З ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ В КЛАСАХ З ПОГЛИБЛЕНИМ ВИВЧЕННЯМ ІНФОРМАТИКИ

І.В. Настенко

м. Київ, Національний педагогічний університет імені М.П. Драгоманова
inastenko@mail.ru

В умовах вдосконалення системи шкільної освіти особливо актуальним є посилення зв'язку між інноваціями в науці та педагогіці і змістом навчання. Сучасна тенденція до збільшення частки самостійної навчальної та наукової роботи учнів із одночасним зменшенням аудиторного навантаження вимагає посиленої уваги до використання інноваційних технологій під час навчання учнівської молоді інформатиці, що сприяє активізації пізнавальної діяльності, розвитку навичок і вмінь самостійної роботи, цілеспрямованості, ініціативності, відповідальності, вольових якостей, формуванню внутрішньої мотивації. Досягнення мети навчити учнів не лише знанням, а навчити їх навчатися можливе лише за допомогою інноваційних технологій навчання інформатиці, особливо в таких складних її розділах, як “Об'єктно-орієнтоване програмування” (ООП).

Об'єктно-орієнтоване програмування в наш час займає провідне місце в програмах класів з поглибленим вивченням інформатики. Вивчення цієї теми сприяє як підготовці високо кваліфікованих майбутніх професіоналів, так і є тим елементом, що сприяє розвитку інтелектуальних здібностей учнів.

За програмою для загальноосвітніх навчальних закладів, ліцеїв, гімназій, спеціалізованих шкіл з поглибленим вивченням інформатики (8-11 класи, автори: Жалдак М.І., Морзе Н.В., Мостіпан О.І.), тематика перших уроків з об'єктно-орієнтованого програмування наступна [1]:

1. Основні поняття об'єктно-орієнтованого програмування (2 год.);
2. Вступ до об'єктно-візуальне програмування. Середовище програмування (4 год.);

Для пояснення нового матеріалу, формування відповідних вмінь та навичок, можна, звичайно, користуватися стандартними методами та прийомами, але використовуючи інноваційні технології (як педагогічні, так і комп'ютерні), ми значною мірою можемо покращити результати навчання, зробити навчання більш продуктивним, і, що дуже важливо, більш особистісно-орієнтованим. Особистісно-орієнтоване навчання на даному етапі розвитку освіти було, є і залишиться досить актуальним, оскільки дозволяє сформувати в учня ті якості особистості, які в наш час потрібні людині для повноцінного особистого і професійного життя. На думку Є.С. Полат [2], такими якостями є гнучке адаптування людини в життєвих ситуаціях, що змінюються, самостійне здобування необхідних знань, вміння застосовувати

їх на практиці для вирішення проблем різного роду, використовуючи сучасні технології; чітке усвідомлення, де і як ті знання, що були придбані, можуть бути використані в навколишній дійсності; спроможність генерувати нові ідеї, творчо мислити; грамотне опрацювання інформації та ін. Також важливо навчити учнів бути комунікабельними та контактними в різних соціальних групах, вміти спільно працювати, попереджаючи конфліктні ситуації.

Формування згаданих якостей особистості є задачею не скільки змісту освіти, скільки технологій, які при цьому використовуються. Тільки тоді, коли нові педагогічні технології будуть широко застосовуватись разом з новими інформаційними технологіями, можлива найбільш ефективна реалізація можливостей, що в них закладено.

Серед найбільш доцільних інноваційних педагогічних технологій з різних спрямувань є: навчання в співробітництві, метод проектів, різнорівневе навчання, “портфель учня”, індивідуальний підхід до навчання, диференційований підхід [2]. Коли наведені технології будуть впроваджуватись разом з використанням нових інформаційних технологій, матимемо можливість в повному обсязі розкрити педагогічні та дидактичні функції цих методів, реалізувати закладені в них потенціальні можливості.

Наведемо приклад використання інноваційних технологій до формування вмінь учнів (за відповідними вимогами до результату навчальної діяльності учнів [1]) на уроках, присвячених темі “Вступ в об’єктно-візуальне програмування. Середовище програмування”.

<i>Вид уміння, що потрібно сформулювати</i>	<i>Методична задача уроку</i>	<i>Вид діяльності</i>	<i>Засоби навчання/вчитель</i>
Завантажувати середовище об’єктно-орієнтованого програмування	Сприйняття, розуміння	Індивідуально	учитель
Змінювати властивості об’єктів типу “форма”, “мітка”, “текстове поле”, “малюнок”	Використання за зразком	Співробітництво в парах	Завдання в файлах типу doc
	Використання за аналогією	Співробітництво в парах	Завдання в файлах типу doc, приклад виконання – pas, dpr, dfm, dcu
	Перевірка роботи в малих групах	Обговорення результатів виконання	учитель
Визначати типи змінних	Сприйняття, розуміння	Перевірка розуміння фронтально, в співробітництві (в трійках/четвірках)	Підручник, тестові завдання (розроблені за допомогою Delphi чи html [3])

<i>Вид умінь, що потрібно сформулювати</i>	<i>Методична задача уроку</i>	<i>Вид діяльності</i>	<i>Засоби навчання/вчитель</i>
Коректно записувати арифметичні вирази	Виконання завдання за аналогією	Співробітництво в парах	Завдання в файлах типу doc
	Контроль	Індивідуально	Роздатковий матеріал
Записувати та зберегти вихідний код програми у середовищі Delphi	Сприйняття, розуміння	Індивідуально	учитель

Коли вчитель активізує опорні знання, перевіряє сприйняття та розуміння викладеного матеріалу, складає запитання тестових завдань, вважаємо доцільним привертати значну увагу на побудову питань, що ставляться перед учнем чи групою учнів, а саме: користуватися ще однією відомою інновацією – систематикою моделювання запитань (по ключовим словам) за Блумом [4].

Найефективнішою інноваційною педагогічною технологією при викладенні даної теми вважаємо навчання в співробітництві, сутність якого полягає в наступному: учні працюють в невеликих групах, відповідають за успіхи кожного, допомагають один одному, що є особливо важливим на перших кроках усвідомлення понятійного апарату нової теми. Розрізняють декілька варіантів навчання в співробітництві [2]: Student Team Learning – навчання в команді, Cooperative Learning, Learning Together – навчаємося разом, дослідницька робота учнів в групах. При цьому, незалежно від варіанту, є одне завдання на групу, одне заохочення на групу та розподіл ролей в групі. Оцінювання учнів диференційоване, тобто учні навчаються за силою своїх можливостей та здібностей і відповідно мають шанс бути оцінені на рівні всіх останніх (при умові, що і слабкий, і сильний учень прикладають максимум зусиль для досягнення свого рівня). На відміну від традиційних методів, тут оцінюються зусилля, які роблять учні для досягнення спільного результату, що дозволяє якомога вище підняти рівень мотивації навчальної діяльності.

Всі наведені варіанти навчання в співробітництві при правильному їх використанні дозволяють підготувати учнів до роботи над тим чи іншим проектом (використання чого є доцільним при викладенні курсу ООП), що являє собою досить складний етап, як з боку тих, хто навчається, так і тих, хто навчає. Метод проектів вимагає від вчителя акумулювати зусилля на створення такого проекту, що буде сприяти розвитку в учнів навичок високого рівня. Основними вимогами до використання методу проектів при цьому будуть [4]: наявність значущої в дослідницькому або творчому плані проблеми чи задачі; практична, теоретична, пізнавальна значущість передбачуваних результатів; самостійна (індивідуальна, парна, групова) діяльність учнів; визначення кінцевої мети проекту; визначення базових знань з

різних галузей, необхідних для роботи над проектом; структурування змістової частини проекту; використання дослідницьких методів; результати виконання проекту мають бути матеріальними, тобто певним чином оформлені.

На даному етапі розгляду теми вважаємо доцільним обрати короткочасний (може бути розроблений на декількох уроках) міні-проект ознайомчо-орієнтовного типу, який ще називають інформаційним (спрямований на збір інформації про об'єкт чи явище, передбачається ознайомлення учасників проекту з цією інформацією, її аналіз. Потребує добре продуманої структури, можливості систематичного корегування протягом опрацювання). При цьому кількість учасників може варіюватися від двох до групи.

Література:

1. Інформатика. Програми для загальноосвітніх навчальних закладів. – Запоріжжя: Прем'єр, 2003. – 304 с.
2. Новые педагогические и информационные технологии в системе образования: Учеб. пособие для студ. пед. вузов и системы повыш. квалиф. пед. кадров / Е.С. Полат, М.Ю. Бухаркина, М.В. Моисеева, А.Е. Петров; Под ред. Е.С. Полат. – М.: Издательский центр «Академия», 2001. – 272 с.
3. Тарасов В.А., Тарасов В.В. Разработка контролирующих html-документов // Информатика и образование. – 2001. – №3.
4. Intel® Навчання для майбутнього. – К. Видавнича група ВНУ, 2004. – 416 с.

ОСОБЕННОСТИ ПРОВЕДЕНИЯ МОДУЛЬНЫХ КОНТРОЛЬНЫХ РАБОТ В ФОРМЕ ТЕСТА

Е.А. Панина, Л.В. Гусева
г. Харьков, Академия гражданской защиты Украины
fd.apbu@list.ru

Одним из видов текущего контроля знаний, успешно используемого для тематической проверки знаний, умений и навыков при изучении отдельных тем, которым посвящаются лекции, семинарские и лабораторные занятия, является контроль знаний в виде тестов.

Один из способов программированного контроля – это стандартизированный или текстовый контроль. Он состоит в предъявлении обучаемому серии вопросов с вариантами ответа на каждый вопрос, причем среди них один или несколько правильные.

Задача тестируемого состоит в указании правильного ответа.

Тестовые задания, сформулированные по темам и используемые на практических и семинарских занятиях, делают работу студентов и преподавателя более продуктивной. Преподаватель за небольшой промежуток времени (10–15 минут) успевает опросить каждого студента группы по всей теме. Это позволяет индивидуализировать работу со студентами, осуществить контроль знаний каждого учащегося на всех этапах обучения.

В этом учебном году в рамках эксперимента по Болонскому процессу в Академии гражданской защиты Украины некоторые дисциплины преподавались по кредитно-модульной системе. По окончании каждого модуля предполагалось проводить опрос слушателей с их последующей оценкой.

В АГЗУ по дисциплине «Информатика и компьютерная техника» разработаны тесты для проверки знаний слушателей по следующим темам: «Аппаратное и программное обеспечение ПК», «Обработка текстовой документации в MS Word», «Анализ статистической информации средствами MS EXCEL», «Современные мультимедийные системы представления информации» и другие.

Для создания тестов и проверки знаний использовались две взаимно независимые системы оценки знаний – для итоговой оценки знаний использовалась автоматизированная система Test2000. Для оперативного контроля, который предполагает быстрый способ оценивания знаний по ходу учебного процесса, использовалась система дистанционного образования MOODLE [1], которая размещена на локальном сайте академии. Цель данной системы – выявить пробелы в знаниях отдельных обучаемых и групп в целом, а также корректировка траектории обучения.

Система MOODLE позволяет создавать дистанционные курсы, которые содержат как материал, излагаемый в курсе, так и удобные средства контроля.

В систему встроен редактор тестов, который предлагает широкий набор вариантов контрольных вопросов:

- вопросы с несколькими вариантами ответов;
- вопросы типа «да-нет»;
- вопросы на соответствие;
- вычисляемые вопросы и т.д.

При создании тестов можно ограничить время тестирования и количество попыток сдачи теста, выбрать тип оценивания слушателя, разрешить/запретить просмотр правильных ответов во время тестирования и т.д.

После создания теста из базы данных вопросов можно создать набор вопросов для теста. Во время тестирования, вся информация сохраняется в базе данных слушателей, и преподаватель может просмотреть результаты работы слушателя по подготовке к сдаче модульного контроля.

Для сдачи итогового модульного контроля использовалась система TEST2000, которая также содержит базу данных вопросов, из которой система самостоятельно выбирает заданное количество вопросов. Система позволяет установить ограничение по времени на сдачу теста, сохраняет результаты тестирования в базе данных.

Использование тестов позволяет охватить весь материал модуля, что мотивирует слушателей к изучению всего материала курса.

К достоинствам системы тестирования следует отнести:

1. Наличие единого для всех тестируемых комплекса заданий, подготовленного по определенным требованиям.
2. Тестовая система может выступать в качестве тренинга, а также средства систематизации и закрепления навыков слушателей.
3. Тест удобен, поскольку экономит время и силы на стадии его проведения и позволяет получить результаты сразу после его проведения, тем самым устраняется «человеческий фактор» в оценке тестируемого.
4. Тестирование почти не уступает другим формам вступительного экзамена при проверке знаний по орфографии, пунктуации, культуре речи.

В качестве недостатков следует отметить:

1. Компьютерное тестирование не предполагает наличия творческих заданий, что существенно снижает его ценность как формы модульного контроля. Без творческих заданий процесс подготовки к тесту будет представлять собой «натаскивание» на тест, превращая слушателя в «ответающую машину».

2. Тестовая система проверяет не столько знания и навыки слушателя, сколько его память, способность логически мыслить и интуитивно выбирать правильный ответ, опираясь на догадку, в результате чего ведущий параметр теста – адекватность (тест должен проверять то, что мы хотим проверить) – не будет реализован.

3. Необходимость подготовки специалистов-тестологов. Для того чтобы тест реализовывал три основных параметра: экономичность, надеж-

ность и валидность, необходима целенаправленная подготовка специалистов-тестологов, которые занимались бы созданием и апробацией тестов. Только при этом условии можно будет достигнуть однозначной формулировки тестового задания, не допускающей двусмысленности, корректного подбора материала. Недостаточно продуманные задания будут вызывать затруднения при выполнении.

4. Недостаточная разработанность критериев оценки теста. Важна и проблема оценивания теста: сколько баллов тестируемый получает за каждое задание, учитывается ли уровень сложности задания (если учитывается, то как), сколько баллов тестер имеет право «снять» за допущенные ошибки. Поэтому для всех тестов должны быть установлены критерии оценки, коэффициент сложности (как для целого теста, так и для отдельного задания) и классификация ошибок. А это значит, что все тесты должны быть стандартизированы.

5. Отсутствие стандартизации тестов. Стандарт предполагает диагностическое описание минимальных обязательных требований, в соответствии с которым будет проводиться проверка и сопоставление фактического уровня знаний с требованиями к ним. Эта часть работы пока не выполнена.

Литература:

1. www.moodle.org.

ВИВЧЕННЯ MS EXCEL У КУРСІ «ІНФОРМАТИКА»

Ю.М. Пилипенко, Л.П. Голубєв

м. Київ, Київський національний університет технологій і дизайну
Golubev@ukrpost.net

MS Excel – один з кращих продуктів MS Office. Його можливості і простота при проведенні різноманітних розрахунків, побудові графіків і веденні «бухгалтерських» книг роблять цей програмний продукт незамінним серед величезної кількості користувачів. Тому ми вважаємо обов'язковим включити основи роботи з MS Excel у програму курсу «Інформатика».

При першому знайомстві з MS Excel доцільне включення наступних розділів:

1. Обчислення в MS Excel.
2. Побудова графіків.
3. Уточнення коренів рівнянь.
4. Рішення задач математичного програмування і систем рівнянь.
5. Побудова і робота з найпростішими базами даних.

У даній роботі даються рекомендації з вивчення цих тем.

Розглянемо більш докладно методику освоєння матеріалу.

Відзначимо, що запропоновані нами завдання побудовані таким чином, що, виконавши їх, студент здобуває практичні навички по відповідному розділі. Наприкінці кожної теми обов'язково присутні контрольні питання.

1. Обчислення в MS Excel

Здійснити табулювання функції на обраному студентом проміжку. При цьому відрізок табулювання необхідно підібрати так, щоб забезпечити обчислення, по кожній галузі функції.

Використовується функція виду $y = \begin{cases} f_1(x), & \text{якщо } x \in (-\infty; x_1] \\ f_2(x), & \text{якщо } x \in (x_1; x_2) \\ f_3(x), & \text{якщо } x \in [x_2; +\infty) \end{cases}$.

Остаточна таблиця повинна мати «східчастий» вид:

Початок табуляції	Кінець табуляції	Кількість інтервалів табуляції	Крок табуляції	
a	b	k	h	
аргумент	$f_1(x)$	$f_2(x)$	$f_3(x)$	y

Так як функції $f_1(x)$, $f_2(x)$, $f_3(x)$ обчислюються тільки на конкретних підінтервалах $[a; x_1]$, $(x_1; x_2)$, $[x_2; b]$ інтервалу табуляції, у той час як функція y обчислюється на інтервалі $[a; b]$.

Число k (кількість інтервалів табуляції) задається викладачем, а крок

табуляції h обчислюється за формулою $h=(b-a)/k$ (звичайно $k = 20$).

Які цілі ми переслідуюмо при виконанні даного завдання:

- навчити працювати з майстром функцій. Підбираючи в якості $f_1(x)$, $f_2(x)$, $f_3(x)$ досить різноманітні математичні функції, можна домогтися того, щоб студент познайомився з тим, як необхідно їх записувати;
- навчити працювати з логічною функцією «ЕСЛИ», аналог якої IF...THEN...ELSE використовується в VBA;
- навчити працювати з маркером автозаповнення;
- навчити правильно користуватися абсолютними, відносними, змішаними адресаціями.

Відзначимо, що стовпчик таблиці «аргумент» зобов'язаний бути заповнений формулами з посиланнями на відповідні комірки, так щоб застосування початку чи кінця інтервалу табуляції відразу приводили до модифікації всієї побудованої таблиці.

2. Побудова графіків

При виконанні завдання студент повинен навчитися будувати графіки протабульованих раніше функцій, використовуючи «Майстер діаграм» [2]. Усі графіки повинні бути розташовані на одній координатній площині.

Необхідно звернути увагу на те, що:

1. Графіки і діаграми – це візуальне відображення інформації, що отримана раніше!
2. Між графіками і даними існує динамічний зв'язок – будь-яка зміна в даних відразу приводить до зміни в графіках і діаграмах.
3. Який тип обраної діаграми і де розташовуються ряди даних – у рядках чи стовпцях. Відзначимо, що для побудови графіків, нам здається, є сенс віддати перевагу «точковій» діаграмі, щоб перший стовпчик таблиці служив інформацією для розмітки вісі аргументів.
4. Де і як формується «легенда», і як її можна відкоригувати.
5. Як роздрукувати результат, розмістивши графіки і таблицю з даними на одному робочому листі. Зверніть увагу на обов'язковість «попереднього перегляду» перед друком.

3. Уточнення коренів рівняння

Пропонується, використовуючи «Подбор параметра», знайти один з коренів рівняння $g_1(x)=g_2(x)$, після чого побудувати графік функцій $g_1(x)$, $g_2(x)$, $g_1(x)-g_2(x)$, тобто лівої і правої частин рівняння, а також їхньої різниці на відрізьку, що містить корінь.

При виконанні роботи студент повинний оформити результат у виді таблиці:

Корінь	Рівняння	Вид рівняння	Початкове наближення кореня

Необхідно звернути увагу на те, що:

1. Знайдений корінь може залежати від того, яке початкове наближення до кореня ми вибираємо (відзначимо, що до включення функції «Подбор

- параметра» у графах «Корінь» і «Початкове наближення кореня» стоять ті самі числа). Тому обов'язково наявність графі «Початкове наближення кореня», щоб знати, від якого числа стартує ітераційний процес.
- У графі «Рівняння» повинна стояти функція Excel, рівна різниці лівої і правої частин, аргументи якої посилаються на «Корінь».
 - Графа «Вид функції» не є функцією Excel, а є коментарем, щоб було зрозуміло, яке рівняння розглядається. Звичайно, тут можна застосувати «Редактор формул».
 - Після застосування «Подбора параметра» у графі «Корінь» стоїть уточнення кореня.
 - При побудові графіків (а це не що інше, як закріплення перших двох тем) звернути увагу, як «уточнюються» графіки при розгляді більш «вузьких» відрізків, що містять корінь.

4. Розв'язання задачі математичного програмування і систем рівнянь

Пропонується розв'язати двовимірну задачу лінійного програмування типу планування виробництва за допомогою функції «Поиск решения». При цьому задача формулюється як задача в змістовних термінах з обмеженнями по ресурсах, по якій необхідно:

- скласти математичну модель;
- графічно розв'язати задачу в зошиті.

Розв'язання задачі в зошиті дозволяє краще зрозуміти, як знаходиться і де лежить розв'язок задачі лінійного програмування, та, крім того, є контрольним прикладом проведених досліджень. Нами пропонується наступне оформлення таблиці з даними і функціями-обмеженнями.

x_1	x_2	Цільова функція		
Коефіцієнт при x_1	Коефіцієнт при x_2	Ліва частина обмеження	Вид обмеження	Права частина обмеження

При такому оформленні легко і зрозуміло, як складені обмеження, а також, який їхній вид.

Звертаємо увагу на зручність використання маркера автозаміни при формуванні лівих частин обмежень, що є функціями Excel.

Після використання «Поиск решения» студент повинен побудувати припустиму область засобами Excel і лінію якогось рівня цільової функції, зіставивши отриману картинку з тією, що була зроблена в зошиті.

При розв'язанні системи рівнянь (для цього пропонується система лінійних рівнянь із трьома невідомими) необхідно акцентувати увагу на те, що систему можна розглядати як окремий випадок задачі математичного програмування, де, наприклад, одне з рівнянь є цільовою функцією, а інші

задають область обмежень. Одержавши розв'язок системи, студент зобов'язаний зробити перевірку рішення, підставивши знайдені числа в кожне з рівнянь, використовуючи засоби Excel, а не калькулятор.

5. Побудова і робота з найпростішими базами даних.

Наступна робота присвячена створенню бази даних і вивченню основних операцій при роботі з базою даних [1].

Студент повинен створити базу даних у відповідності зі своїм варіантом, що містить інформацію про рух товару в деякій організації. Вихідними є наступні поля бази даних: «Місяць»; «Найменування товару»; «Виготовлено»; «Собівартість»; «Продана»; «Ціна».

Вивчення відбувається за наступним планом:

1. Робота з формулами, функціями MS Excel. Абсолютна і відносна адресація.

- додати стовпець «Дата», ввівши інформацію про дату операції, з огляду на інформацію в поле «Місяць»;

- визначити сумарні витрати на виробництво продукції та суму, що надійшла від реалізації продукції (Використовувати функцію СУММПРОИЗВ і додати примітку);

- визначити залишки товару на складі. Попередньо відповідним діапазонам необхідно привласнити імена і використовувати їх у формулі;

- визначити отриманий прибуток від реалізації по кожному найменуванню товару і сумарний прибуток.

- визначити процентне співвідношення прибутку;

- визначити середні собівартість і ціну товарів (використовувати функцію СРЗНАЧ());

- здійснити перерахування прибутку в USD за поточним курсом. При виконанні операції швидкого копіювання використовувати абсолютну адресацію;

- використовуючи умовне форматування, визначити товар, від реалізації якого отриманий максимальний прибуток.

2. Сортування і фільтрація інформації.

- здійснити сортування даних у таблиці за зростанням кількості, що залишилося на складі товару;

- здійснити сортування даних у таблиці за спаданням дати надходження;

- за допомогою фільтрації визначити товари, виготовлені в період з 01.01.05 до «№ варіанта».02.05;

- за допомогою фільтрації визначити товари, собівартість яких більше 1000 грн.

3. Підбор параметра. Функції ЕСЛИ і ВПР.

- ввести новий стовпець «Податок» і, використовуючи функцію ЕСЛИ, виконати обчислення по формулі:

$$\text{Податок} = \begin{cases} \text{прибуток від реалізації} \cdot 1\% \cdot \text{№ варіанта,} & \text{якщо} & \text{прибуток} < 1000 \\ \text{прибуток від реалізації} \cdot 2\% \cdot \text{№ варіанта,} & \text{якщо} & 1000 \leq \text{прибуток} \leq 5000 \\ \text{прибуток від реалізації} \cdot 3\% \cdot \text{№ варіанта,} & \text{якщо} & \text{прибуток} > 5000 \end{cases}$$

- створити нову таблицю Покупці, що містить наступні стовпці: Код, Організація, Телефон і занести довільну інформацію про 5 організацій. Додати в першу таблицю стовпці Код і Організація і довільно заповнити стовпець Код цифрами кодів із таблиці Покупці. Установити зв'язок між першою таблицею і таблицею Покупці, використовуючи функцію ВПР. У першій таблиці сховати стовпець Код і, використовуючи функцію ВПР, визначити назву організації, з якою була виконана відповідна операція;

- у першій таблиці додати стовпець «Чистий прибуток» і виконати обчислення по формулі «Чистий прибуток» = «Прибуток» – «Податок». Визначити кількість проданого товару по категоріях, за умови одержання заданого чистого прибутку. (Використовувати засіб Подбор параметра).

4. Побудова простих і змішаних діаграм. Створення макросів.

- побудувати діаграму “Надходження, продаж і залишки” з відповідним заголовком, причому на одній діаграмі об’єднати два типи діаграм: для Надходження і Продажу використовувати гістограму, а для Залишків – графік;

- створити макрос побудови кругової діаграми “Прибуток від реалізації”. Створити кнопку “Кругова діаграма” і зіставити її із створеним макросом.

5. Створення зведених таблиць. Консолідація й угруповання даних.

- побудувати зведену таблицю, помістивши в рядок – «Найменування товару», у стовпець – «Місяць», а в поле даних «Прибуток». Побудувати діаграму прибутку за місяцями;

- скопіювати на новий лист створену зведену таблицю, перейменувавши аркуші книги у «Склад1» і «Склад2» відповідно. На новому листі створити консолідовану таблицю на основі 2-х зведених таблиць;

- здійснити групування отриманої таблиці по категоріях товарів (комп’ютери, принтери й ін.) і підсумковим даним.

- відформатувати отриману таблицю, використовуючи автоформат і зберегти дані у файлі з паролем захистом.

Приклад виконання лабораторної роботи з баз даних у MS-Excel

Місяць	Найменування товару	Виготовлено	Собівартість	Продано	Ціна	Залишки	Прибуток	Прибуток (%)	Прибуток у USD	Податок	Телефон
січень	Комп’ютер P-466	190	1 833 грн.	182	1 904 грн.	8	12922	20%	2349,45	775,32	2213290
січень	Комп’ютер P-533	10	2 000 грн.	8	2 130 грн.	2	1040	2%	189,09	41,6	2902177
січень	Комп’ютер P-433	90	1 800 грн.	60	1 899 грн.	30	5940	9%	1080	356,4	4412755

січень	Комп'ютер P-466	190	1 827 грн.	22	1 870 грн.	168	946	1%	172	18,92	2902177
лютий	Комп'ютер P-500	130	2 215 грн.	33	2 300 грн.	97	2805	4%	510	112,2	2213290
лютий	Комп'ютер P-500	190	2 200 грн.	45	2 270 грн.	145	3150	5%	572,73	126	2905148
лютий	Принтер EPSON Stylus	30	400 грн.	28	433 грн.	2	924	1%	168	18,48	5532233
лютий	Комп'ютер P-500	70	1 975 грн.	55	2 354 грн.	15	20845	32%	3790	1250,7	2905148
березень	Комп'ютер P-500	150	2 205 грн.	103	2 350 грн.	47	14935	23%	2715,45	896,1	2905148
березень	Принтер EPSON Stylus	25	400 грн.	22	435 грн.	3	770	1%	140	15,4	5532233

Література:

1. Орвис В. Excel для ученых, инженеров и студентов. – К.: Юниор, 1999. – 522 с.
2. Гибхардт Р. Excel 97. Справочник. – М.: Бином, 1998. – 464 с.

ФОРМУВАННЯ ЗМІСТУ ЗАДАЧ ЯК ЗАСОБУ РОЗВИТКУ УЧНІВ ПІД ЧАС ВИВЧЕННЯ ГРАФІЧНИХ РЕДАКТОРІВ

В.О. Потієнко
м. Київ, Ліцей №100 “Поділ”
Chatti@ua.fm, lpodil@mail.ru

Постановка проблеми. Створення навчальних закладів нового типу, бурхлива інформатизація суспільства вимагає нових форм, методів, підходів до викладання предметів, розв’язання задач розвитку мислення, формування навичок оперування формальними поняттями, характерними для взаємодії учнів з комп’ютером. Особливо це стосується викладання в середньому загальноосвітньому закладі курсу інформатики.

Основи роботи з графічними редакторами є обов’язковою складовою діючих програм викладання основ інформатики, пропедевтичного курсу “Вступ до інформатики. 5-6 класи”, курсу “Інформатика. 7-9 класи” [1]. Стандартними вимогами до вмінь учнів з цієї теми є створення простого малюнку. При цьому виникає протиріччя: учні 5-7 класів вміють на високому естетичному рівні створювати зображення, орнаменти, візерунки на папері, виконувати апікаційні роботи, але в них виникають труднощі під час створення подібних зображень на екрані. Проблема полягає в тому, що в учнів цієї вікової категорії ще недостатньо розвинене образне мислення, просторова уява, як наслідок, у них не вистачає умінь ефективно використати комп’ютер для створення зображення на екрані.

Вирішення цієї проблеми можливе через переорієнтацію напрямку навчання учнів роботі в середовищі графічного редактора. Графічний редактор доцільно розглядати не тільки як програмний засіб, що дозволяє навчити учнів 5-7 класів створенню та обробці графічних зображень, сформувати навички користування маніпулятором “миша”, провести пропедевтичну роботу з використання інформаційних технологій, а як засіб з можливостями вирішення загальних завдань навчання та розвитку школярів. Під час роботи в середовищі графічного редактора в учнів з’являються такі уміння, як:

- уміння визначати властивості графічних об’єктів;
- уміння встановлювати залежність між властивостями частин об’єктів або між різними об’єктами;
- уміння встановлювати зв’язки між різними на перший погляд властивостями;
- уміння класифікувати об’єкти за певними ознаками;
- уміння досліджувати закономірності;
- уміння знаходити аналогії.

Завдяки цим вмінням формуються навички учнів аналізувати та розв’язувати задачі взагалі, в результаті розвитку таких умінь підвищується

рівень розумових здібностей учнів, рівень розвитку логічного та образного мислення.

Аналіз останніх досліджень. Нині, коли кожна особистість суспільства оточує неосяжне інформаційне середовище, немає потреби примушувати учнів запам'ятовувати якомога більше фактів. За висловленням К.Д. Ушинського, учень – не посудина, яку слід наповнити, а факел, який треба запалити [2]. Саме тому досягти мети ефективного навчання учнів, розвитку їх розумових здібностей можна через застосування задач відповідної спрямованості.

Питанню застосування учбових задач в процесі навчання, з'ясування їх сутності, значення їх для активізації розумової діяльності приділяють багато уваги психологи, педагоги. Поняття задачі може мати зміст задач навчання. Також це поняття використовується для конкретно спрямованих задач, що здійснюють зв'язок теоретичних знань з їх практичним застосуванням. За визначенням Ю.І. Машбиця [3], учбовими задачами називаються такі задачі, які пропонуються учням для того, щоб їх розв'язання забезпечило досягнення мети навчання.

Поняття “задача” також використовується для опису форм подання учбового матеріалу, спеціальних учбових завдань. Якщо в діяльності учнів перевага надається практичним формам роботи, задачам, навчання буде спрямоване не на засвоєння готової, наданої учителем, інформації, а на розвиток розумових здібностей учнів [4].

В навчальному процесі доцільне використання системи задач з конкретної тематики. Систему задач необхідно розуміти як сукупність задач, на якій реалізоване будь-які педагогічні відношення з визначеними властивостями впливу на зовнішню та внутрішню діяльність учня [5]. Розробка задач системи впливає на розуміння учнями матеріалу теми, в залежності від того, які задачі пропонуються учням, які методи потрібні для їх розв'язання залежить результат педагогічної діяльності як з вивчення конкретної теми, так і з вирішення загальних завдань розвитку учнів під час навчального процесу [6]. Про побудову системи задач, на думку Ф.В. Толкачова [7], можна говорити тільки після визначення критеріїв, за якими б із задач формували комплект, систему з урахуванням призначення те тільки кожної задачі, а всього набору. З'ясування критеріїв побудови системи задач різних типів, різного призначення є важливим напрямком теорії задач. За [7] до понять, за якими з'ясовуються критерії проектування системи вправ, можна віднести поняття “система вправ”, “структура системи вправ”, зміст системи вправ, термін отримання результатів навчання, множина методів, форм та засобів для отримання результату виконання вправ, завдань.

Формування цілей статті (постановка завдання). Метою даної роботи є спроба визначення змісту задач, що сприяють розвитку в учнів просторової уяви, логічного мислення, образного мислення та, як результат, розвивають в учнів уміння для створення малюнків в середовищі графічного ре-

дактора, визначення послідовності комплектування системи із задач в залежності від їх тематики та складності.

Основна частина. Враховуючи вищезазначене, система задач з теми «Графічні редактори» повинна відповідати загальним положенням організації навчального процесу, формування нових знань, умінь та навичок учнів:

- I. Відповідність структури системи задач програмі курсу;
- II. Відповідність змісту задач засвоєнню понять, формуванню вмінь, виробленню навичок, визначених програмою;
- III. Спрямованість змісту задач на розвиток учнів.
- IV. Розподіл задач за рівнями складності.
- V. Здійснення контролю знань з теми.

Відповідно першим двом принципам структура системи задач відповідає структурі викладання теми «Графічний редактор», забезпечує зв'язок між поняттями теми і сприяє засвоєнню учнями системи знань з вищезазначеної теми. Дотримання наступних принципів відбувається за рахунок формування умов кожної окремої задачі.

В середовищі графічного редактора учень користується геометричними об'єктами, але для побудови зображення на екрані, яке б відповідало заданій темі малювання або уявленому образу, йому необхідно визначити відповідність між образом та комбінацією об'єктів, що пропонуються редактором. Звичайно, для жодного учня не виникне проблеми малювання на екрані фарбованого круга, але не кожний учень одразу збагне, що зображення місяця створюється тим же інструментом редактора – фарбованим кругом, використаним двічі, ці круги відрізняються кольором та місцем розташування. Вміння визначати вищезазначену відповідність для малювання необхідним інструментом редактора елементу образу або використання команди редактора для трансформації вже намальованого такого ж елементу з іншими характеристиками формується завдяки розвитку образного мислення та просторової уяви учнів. Саме з цієї причини особлива увага приділяється змісту задач. Таким чином, метою складання задачі є формування в учнів визначених прийомів мислення.

Стосовно тематики вивчення графічного редактора задачі спрямовані на вироблення в учнів таких умінь: орієнтуватись в загальному призначенні фіксованого набору інструментів редактору; обирати необхідні режими використання окремих інструментів, спланувати вірну послідовність дій для використання інструментів та команд редактора; встановити однозначну відповідність між елементами образу, що буде зображений на екрані, та можливостями редактора для зображення цих елементів. Таким чином, до системи задач увійдуть задачі, спрямовані на засвоєння можливостей побудови образу на екрані за допомогою інструментів, запропонованих редактором; та задачі, що сприяють виробленню в учнів умінь доцільно використовувати команди редактора.

Розвиваючий зміст задач має основою положення розвивального на-

вчання за Л.С. Виготським [8]: за задачами системи передбачається поступове формування в учнів вмінь, наприклад, таких як вміння аналізувати, порівнювати, класифікувати. Класифікація об'єктів за визначеними ознаками є результатом аналізу та порівняння цих ознак, тому саме на формування цих вмінь спрямовані початкові задачі системи.

У графічному редакторі на екрані формують абстрактні образи [9], властивостями яких є форма, розмір, колір, просторове співвідношення їх частин. Умови задач представлені в графічному вигляді; початковими даними є зображення на екрані, властивості зображень визначаються та аналізуються для отримання відповіді, яка теж повинна бути представлена малюнком на екрані.

Як і задачі з засвоєння будь-якого шкільного предмету, в системі задачі розташовані за рівнем складності. На кожному рівні передбачається поетапне формування в учнів нових вмінь та навичок, наприклад, таких:

- визначення ознак порівняння;
- за однією властивістю об'єкта;
- за декількома властивостями об'єкта;
- визначення ознак порівняння двох, декількох графічних об'єктів.

Вміння аналізувати, визначати ознаки порівняння необхідні для розв'язання: 1) задач на пошук закономірностей зміни властивостей графічних об'єктів (в умові задач представлені об'єкти, серед яких не вистачає або є зайві, що не відповідають вищезазначеній закономірності); 2) задач на пошук аналогій: прямих – необхідно створити образи об'єктів з властивостями, аналогічних властивостям даних об'єктів та зворотних – необхідно створити образи об'єктів з властивостями, які відсутні в даних об'єктах; 3) задачі з поєднанням пошуку закономірностей та аналогій [10]. В задачах з класифікації об'єктів формуються навички аналізу, диференціації та синтезу – за основою класифікації, наданої в умові аналізуються, порівнюються властивості об'єктів, за спільними властивостями формуються з них групи (групуються готові образи об'єктів або додаються до груп свої зображення, свої групи зображень).

Цілеспрямовані формування вищезазначених прийомів розумової діяльності, розвиток виділяти ознаки об'єктів, визначати їх властивості, будову дозволяє перейти від випадково вірного до свідомого вибору необхідного інструменту редактора, до вмілого та доцільного вибору команд редактора.

Важливою ланкою навчального процесу в школі є контроль знань учнів. Саме він дозволяє вчителю проаналізувати ефективність застосованої методики навчання, доцільність використання дидактичного матеріалу, тому завданнями контрольних робіт з теми “Графічний редактор” можуть бути як завдання зі створення малюнку, так і завдання визначити, які інструменти редактора, його команди необхідно використати для малювання запропонованого образу, завдання описати послідовність дій чи за визначений час створити образ на екрані за зразком із визначеної теми.

Висновки. Для засвоєння можливостей роботи в середовищі графічного редактора в учнів середнього шкільного віку повинно бути розвинене образне мислення, просторова уява. Саме тому доцільне формування системи розвиваючих задач, умова яких оснований на властивостях образів, що можна побудувати на екрані, а розв'язок яких є результатом роботи учнів в середовищі редактора. Формування розвиваючих задач для вивчення теми "Графічні редактори" сприяє як активізації розумової діяльності учнів, так і підвищенню їх інтересу до засвоєння даної теми. Такі задачі також мають значення для отримання пропедевтичних знань з геометрії (формування вмінь розкласти складну фігуру на простіші, які легше аналізувати, підвищує ефективність розв'язку задач), зі створення анімаційних зображень, в яких просторові властивості об'єктів на екрані змінюються з часом.

Література:

1. Інформатика. Програми для загальноосвітніх навчальних закладів. – Запоріжжя: Прем'єр, 2003. – 304 с.
2. Ушинський К.Д. Избр. пед. соч.: В 2 т. Т. 1. – М., 1953. – С. 321.
3. Машбиц Е.И. Психологические основы управления учебной деятельностью. – К.: Вища шк., 1987. – 224 с.
4. Вишневецький О. Побудуємо систему задач, доступних кожній дитині // Освіта. – 2001. – 2-9 травня. – С. 9.
5. Бурдин А.О. О классификации задач // Совершенствование содержания и методов обучения естественно-математическим дисциплинам в средней школе. – М., 1981. – С. 3-7.
6. Балл Г.А. Теория учебных задач: психологический аспект. – М.: Педагогика, 1990. – 184 с.
7. Толкачев Ф.В. Система упражнений по императивному программированию в фундаментальной подготовке будущих учителей информатики: Автореф. дис. ... канд. пед. наук: 13.00.02 / Рос. гос. пед. ун-т. –С.-П., 2000. – 20 с.
8. Выготский Л.С. Развитие высших психических функций. – М.: Изд-во Ак. пед. наук, 1960. – 500 с.
9. Дорошенко Ю.О., Очеретний В.О. Геометричне моделювання у побудові лінійних зображень засобами комп'ютерної графіки // Науковий вісник Ізмаїльського державного гуманітарного університету. – Ізмаїл, 2004. – Вип. 16. – С. 118–126.
10. Потієнко В.О. Розвиваючі задачі як засіб підвищення ефективності вивчення графічного редактора // Науковий вісник Ізмаїльського державного гуманітарного університету. – Ізмаїл, 2004. – Вип. 16. – С. 189–192.

ИЗУЧЕНИЕ ТЕХНОЛОГИИ СОЗДАНИЯ ЭКСПЕРТНЫХ СИСТЕМ С ПОМОЩЬЮ ОБОЛОЧКИ *XPERTRULE KNOWLEDGE BUILDER*

О.А. Синявская^а, Б.А. Железко^б

Беларусь, г. Минск, Белорусский государственный экономический университет

^а o.siniavskaya@sam-solutions.net

^б boriszh@yandex.ru

Введение

С развитием информационных технологий накопление знаний и их последующее применение в различных ситуациях реализованы в системах искусственного интеллекта. К данному классу относятся экспертные системы (ЭС). ЭС определяются как программы ЭВМ, моделирующие действия эксперта-человека при решении задач в узкой предметной области, на основе накопленных знаний, составляющих базу знаний. Назначение экономических экспертных систем состоит в обработке больших массивов информации, выборе на основе такой обработки оптимальных решений и их экономическом обосновании.

Создавать ЭС можно двумя способами: применяя алгоритмические языки программирования (C, LISP, Pascal, PROLOG и др.) или оболочки ЭС. Оболочка экспертных систем представляет собой готовую программную среду, которая может быть приспособлена к решению определенной проблемы путем создания соответствующей базы знаний. То есть это готовая экспертная система без базы знаний. Достоинство оболочек в том, что они вообще не требуют работы программистов для создания готовой экспертной системы. Необходим только специалист в данной предметной области для заполнения базы знаний. Применение оболочек позволяет создавать ЭС значительно быстрее и легче в сравнении с программированием. Таким образом, оболочки ЭС могут эффективно применяться в учебном процессе для ознакомления студентов, чьи специальности не связаны с программированием, с принципами создания и эксплуатации экспертных систем.

Цель работы

Целью работы с оболочкой ЭС «XpertRule Knowledge Builder» является получение практических навыков построения экспертных систем с помощью оболочек, наполнения базы знаний фактами и правилами, освоение принципов работы с экспертной системой путём решения задачи из профессиональной сферы.

Условие задачи

Для создания простейшей экспертной системы используется вполне работоспособная демо-версия оболочки ЭС «XpertRule Knowledge Builder».

Рассмотрим пример построения с помощью данной оболочки экспертной системы «EXPENSES». ЭС «EXPENSES» помогает решить, принять

или отклонять заявление на возмещение командировочных затрат на гостиницу от сотрудников компании. Решение принимается на основе 3 факторов:

- должность сотрудника в компании – переменная «Grade» в ЭС;
- отдел, в котором работает сотрудник – переменная «Department» в ЭС;
- уровень гостиницы, в которой останавливался сотрудник – переменная «Hotel» в ЭС.

Созданная система на выходе имеет два возможных результата обработки заявления:

- «Pass» – заявление принимается к оплате;
- «Reject» – заявление отклоняется.

В экспертной системе будут использоваться следующие правила:

1. Директорам могут оплачиваться любые гостиницы.
2. Менеджерам старшего звена могут оплачиваться гостиницы уровня С, а также, если они работают в отделе «Account», то и гостиницы уровня В.
3. Менеджерам младшего звена могут оплачиваться только гостиницы уровня С.

Последовательность выполнения лабораторной работы по созданию ЭС с помощью оболочки «XpertRule Knowledge Builder»

1. Запустите программу через меню «Пуск» → «XpertRule Knowledge Builder» → «Developer».

2. В меню «File» выберите команду «New Project». Программа активирует мастера создания новых проектов, следуя указаниям которого, создайте новый проект.

2.1. В появившемся окне для перехода к созданию проекта нажмите кнопку «Next», данная кнопка будет использоваться в мастере проектов для перехода к следующим этапам создания проекта.

2.2. Выберите тип базы данных «New MS Access project database».

2.3. Выберите место сохранения – D:\Мои документы\<Ваша личная папка>\expenses. В названии файла базы данных (в приведенном примере expenses) следует использовать только латинские буквы и символ _ .

2.4. В окне «User name» и «Password» ничего не изменяйте.

2.5. Завершите создание нового проекта нажатием кнопки «Finish».

3. Щелкните правой кнопкой мыши на «Project» (в левом верхнем углу) и выберите «New» → «Knowledge Module». Переименуйте модуль в «Expenses». Разверните модуль, нажав на значок +. Работа с модулями организована по аналогии с проводником.

4. Нажмите правую кнопку мыши на категории «Attributes» и выберите «New» → «List_Attribute». Данное действие создает новую логическую переменную, которая может принимать одно из заданного экспертом списка значений. В появившемся окне введите имя «Department». Аналогично создаются переменные «Expenses», «Grade» и «Hotel». В результате окно

структура модуля должна принять вид, представленный на рис. 1.

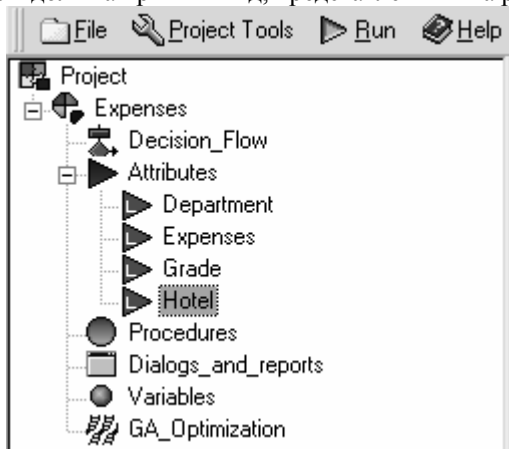


Рис. 1. Структура модуля *Expenses*

4. Задайте возможные значения переменным. После двойного нажатия левой клавиши мыши на переменной «Department» справа открывается окно редактора. Выберите закладку «Instance Properties» путем нажатия на кнопку третью сверху кнопку в левой панели кнопок (в виде таблицы).

5. С помощью значка «+», расположенного в левом углу таблицы, добавьте значения переменной «Account» и «Sales». Таблица примет вид, представленный на рис. 2. Повторите процедуру для остальных переменных, введя следующие значения:

- «Grade» – «Director», «Senior_Manager», «Junior_Manager»;
- «Expenses» – «Pass», «Reject»;
- «Hotel» – «A», «B», «C».


6. Сохраните изменения кнопкой . Данная кнопка сохраняет новые данные только во временном файле. Для того, чтобы внести изменения в файл базы данных, необходимо выбрать в главном меню «File» пункт «Commit changes to Database».



Рис. 2. Задание значений переменной *Department*

7. В редакторе для переменной «Hotel», с помощью значка «+» расположенного в правом верхнем углу таблицы добавьте еще одну колонку. Для этого в выпадающем меню выберите «String Property» (свойство строки). Двойным щелчком мыши на заголовке колонки вызовите окно свойств и переименуйте колонку в «Runtime_Text». В этом же окне зайдите на закладку «Runtime Display» и выберите пункт «Default Order». Данная операция активизирует отражение новых данных в процессе диалога созданной экспертной системы с пользователем. Заполните строки новой колонки так, как показано на рис. 3.

	Value	Runtime_Text	+
1	A	Grade A Hotel	
2	B	Grade B Hotel	
3	C	Grade C Hotel	

Рис. 3. Описание значений переменной *Hotel*

8. Войдите в окно редактора для переменной «Expenses» и нажмите кнопку верхнюю кнопку в панели кнопок слева. Данная кнопка вызывает мастера создания объектов (рис. 4).

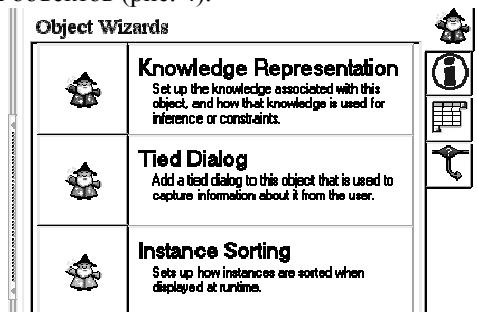




Рис. 4. Мастера создания объектов

Выберите мастера «Knowledge Representation» (представления знаний). Следуя указаниям мастера, поставьте отметку в пункте напротив графы «Tree». Данное действие включает в редакторе кнопку , позволяющую создавать и редактировать дерево решений.

8.1. Правило №1 (директорам могут оплачиваться любые гостиницы). Нажмите на кнопку  в редакторе для переменной «Expenses». Удерживая левую кнопку мыши на переменной «Grade» в окне модуля «Expenses», перетащите её на объект «Empty» в окне редактора. Двойным щелчком мыши на объекте «All Values» (все значения) выберите «Multi-split». В окошке значения напротив объекта «Director» двойным щелчком выберите «Pass» (рис. 5). Таким образом, отображается первое правило.

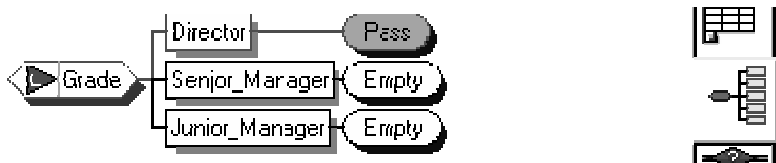


Рис. 5. Реализация правила №1 в ЭС «Expenses»

8.2. Правило №2 (менеджерам старшего звена могут оплачиваться гостиницы уровня C, а также, если они работают в отделе «Account», то и гостиницы уровня B). Удерживая левую кнопку мыши на объекте «Hotel», перетащите его на блок «Empty» напротив объекта «Senior_Manager». По аналогии с предыдущим правилом, приведите дерево решений к виду, представленному на рис. 6.

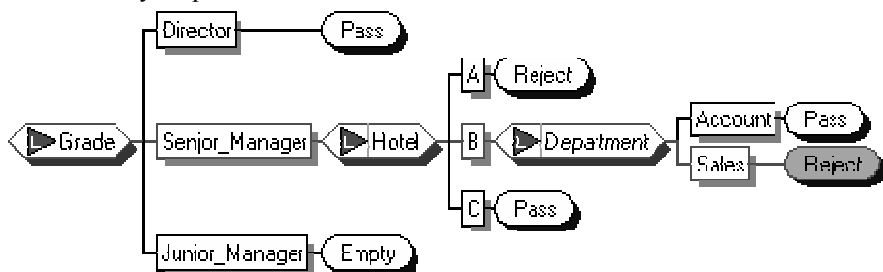


Рис. 6. Реализация правила №2 в ЭС «Expenses»

8.3. Правило №3 (Менеджерам младшего звена могут оплачиваться только гостиницы уровня C). Перетащив объект «Hotel» на блок «Empty» напротив объекта «Junior_Manager», правой кнопки мыши вызовите контекстное меню и выберите пункт «Edit List split». В появившемся окне «Edit List split» перетащите значение A к значению B, они объединятся в одну ветвь. Удалите пустую ветвь (для этого выделите её и нажмите «Delete» на клавиатуре). Затем нажмите «ОК». Закончите ввод правила по аналогии с предыдущими правилами. Окончательно сформированное дерево решений представлено на рис. 7.

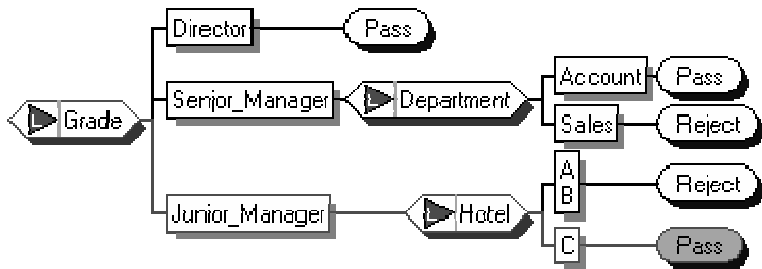


Рис. 7. Дерево решений в ЭС «Expenses»

9. В окне модуля перейдите на название модуля «Expences». Удалите весь текст в окне «Main Agenda» справа и наберите следующее:

@Do Expenses

@Debug Expenses

10. Сохраните все изменения в файле базы данных и запустите приложение, выбрав пункт «Run» в главном меню. Протестируйте систему, задавая различные исходные данные.

Примеры задачи из предметной области специалистов

После рассмотрения примера студентам предлагается выполнить аналогичное задание, используя знания из своей предметной области (экономика, финансы, маркетинг и т.д.). Например, специалистам в области маркетинга задание может быть сформулировано следующим образом.

Построить экспертную систему, позволяющую определить маркетинговую стратегию фирмы на основе анализа данных рынка и финансового состояния организации. В список переменных включить цену и качество продукции, объём продаж, организацию торговли, степень конкуренции и т.д. Значения данных переменных задать в лингвистической форме (высокая, низкая и т.д.). Результатом работы с системой должна быть рекомендация о дальнейших маркетинговых решениях фирмы (улучшить качество, улучшить производство или начать выпуск новой продукции).

В области финансов выделяется 4 основных класса ЭС, которые могут быть предложены как темы для выполнения задания финансистам [1]:

- ЭС для финансового анализа;
- ЭС для анализа причин успешного (неуспешного) развития бизнеса;
- ЭС для анализа рынка;
- ЭС для получения знаний в области финансов.

Выводы

Выполнение рассмотренного задания позволяет студентам понять принципы построения экспертной системы, получить практические навыки работы с ЭС, а также систематизировать свои знания как эксперта. Эксперты в сфере финансов владеют знаниями, полученными из практики, которые не могут быть найдены в литературе или получены каким-либо другим способом, но которые бесценны для обеспечения успешной деятельности фирмы или финансовой организации. Повсеместное применение информационных технологий, в том числе сетевых, позволяет систематизировать и накапливать данные знания в экспертных системах, повторно использовать их для эффективного решения экономических задач и обновлять знания. Поэтому изучение технологии создания экспертных систем для экономистов является актуальным и необходимым.

Литература:

1. L. Nedovic, V. Devedzic. Expert systems in finance – a cross-section of the field // Expert Systems with Applications. – 2002. – №23. – P.49–66.

МЕТОДИЧНІ УМОВИ ПІДГОТОВКИ СТУДЕНТІВ ЕКОНОМІЧНИХ СПЕЦІАЛЬНОСТЕЙ ДО РОЗВ'ЯЗАННЯ ТВОРЧИХ ФАХОВИХ ЗАДАЧ НА ЗАНЯТТЯХ З ІНФОРМАТИКИ

О.Г. Смілянець

м. Вінниця, Вінницький торговельно-економічний інститут
Київського національного торговельно-економічного університету
Smilynez2002@ukr.net

Сучасне суспільство вимагає від вищих навчальних закладів підготовки кваліфікованих, заповзятливих, ініціативних фахівців, що максимально використовує свої творчі задатки і здатні діяти в умовах нестабільності і невизначеності, що вміють оперативно і якісно працювати з економічною і правовою інформацією, залучаючи для цього сучасні засоби і методи.

Саме інформатика, як прикладна наука, при підготовці студентів в інституті має необмежені можливості. Курси “Інформатика та комп’ютерна техніка” та “Комп’ютерний практикум”, що викладаються студентам економічних спеціальностей у Вінницькому торговельно-економічному інституті, в силу своєї специфіки, складності та прикладної спрямованості, мають значні можливості для розвитку особистості студента.

Спираючись на висновки В.І. Андрєєва про те, що для організації навчального процесу, який сприяє розвитку потенціальних творчих можливостей учнів, можна ефективно використовувати навчально-творчі задачі, для підготовки студентів фінансово-економічних спеціальностей ми пропонуємо у навчальний процес ввести фахові творчі задачі. В.І. Андрєєв визначає навчально-творчу задачу як таку форму організації змісту навчального матеріалу, за допомогою якої педагог створює учням творчу ситуацію, прямо чи опосередковано задає мету, умови та вимоги до навчальної творчої діяльності, в процесі якої учні активно оволодівають знаннями та навичками, розвивають свої творчі здібності [1, с. 37].

При розв’язанні творчих фахових задач діяльність студента концентрується на:

- винахідливості – здатності мислити, генерувати нові корисні ідеї або принципи, які необхідні для досягнення поставленої мети;
- вмінні проводити творчий аналіз – здатності аналізувати окремі елементи, систему або процес з використанням ефектів і навчальних принципів з метою оперативного отримання правильного розв’язку;
- економічних знаннях – досконале знання і його засвоєння з загально-економічних дисциплін та ін.;
- широкій спеціалізації – здатності компетентно і впевнено володіти основними проблемами або ідеями навчальних дисциплін, які знаходяться за межами даної спеціалізації;
- вмінні приймати рішення в умовах невизначеності, але з повним і все-

- бічним врахуванням усіх існуючих факторів;
- вмінні передавати інформацію про одержані результати – здатності виражати свої думки чітко і впевнено (усно, письмово, графічно).

Аналіз психолого-педагогічної літератури показує, що системне вивчення евристичних прийомів розпочато відносно недавно, але окремі їх види, особливо ті, що стосуються розв'язування математичних задач, були досліджені раніше. Так, наприклад, проблемним евристичним прийомам розв'язування задач присвячена книга Д. Пойа “Як розв'язувати задачі?”. В ній всі прийоми розв'язування математичних задач зведені у таблицю, і автор рекомендує, насамперед, добре зрозуміти умову задачі, зробити креслення, записати умову, розбити розв'язок задачі на окремі етапи. Д. Пойа пропонує загальний метод, загальні вказівки щодо розв'язування задачі, а не описує конкретні прийоми знаходження відповіді [1, с. 177].

Методичні умови проведення занять визначаються специфікою суб'єкта навчання в освіті. Як відомо, студенти мають наступні специфічні відмінні характеристики:

- 1) усвідомлюють себе самостійними, самокерованими особистостями;
- 2) мають запас життєвого і професійного досвіду, що стає усе більш важливим джерелом навчання;
- 3) в основі готовності до навчання лежить прагнення за допомогою навчання вирішити свої життєво важливі проблеми і досягти конкретних цілей;
- 4) прагнення до невідкладної реалізації придбаних компетенцій та якостей.

При підготовки студентів до рішення творчих фахових задач на лабораторних заняттях з інформатики увагу викладача повинна привертати міра розвитку у студентів економічної та наукової інтуїції, комп'ютерної грамотності, механізмів конструювання правдоподібних умовиводів, здогадок, здібностей до створення комп'ютерних образів економічних об'єктів, уяви та фантазії.

В дидактиці стверджується, що для найкращого задіяння і розвитку цих механізмів необхідно в заняття вносити якість диво, якусь таємницю, інтригу; там, де це можливо і доцільно “розставляти пастки” для студентів, щоб продуктивно розвивати їхнє мислення, навички інтелектуального пошуку; залучення до творчої дослідницької діяльності студентів, що за тих чи інших обставин погано встигають з інформатики, а також представників так званої категорії “середніх”.

Пробуджуючи у студентів активні дослідницькі інтереси і доцільно організовуючи їхню творчу діяльність, можна дидактично грамотно розвивати їхнє творче мислення. Але для того, щоб цей розвиток відбувався продуктивно, важливим є формування для кожного студента індивідуальної зони-ситуації творчого розвитку, яка близько співвідноситься із зоною найближчого розвитку особистості.

В нашому дослідженні при підготовці до розв'язання творчих фахових задач майбутніх спеціалістів фінансово-економічного профілю використовувались узагальнені орієнтири, на які студенти мусять спиратися при розв'язку будь-якої розрахункової задачі [3, с.129]:

- 1) проаналізувати запитання задачі і з'ясувати, що дано і, що потрібно знайти;
- 2) визначити, які дані необхідно для відповіді на запитання задачі;
- 3) з'ясувати, чи всі необхідні дані наведені в умові задачі; якщо ні, то необхідно визначити спосіб знаходження відповідних величин;
- 4) спланувати послідовність операцій, спрямованих на знаходження відповіді (алгоритму розв'язання);
- 5) реалізувати запланований шлях розв'язку;
- 6) перевірити розв'язок задачі.

Застосування аналогічних орієнтирів на заняттях з інформатики дає студентам можливість свідомо оволодівати загальним методом розв'язування навчально-творчих фахових задач. Студентам з високим рівнем творчого розвитку засвоєння прийомів узагальненої розумової діяльності забезпечує ефективне оволодіння умінням застосовувати на практиці знання, які необхідні при розв'язуванні задачі. Такі студенти, як правило, не потребують допомоги з боку викладача. Саме тому їм слід пропонувати творчі задачі і мати на увазі, що їх творча діяльність може мати як об'єктивну, так і суб'єктивну новизну і оригінальність. Студентам з низьким і середнім рівнем творчого розвитку необхідна диференційована допомога. Тому викладач радить таким студентам звернутися до аналогічної задачі, рекомендує використати певний довідковий матеріал, розділ підручника. Викладач може ознайомити “слабкого” студента з алгоритмом розв'язування даної задачі. Таким чином, задача творча перетворюється у навчальну, рівень її креативності знижується. Така організація творчої навчальної діяльності студентів дозволяє кожному працювати в оптимальному темпі, не відчувати своєї “інтелектуальної неповноцінності”, поступово навчатися узагальненим прийомам розумової діяльності, методам розв'язування широкого кола фахових задач. При цьому викладач має можливість планомірно розвивати в студентів творчий інтерес, допитливість, потяг до пошуку нової інформації та фактів, сміливість, самостійність, ініціативність, впевненість у своїх силах та здібностях, цілеспрямованість, наполегливість, працелюбність, вміння доводити почату справу до кінця, тобто весь спектр особистісних якостей важливих для успішної творчої діяльності людини.

Так, для студентів зі спеціальності бухгалтерський облік та аудит запропоновується завдання для розрахунку заробітної оплати і податків на зарплату. Для розв'язання цієї задачі студентам необхідно вміти користуватись багатьма інструментами і застосовувати їх для рішення практичної задачі з обліку. При розв'язанні цієї задачі в середовищі електронних таблиць

потрібно вміти застосовувати логічні функції, вбудовані функції, форми для зміни умов та алгоритмів обчислень, використання абсолютних та змішаних посилань у формулах.

Корисною в плані надання творчого характеру інтелектуальній праці студентів є практика розв'язування задач економічного змісту, які вони розв'язували в інших дисциплінах. При цьому в пошуковій діяльності студентам допомагатимуть програмно реалізовані можливості відображення на екрані комп'ютера прикладів подібних задач.

Форма проведення лабораторних занять з інформатики, на яких розв'язуються творчі фахові задачі, може бути різноманітною: від практикумів, під час яких всі студенти отримують диференційовано дібрані задачі та індивідуально розв'язують їх, зберігаючи файли з реалізованими побудовами та записуючи в зошиті хід розв'язання задачі, до ігор типу "брейнстормінг" з жорсткою регламентацією часу, з дотриманням наперед визначених правил, ролей, але в той же час з обов'язково реалізованими елементами невимушеності обстановки, припустимості експериментування, вільного висловлення ідей та пропозицій.

Проте завжди на подібних заняттях творчій експериментальній діяльності студентів повинно передувати пропедевтичне спрямовуюче розумову діяльність пояснення викладача, а в процесі проведення – ненав'язливий контроль та стимулююча підтримка. Наприкінці заняття підводяться підсумки, відмічаються лідери, оцінюється діяльність всього колективу і кожного студенту окремо.

На відміну від традиційного методу розв'язування при цьому у студентів виробляється корисна тенденція реалістично підходити до уявлення об'єктів, про які йдеться в умові. Це значною мірою сприяє усвідомленню задачного матеріалу, виробленню у студентів навичок застосування комп'ютерної техніки до розв'язування задач, зокрема з фаховим фінансово-економічним змістом.

В процесі навчання велика роль належить викладачу, який здатний спрямувати студентів на шлях пошуків, викликати у них пристрасть до пошуку. Але без особистого захоплення пізнанням, без наявності педагогічного таланту і такту цього домогтися складно. Студент повинен мати зразок, приклад для наслідування, в ньому потрібно посіяти іскру, з якої з часом спалахне полум'я пошуку, незадоволеності досягнутим. Викладач допомагає молодим людям увійти в атмосферу творчості, в круг ідей, робота над якими розкриває широкі можливості для самостійного пошуку і для нових наукових знахідок.

В педагогічному спілкуванні з студентами викладачі повинні дотримуватись певних правил, наприклад, не погоджуватись з відповіддю, якщо вона просто стверджується і приймається на віру, вимагати доведень; ніколи не розв'язувати суперечку студентів найлегшим способом, тобто просто повідомляючи їм правильну відповідь або правильний спосіб розв'язування

задач; уважно вислуховувати їх, щоб не пропустити моменту розкриття для них чогось нового; постійно пам'ятати, що навчання повинно спиратись на інтереси, мотиви і бажання студента; розклад занять та дзвінки не повинні бути визначальним фактором навчального процесу; поважати власні “шалені” ідеї та прищеплювати іншим смак до нестандартного мислення; ніколи не говорити учневі, що немає часу обговорювати його ідею; не скупитися на схвальне слово, доброзичливу посмішку, дружню підтримку; в процесі навчання не може бути постійної методики та раз і назавжди встановленої програми [2, с. 64].

Розвиток творчих здібностей вимагає тривалого впливу і повинен бути предметом уваги педагогічного колективу буквально з перших днів навчання. Вихованню прагнення до творчості слід приділяти пильну увагу на всіх етапах навчання. І кожний предмет приховує в надрах своїх теорій великий потенціал засобів впливу на розвиток творчості студента.

Дійсно, пошук розв'язків нестандартних творчих задач, нестандартних шляхів розв'язування фахових фінансово-економічних задач, роздуми про парадокси, пошук помилок в міркуваннях, аналіз змісту завдання і сутності їх рішення, бесіди про творчі лабораторії відомих вчених – все це утворює важливі складові на шляху творчого розвитку майбутніх фахівців економічного профілю.

Таким чином, основними умовами ефективності запровадження творчих фахових задач у навчальний процес з використанням інформаційних технологій є:

- обсяг, складність і кількість інформації, яка сприймається студентами;
- навчальні можливості студентів, тобто психологічна готовність студентів розв'язувати задачі засобами ІТ;
- професійна підготовка викладачів запроваджувати в навчальний процес творчі фахові задачі та вирішувати їх засобами ІТ.

Література:

1. Андреев В.И. Диалектика воспитания и самовоспитания творческой личности. – Казань: Изд-во Казанского университета, 1988. – 228 с.
2. Лук А.Н. Мышление и творчество. – М.: Политиздат, 1976. – 144 с.
3. Освітні технології: Навч.-метод. посіб. /О.М. Пехота, А.З. Кіктенко, О.М. Любарська та ін.; За заг. ред. О.М. Пехоти. – К.: А.С.К., 2001. – 256 с.

ИСПОЛЬЗОВАНИЕ ПРОГРАММЫ HYPERCAM ДЛЯ СОЗДАНИЯ ОБУЧАЮЩИХ ВИДЕО-РОЛИКОВ

Ю.А. Супрунова

г. Кривой Рог, Криворожский филиал Института экономики и новых технологий
j@alba.dp.ua

Современные информационные технологии представляют студенту очень широкий спектр возможностей. Целью данной статьи является показать возможность использования мультимедийных средств в процессе преподавания различных дисциплин, связанных с компьютерной техникой.

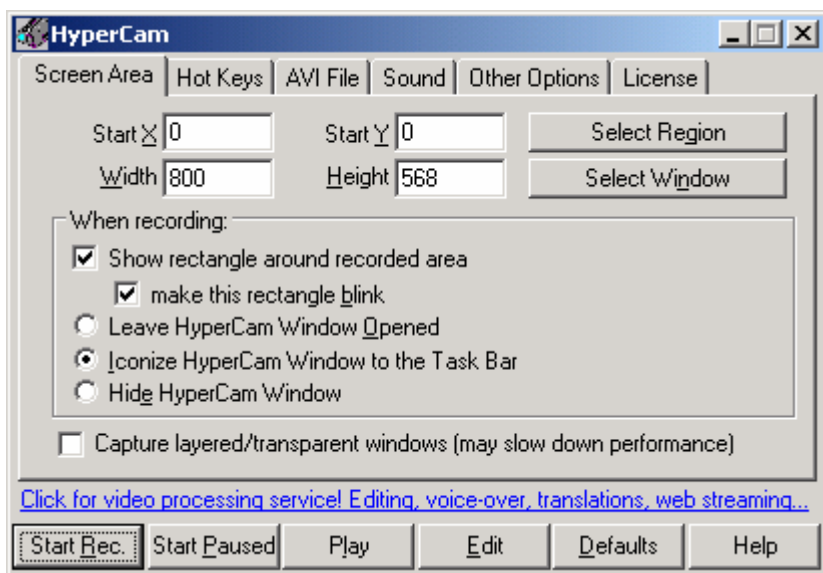
Наличие мультимедийных классов позволяет в значительной мере усовершенствовать методику преподавания основ работы с каким-либо прикладным программным обеспечением. Визуальное представление последовательности действий для выполнения необходимых операций в значительной мере помогает студенту разобраться с приемами работы в среде какого-либо программного продукта.

Мы считаем, что для этих целей целесообразнее всего использовать обучающие видео-ролики, которые дают студентам наглядное представление о возможностях и технологиях работы с программой. Выбор существующих мультимедийных продуктов достаточно широк, но, как правило, большинство из них не соответствует требованиям учебной программы или особенностям методики преподавания предмета. Выходом из этого положения может быть создание собственных видео-роликов, что достаточно легко реализовать при помощи программы HyperCam.

HyperCam – это чрезвычайно простая в использовании программа, которая представляет собой виртуальную видеокамеру с возможностью последующей записи последовательности действий в формате avi и наложением звуковой дорожки. Программа позволяет настраивать множество параметров, таких, как размер конечного файла, скорость передачи кадров и качество сжатия для захвата видео. HyperCam поддерживает текстовые аннотации, звук, экранные примечания и является незаменимым помощником при создании обучающих фильмов и автоматических демонстрационных версий программ.

Незарегистрированная версия программы не имеет никаких функциональных ограничений. Единственное ее отличие – отображение значка Unregistered HyperCam в углу видеоролика.

Ниже представлено рабочее окно программы HyperCam. Кнопка Select Region позволяет выбрать область видеозахвата, кнопка Select Window – окно той программы, в которой будут производиться записываемые операции.



В качестве иллюстрации возможностей Hyper Cam предлагается серия обучающих роликов о приемах работы в среде САПР AutoCAD 2002, которые используются в ходе изучения дисциплины «Инженерная и компьютерная графика». В данных роликах наглядно представлен процесс выполнения типовых контрольных заданий – чертежей и трехмерных моделей с использованием элементов визуализации. Изучение основных инструментов создания и редактирования объектов происходит на примере решения конкретных задач, что значительно облегчает восприятие студентами материала. При этом процесс выполнения сопровождается звуковыми комментариями, поясняющими студенту смысл и назначение отображаемых на экране действий. В видео-роликах также представлен процесс первоначальной настройки системы, возможности создания слоев и настройки их параметров, использование шаблонов, инструмента объектной привязки, различных режимов вычерчивания объектов – того необходимого минимума, без которого невозможна грамотная и продуктивная работа в среде САПР AutoCAD 2005.

При аудиторном использовании видео-роликов в любой момент воспроизведение можно остановить и ответить на появившиеся вопросы, предоставить студентам возможность записать основные тезисы лекции. Видео-ролики можно также использовать в качестве дополнения к методическим пособиям. При самостоятельном изучении материала студент, просматривая видео-ролик, может пошагово выполнять описанные действия на своем компьютере. Изучив последовательность выполнения типового задания, студент может приступить к выполнению заданий согласно своему ва-

рианту, закрепив на практике приобретенные навыки работы.

При помощи данной программы можно с минимальными затратами времени создавать обучающие фильмы по технологии использования любого программного обеспечения – графических и текстовых редакторов, электронных таблиц и других прикладных программных пакетов.

Обобщая опыт использования описанного программного продукта, хотелось бы отметить следующее. Использование обучающих видеороликов как дополнения к методическим пособиям позволяет, во-первых, в значительной мере улучшить восприятия материала студентами, особенно это касается студентов заочной формы обучения, количество аудиторных часов которых весьма невелико, во-вторых, ограничить объем самих методических пособий за счет возможности более сжатого изложения информации.

ОСОБЛИВОСТІ МЕТОДИКИ ВИКЛАДАННЯ КУРСУ “ЧИСЕЛЬНІ МЕТОДИ” ДЛЯ СТУДЕНТІВ ЕНЕРГЕТИЧНИХ СПЕЦІАЛЬНОСТЕЙ

О.В. Тумашова, О.М. Коссак
м. Львів, Національний університет “Львівська політехніка”
kora@LITech.net

Технологією побудови та аналізу за допомогою сучасних персональних комп’ютерів (ПК) математичних моделей є обчислювальний експеримент. В основі його лежить розробка чисельних методів і алгоритмів розв’язування даної математичної моделі, програмування на ПК обчислювального алгоритму, проведення обчислень та аналіз результатів. При розв’язуванні задачі на ПК ми завжди одержуємо не точний, а деякий наближений розв’язок відповідної задачі. Одній і тій же математичній задачі можна поставити у відповідність множини різних дискретних моделей.

Бурхливий розвиток ПК дав можливість розв’язувати складні інженерно-технічні задачі. Але без практичних навичок застосування наближеного та чисельного аналізів ці задачі розв’язати практично неможливо. Цим пояснюється надзвичайно велика увага до чисельних методів.

Методику викладання курсу “Чисельні методи” розглянемо на прикладі методу сіток розв’язування крайових задач для диференціальних рівнянь еліптичного типу, які застосовуються для розв’язування задач в енергетиці.

До рівнянь еліптичного типу зводяться задачі розрахунку напружень, які виникають при пружному крученні циліндричного стержня, розрахунку стаціонарних потоків тепла в двомірному тілі, тощо.

Розглянемо задачу Діріхле для рівняння Лапласа:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad (1)$$

в деякій області $R = \begin{cases} 0 < x < A \\ 0 < y < B \end{cases}$. На границі цієї області Γ задано граничну

умову (умову Діріхле)

$$u(x, y) = f(x, y), \quad x, y \in \Gamma.$$

Для простоти викладу будемо вважати область прямокутною, зорієнтованою по осях координат. Поділимо сторону A на n інтервалів, а сторону B на m інтервалів. Тоді $h = \frac{A}{n}$, $k = \frac{B}{m}$, тобто будуємо сітку з вузлами: $x_i = ih$, $y_j = jk$. В результаті маємо $(n+1)(m+1)$ вузлів сітки. Нехай початок координат збігається з точкою $(0, 0)$. Позначимо $u(ih, jk) = u_{i,j}$, $f(ih, jk) = f_{i,j}$. Тоді граничну умову можна подати у вигляді:

$$u_{i,0} = f_{i,0}, \quad i=0, 1, 2, \dots, n-1;$$

$$u_{i,m} = f_{i,m}, \quad i=0, 1, 2, \dots, n-1;$$

$$u_{0,j} = f_{0,j}, \quad j=0, 1, 2, \dots, m-1;$$

$$u_{n,j}=f_{n,j}, j=0, 1, 2, \dots, m-1;$$

Апроксимуємо похідні в кожному внутрішньому вузлі сітки центральними різницями другого порядку. Рівняння Лапласа в скінчених різницях запишеться

$$\frac{1}{h^2} (u_{i+1,j} - 2u_{i,j} + u_{i-1,j}) + \frac{1}{k^2} (u_{i,j+1} - 2u_{i,j} + u_{i,j-1}) = 0.$$

Якщо ввести $\lambda = \frac{k}{h}$, то можна записати

$$\lambda^2 u_{i+1,j} + \lambda^2 u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 2(1 + \lambda^2) u_{i,j} = 0, \\ i=1, 2, \dots, n-1; j=1, 2, \dots, m-1. \quad (2)$$

Отримаємо систему $(n-1)(m-1)$ лінійних алгебраїчних рівнянь відносно $(n+1)(m+1)$ невідомих. За допомогою граничних умов виключаємо $2(m+n)$ невідомих, і залишається $(n-1)(m-1)$ невідомих.

Систему $(n-1)(m-1)$ лінійних алгебраїчних рівнянь можна розв'язати ітераційним методом.

Розглянемо випадок, коли $h=k, \lambda=1$. Тоді з (2) отримаємо:

$$u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} = 4u_{i,j}; \\ i=1, 2, \dots, n-1; j=1, 2, \dots, m-1.$$

Для цієї системи рівнянь виконуються умови збіжності методу Зейделя, а саме: кожен з діагональних коефіцієнтів дорівнює 4, а сума решта коефіцієнтів в кожному рівнянні є менша, ніж 4.

Метод Зейделя дасть наступну послідовність ітерацій:

$$u_{i,j}^{(s+1)} = \frac{1}{4} (u_{i+1,j}^{(s)} + u_{i-1,j}^{(s+1)} + u_{i,j+1}^{(s)} + u_{i,j-1}^{(s+1)}), s = 0, 1, \dots$$

де s – номер ітерації. За початкове наближення вибираємо значення функції $u(x, y)$ на границі Γ області R . Для цієї різницевої схеми виконуються умови стійкості (малі зміни початкових даних приводять до малих змін розв'язку) і збіжності (при $h \rightarrow 0$ розв'язок різницевого рівняння прямує до точного розв'язку).

Для прикладу наведемо програму розв'язку задачі Діріхле методом сіток для рівняння Лапласа на мові Turbo Pascal [1].

Розв'язати на сітці (5×5) задачу Діріхле:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad 0 < x < 1; \quad 0 < y < 1$$

$$u(0, y) = 0; \quad u(1, y) = y; \quad 0 \leq y \leq 1$$

$$u(x, 0) = 0; \quad u(x, 1) = x; \quad 0 \leq x \leq 1$$

Program Dirihle;

```
const
  n=4;
  m=4;
type
  vek1=array[0..n] of real;
```

```

    vek2=array[0..m] of real;
    matr=array[0..n+1,0..m+1] of real;

var
    i,j,m1,n1:integer;
    a,b,hx,hy,s,t,w,r,e:real;
    x:vek1;
    y:vek2;
    u:matr;

function f1(y:real):real;
begin
    f1:=<f1(y)>;
end;

function f2(y:real):real;
begin
    f2:=<f2(y)>;
end;

function f3(x:real):real;
begin
    f3:=<f3(x)>;
end;

function f4(x:real):real;
begin
    f4:=<f4(x)>;
end;

begin
    writeln('введіть a,b,e');
    readln(a,b,e);
    hx:=a/n;
    hy:=b/m;
    t:=sqr(hx/hy);
    for j:=0 to m do
    begin
        y[j]:=j*hy;
        u[0,j]:=f1(y[j]);
        u[n,j]:=f2(y[j]);
    end;
    for i:=0 to n do
    begin
        x[i]:=i*hx;
        u[i,0]:=f3(x[i]);
        u[i,m]:=f4(x[i]);
    end;
    m1:=m-1;
    n1:=n-1;
    {розв'язок СЛАР за методом Зейделя}
    for i:=1 to n1 do
        for j:=1 to m1 do
            u[i,j]:=1;{початкові значення}
        repeat

```

```

w:=0;
for i:=1 to n1 do
  for j:=1 to m1 do
    begin
      s:=(u[i-1,j]+u[i+1,j]+t*(u[i,j-1]+u[i,j+1]))/(2*(1+t));
      r:=abs(s-u[i,j]);
      if r>w then
        w:=r;
        u[i,j]:=s;
      end;
    until w<e;
  write('  y');
  for j:=0 to m do
    write('      ',y[j]:6:3);
  writeln;
  writeln('x');
  for i:=0 to n do
    begin
      write(x[i]:4:2);
      for j:=0 to m do
        write('      ',u[i,j]:8:3);
      writeln;
    end;
  end.

```

В програмі наведені наступні позначення:

n – число кроків сітки по x .

m – число кроків сітки по y .

a, b (A, B) – граничні точки області R .

ϵ – точність, з якою розв'язується СЛАР методом Зейделя.

h_x, h_y – крок сітки, відповідно по осі x та y .

$f_1(y), f_2(y), f_3(x), f_4(x)$ – відомі функції з граничних умов, відповідно по осі y та x .

x, y – одновимірні масиви розмірності $(n+1)$ та $(m+1)$ – вузли сітки.

u – двовимірний масив $(n+1)*(m+1)$ розв'язків задачі в вузлах сітки.

В програмі потрібно задати:

$n:=4; m:=4; f_1:=0; f_2:=y; f_3:=0; f_4:=x; a:=1; b:=1; \epsilon:=0.001$

Результат роботи програми:

	y 0.000	0.250	0.500	0.750	1.000
x					
0.00	0.000	0.000	0.000	0.000	0.000
0.25	0.000	0.063	0.126	0.188	0.250
0.50	0.000	0.126	0.251	0.375	0.500
0.75	0.000	0.188	0.375	0.563	0.750
1.00	0.000	0.250	0.500	0.750	1.000

Література:

1. Коссак О., Тумашова О., Коссак О. Методи наближених обчислень: Навч. посібн. – Львів: БаК, 2003. – 168 с.

МЕТОДИЧЕСКАЯ РАЗРАБОТКА КУРСА «КОМПЬЮТЕРНОЕ ДЕЛОПРОИЗВОДСТВО» ДЛЯ СПЕЦИАЛЬНОСТИ «ИНФОРМАТИКА И ЭКОНОМИКА»

О.Н. Туравинина

г. Кривой Рог, Криворожский государственный педагогический университет

Дисциплина «Компьютерное делопроизводство» сравнительно недавно читается на физико-математическом факультете КДПУ специальности «Информатика и экономика», поэтому методическая поддержка курса претерпевает постоянные изменения, что связано с подборкой наиболее актуальных для современного делопроизводства задач.

Само понятие «Компьютерное делопроизводство» сегодня выходит далеко за рамки умения правильного оформления документов и их подготовки с помощью текстового редактора и других подобных прикладных программ. Финансовый менеджер, секретарь-референт, и, собственно, любой уважающий себя специалист должны уметь производить статистическую обработку информации, финансовые расчеты и их анализ, уметь делать прогноз финансового развития, вести учет кадров, планировать производство и управлять сбытом.

Анализируя возможности многих прикладных программ, позволяющих выполнять указанные выше функции, особое внимание, по праву, следует уделить выполнению различных задач, используя табличный процессор Excel. Это объясняется тем, что на данный момент эта программа находится в числе лидеров по количеству функциональных возможностей для обработки информации, представленной в табличном виде, построения графиков и диаграмм: имеет удобный и несложный интерфейс пользователя, относится к семейству Microsoft Office, приобрести и установить который не составляет особых трудностей. На многих предприятиях, это не секрет, расчеты зарплаты и налогов, ведение учета кадров, затрат и т. п. производится именно в Microsoft Excel.

В методической разработке к курсу содержатся задания различной степени сложности, условно разделенные на 4 блока:

– *блок 1* содержит задания, направленные на формирование умения представлять текстовую информацию в табличном виде, применять математические и статистические функции, строить графики и диаграммы;

– *блок 2* включает задачи, связанные с расчетом зарплаты при различных условиях ее начисления и производимых удержаниях, что невозможно осуществить без знания простых и сложных логических функций;

– *блок 3* содержит набор финансовых задач на применение функций НОРМА, ПЛПРОЦ, ППЛАТ, ЧАСТОТА и т. д.;

– *блок 4* – это задачи, направленные на выполнение анализа финансовой ситуации.

При решении такого рода задач используются элементы математического моделирования, поскольку появляется возможность, изменяя значения переменных, влиять на конечный результат.

Перед каждым блоком производится подробное рассмотрение и демонстрация выполнения типового задания, после чего следуют задания для самостоятельного решения.

Ниже приводится пример подачи информации в методической разработке к блоку 4.

Тема: *Анализ данных с помощью MS Excel.*

Теоретическая часть.

Возможности анализа данных расширены специальными дополнительными средствами табличного процессора. Например, функция подбора параметра определяет значение в зависимых ячейках, если указан результат вычислений. Процедура поиска решения разрешает решать сложные задачи оптимизации при наличии системы ограничений. Мастер сводных таблиц разрешает наглядно представлять данные и создавать отчеты. Таблицы подстановки разрешают показывать результаты формул в зависимости от значений одной или двух переменных, которые используются в этих формулах.

Для того, чтобы все эти функции анализа данных были доступными пользователю, необходимо загрузить программу надстроек: *Надстройки – Установка функций надстроек*. Для этого используются соответствующие команды главного меню программы: *Сервис – Надстройки*. После выбора необходимой опции в появившемся окне необходимо нажать на *OK*, в результате чего пункты меню дополнятся соответствующими командами (рис. 1).

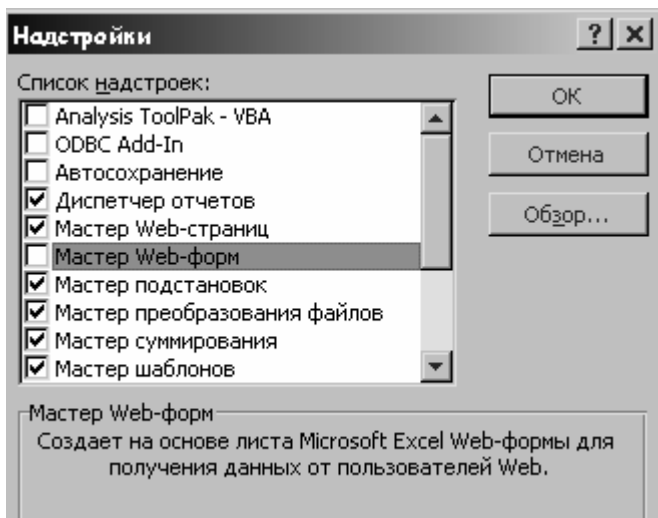


Рис. 1. Диалоговое окно Надстройки.

Подведение итогов.

Для подведения итогов используется команда *Итоги* пункта *Данные* главного меню.

Перед использованием этой команды данные следует представить в виде списка. Эта опция прибавляет строки промежуточных итогов для каждой группы элементов списка. При этом можно использовать разнообразные функции для подведения итогов (среднее значение в столбце, количество строк каждой группы, количество пустых ячеек, стандартное отклонение, и т.п.).

Для подведения итогов в списке, прежде всего, необходимо отсортировать записи по категориям (*Данные, Сортировка*). После этого необходимо активизировать одну из ячеек таблицы и выбрать из пункта *Данные* команду *Итоги*. В открывшемся окне *Промежуточные итоги* следует задать название столбца, который будет определять деление списка на группы, и определить функцию итоговых вычислений, задать название столбца, данные которого будут анализироваться (рис. 2). В случае добавления к итоговой таблице еще одного показателя для каждой группы записей, которая будет отображать результаты вычислений по другой формуле, необходимо еще раз открыть диалоговое окно *Промежуточные итоги* и задать в соответствующем поле этого окна соответствующую функцию.

Для отображения в таблице всех итоговых вычислений (за всеми последовательно выбранными функциями), необходимо исключить опцию *Заменить текущие итоги*.

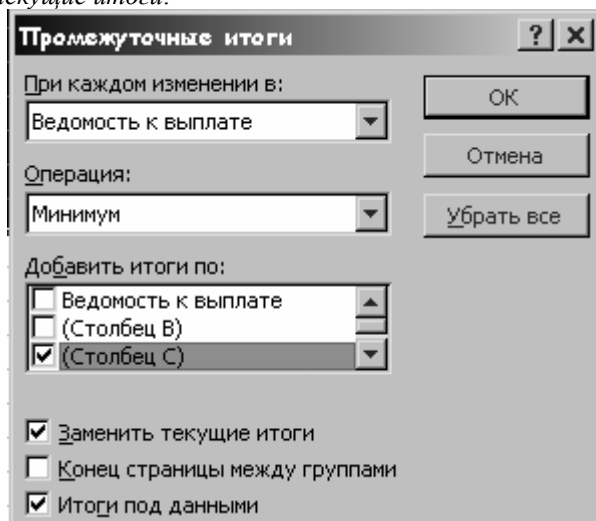


Рис. 2. Диалоговое окно Промежуточные итоги.

Для распечатки каждой группы данных с итоговыми вычислениями, нужно установить опцию *Конец страницы между группами*. Для уничтоже-

ния строк предназначена кнопка *Убрать все*.

Создание сводной таблицы.

Таблица, на основе которой создается сводная, должна содержать заголовки строк или столбиков, которые необходимо отобразить в результирующей таблице. Эти заголовки являются полями данных.

Мастер сводных таблиц загружается из пункта *Данные*, командой *Сводная таблица* главного меню. Мастер выполняет три действия:

- первое – определение источника данных;
- второе – определение диапазона данных;
- третье – создание макета.

Определение макета сводной таблицы выполняется кнопкой *Макет*, таким образом создается структура таблицы, что наилучшим образом отобразит информацию (рис. 3а, 3б).

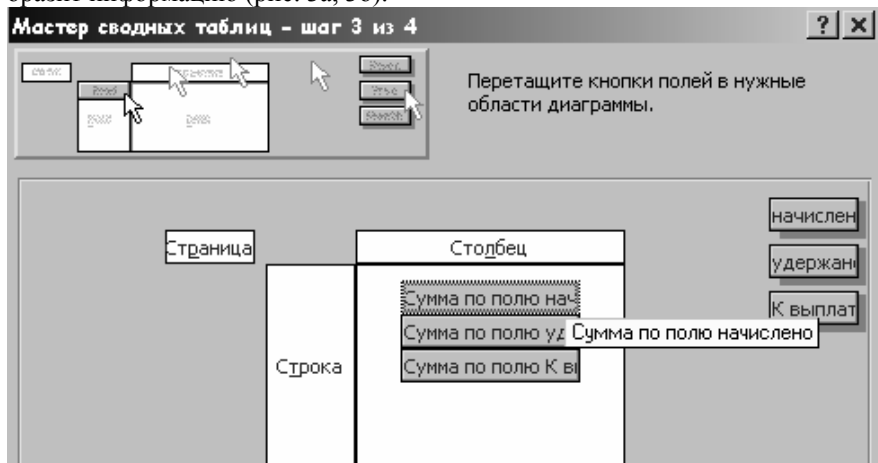


Рис. 3а. Создание макета сводной таблицы

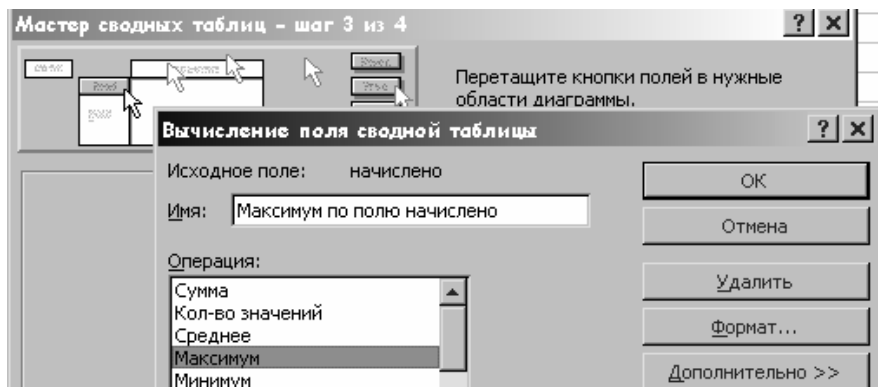


Рис. 3б. Создание макета сводной таблицы

В центре окна размещены области строк, столбцов, страниц и данных. Все заголовки таблицы находятся по правую сторону. В часть окна *Столбец* перетягиваются кнопки с названиями столбцов, которые являются полями, в часть *Данные* – кнопки с названиями столбцов, значения которых рассчитываются. По умолчанию итог, который будет рассчитываться, – это функция *Сумма*, для определения другой функции надо дважды щелкнуть на кнопке с названием поля в части *Данные*.

Кнопка *Параметры* разрешает открыть диалоговое окно для изменения параметров таблицы (форматирование, вид параметры источника данных). Редактирование сводной таблицы можно проводить с помощью соответствующих команд панели инструментов *Сводная таблица*.

Консолидация данных.

Для создания сводной таблицы на основе данных из разных таблиц используется команда *Консолидация* меню пункта *Данные* главного меню.

Консолидированная таблица размещается на новом листе рабочей книги. Сначала выбирается ячейка, которая будет началом диапазона ячеек с итоговыми данными. После этого выполняется команда *Консолидация* и заполняется диалоговое окно (рис. 4).

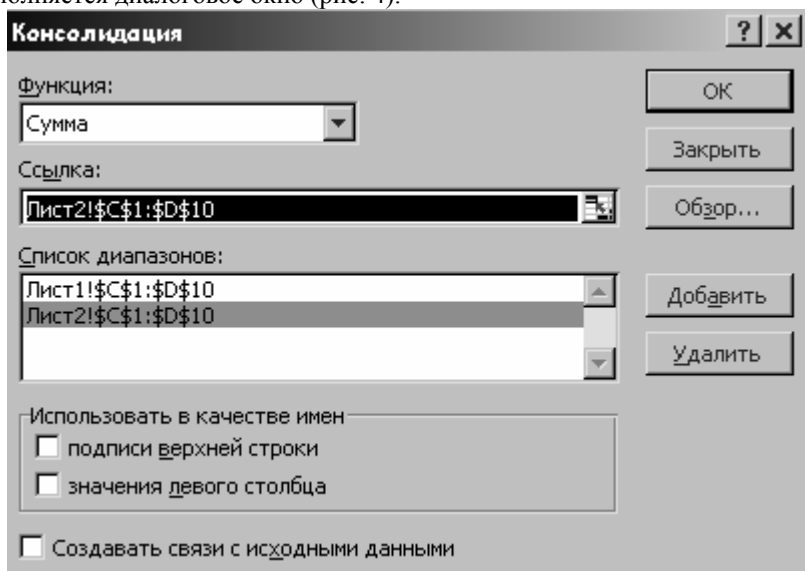



Рис. 4. Диалоговое окно Консолидация.

В окне выбирается функция (поле *Функция*), которая будет использоваться при объединении данных. Далее задаются координаты (поле *Ссылка*) всех диапазонов ячеек с данными, которые консолидируются. Если диапазоны находятся на разных листах, достаточно после введения первого диапазона перейти на другой лист, нажимая кнопку . Во время такого выде-

ления диапазонов окно *Консолидация* свертывается. По окончании выделения любого из диапазонов нажимается кнопка *Добавить*. В диапазон ячеек, который может консолидироваться с другими, необходимо включить и заголовки строк, иначе можно утратить ориентацию в таблице.

Задание.

1. На основе фактических материалов, полученных на производстве или из литературы создать входные таблицы.

2. Используя функции *Итоги* и *Консолидация*, создать итоговые сведения.

Литература:

1. Глушаков С.В., Сурядный А.С., Хачиров Т.С. Домашний ПК. – Харьков: Фолио, 2004. – 490 с.

УДОСКОНАЛЕННЯ ФОРМ І МЕТОДІВ КОНТРОЛЮ ЗНАТЬ З ІНФОРМАТИКИ У ВНЗ

І.Є. Фільо

м. Рівне, Національний університет водного господарства
та природокористування

Контроль знань є частиною навчального процесу і забезпечує зворотній зв'язок зі студентами. В ході контролю оцінюється ступінь і рівень навчання, фіксується об'єм праці, яка вкладена студентом при виконанні контрольної роботи. За результатами перевірки здійснюється управління навчальним процесом: аналізуються типові помилки, коректуються знання і уміння, проводиться коректування навчальних програм.

Для того щоб контроль знань був ефективним, цікавим, викликав активність та емоційний настрій у студентів, викладач має володіти інформацією щодо методів і форм контролю знань, постійно їх удосконалювати. Методика викладання курсу інформатики та комп'ютерних технологій відрізняється від інших методик, тому доцільним вбачається навести класифікацію методів контролю знань з інформатики, що можуть використовуватись у ВНЗ. Розглянута нижче класифікація адаптована, але в основному використовує схему, наведену в [1–3].

Методи контролю класифікують:

1. **За місцем контролю** на етапах навчання:

- *попередній* (вхідний) вид контролю особливо актуальний для інформатики, оскільки в одній групі можуть бути студенти з різним рівнем підготовки. Вхідний контроль дозволить визначити початковий рівень підготовки і можливість використання диференційованого підходу;
- *поточний* (оперативний) *контроль* виявляє рівень і ступінь підготовки студента в процесі навчання, реалізує діагностичну функцію і встановлює зворотній зв'язок з кожним студентом. Поточний контроль – це систематична перевірка і оцінювання освітніх результатів студента з конкретних тем та на окремих заняттях;
- *повторний контроль* передбачає перевірку знань паралельно з вивченням нового матеріалу, що сприяє міцності і системності знань студентів;
- *тематичний контроль* навчальних досягнень студента є обов'язковим. Основною одиницею оцінювання є навчальна тема.
- *періодичний контроль* здійснюється щодо цілого розділу або модуля. Метою такого методу контролю є діагностування якості засвоєння студентом структурних основ і взаємо-

зв'язків розділу, його особистих освітніх прирощень. Завдання періодичного контролю – навчаюче, оскільки студенти навчаються систематизації, узагальненню, цілісному баченню крупного блоку навчальної інформації і пов'язаної з нею діяльності.

- *підсумковий контроль* проводиться в кінці вивчення навчального курсу, як правило у вигляді іспиту або заліку. Даний метод контролю передбачає комплексну перевірку освітніх результатів з усіх ключових цілей і напрямків.

2. **За способом оцінювання:**

- *оціночна* технологія є традиційною – оцінка виставляється за п'ятибальною системою;
- *рейтингова* технологія заснована на наборі балів, одержаних за освоєння різних розділів (тем) програми, і їх підсумовуванні. При такій системі оцінювання можуть бути передбачені додаткові бали, які використовуються для заохочення найактивніших студентів;
- *якісна* технологія в основному використовує поєднання методу спостереження з експертною оцінкою знань і умінь. Оцінювання може бути представлено у вигляді висновків за допомогою ключових слів: «освоїли – не освоїли», «засвоїли – не засвоїли», «зрозуміли – не зрозуміли», «оволоділи – не оволоділи».

3. **За способом організації контролю:**

- *комп'ютерний*;
- *взаємний*;
- *контроль викладача*;
- *самоконтроль*.

4. **За ведучою функцією:**

- *діагностичний* контроль дозволяє здійснювати спостереження за динамікою розвитку знань, умінь і навичок, одержувати відомості про стан освоєння навчального матеріалу для своєчасної корекції, ефективної організації повторення, оптимізації навчального процесу;
- *стимулюючий* контроль допомагає налаштувати студента на систематичну роботу, ініціювати його творчу діяльність, активізувати увагу, розвинути ініціативу і самостійність, розкрити сильні і слабкі сторони;
- *констатуючий* контроль фіксує ступінь і рівень підготовки студента, використовується в процесі проведення іспитів і заліків.

5. **За способом отримання інформації в ході контролю:**

- *усний* контроль;

- *письмовий* контроль;
- *практичний* контроль.

На заняттях з інформатики можна використовувати різні форми контролю, як традиційні так і нетрадиційні. Головне пам'ятати: для того, щоб опитування пройшло ефективно, викладач повинен мати велику кількість дидактичних матеріалів і урізноманітнювати форми контролю.

Усна форма перевірки знань або усне опитування – найпоширеніша форма вимірювання знань. Усну форму доцільно використовувати лише при вимірюванні знань, пов'язаних із знанням та розумінням основних термінів та понять, що вивчаються. При усному опитуванні викладач розчленовує матеріал, що вивчається, на окремі смислові одиниці (частини) і по кожній з них ставить студентам питання. Будучи ефективним і найпоширенішим методом перевірки і оцінки знань, усний контроль має, проте, і свої недоліки. З його допомогою на занятті можна перевірити знання не більше 3–4 студентів. Тому на практиці застосовуються різні модифікації цього методу, зокрема, фронтальне і ущільнене опитування, а також “загальний бал” за заняття. До усних форм контролю відносяться:

- *фронтальне опитування*;
- *індивідуальне опитування*;
- *експрес-опитування*;
- *опитування-гра*;
- *усний іспит*;
- *дискусія*;
- *співбесіда*;
- *виступ на семінарі*;
- *опитування по ланцюгу*.

Письмова форма перевірки знань або письмова робота є пошуком більш точних, ніж при усній формі контролю методів вимірювання та оцінювання знань. До письмових форм контролю відносяться:

- *письмова контрольна робота*;
- *письмова самостійна робота*;
- *письмова домашня робота*;
- *розв'язування задач*;
- *теоретичний диктант*;
- *експрес-диктант*;
- *тестові завдання*;
- *письмовий залік*;
- *письмовий іспит*;
- *письмове оформлення лабораторних робіт*;
- *письмове оформлення індивідуальних курсових проєктів*;
- *написання рефератів*;
- *письмове розв'язування кросвордів*;
- *написання статті*.

Практична форма контролю є особливо важливою при проведенні занять з інформатики. Згідно сучасних вимог, студенти повинні оволодіти різноманітними глибокими знаннями та вміннями, практично використовувати персональний комп'ютер як: об'єкт вивчення, інструмент обробки інформації за допомогою готових прикладних програм, засіб навчання, інструмент створення програм у ході вивчення алгоритмізації та програмування, засіб комунікації. Такі вміння та знання студентів треба узагальнити, проконтролювати й оцінити. До практичних форм контролю відносяться:

- *демонстрація проекту;*
- *розробка демонстраційної презентації;*
- *лабораторна робота;*
- *практична робота;*
- *тестування;*
- *спостереження.*

Перевірка, і оцінка знань, умінь і навичок студентів є важливим структурним компонентом процесу навчання і відповідно до принципів систематичності і послідовності навчання повинна здійснюватися протягом всього періоду навчання. Необхідне виконання наступних умов:

- 1) жодне завдання не повинне бути залишено без перевірки і оцінювання з боку викладача;
- 2) обов'язкове повідомлення результатів перевірки;
- 3) максимальна участь студента в процесі перевірки виконаного їм завдання.

Істотне значення має також те, як відноситься студент до навчання, чи працює він постійно або ж ривками і т.д. Все це обумовлює необхідність постійно удосконалювати форми і методи контролю знань на заняттях з інформатики, вживати всю сукупність методів перевірки і оцінки знань, які наведені вище.

Література:

1. Бабанский Ю.К. Методы обучения в современной общеобразовательной школе. – М.: Просвещение, 1985.
2. Майоров А.Н. Мониторинг в образовании. – СПб.: Образование-Культура, 1998.
3. Морзе Н.В. Методика навчання інформатики. Навч. пос. – К.: Навчальна книга, 2003.

ИСПОЛЬЗОВАНИЕ ПАКЕТА СИМВОЛЬНОЙ МАТЕМАТИКИ MAXIMA ПРИ ИЗУЧЕНИИ ЛИНЕЙНЫХ СПИСКОВ В КУРСЕ «СТРУКТУРЫ ДАННЫХ И АЛГОРИТМЫ ИХ ОБРАБОТКИ»

Н.А. Хараджян

г. Кривой Рог, Криворожский государственный педагогический университет

Второе место после массивов по частоте использования в программах занимают списки. Поэтому при изучении темы «Линейные списки» в курсе «Структуры данных и алгоритмы их обработки» необходимо показать принципы работы со списками, структуру списка в более общих чертах, чем в конкретных реализациях на языках программирования C, Pascal. Этот курс направлен на изучении алгоритмов обработки различных типов данных, а не на их представление в языках программирования. Написание и отладка кода занимает большую часть времени, отведенного на изучение самих приемов работы со списками.

Альтернативным способом изучения линейных списков является использование пакетов программ символьной математики, в которых реализованы основные функции для работы со списками, содержащими любые типы данных, при этом конкретная реализация скрыта от студента (механизм работы с указателями, сохранение начала списка и т.д.).

Наиболее известными программами, которые позволяют выполнять символические вычисления, являются Mathematica, Maple, MathCad. Версии этих программ доступны пользователям различных операционных систем, в том числе Windows и Linux. Следует отметить, что эти программы стоят достаточно дорого.

Система компьютерной алгебры Maxima распространяется под лицензией GNU, что позволяет рекомендовать ее широкому кругу пользователей. Система *Maxima* доступна как пользователям ОС Linux, так и пользователям MS Windows.

Maxima – компьютерная программа для выполнения математических вычислений, символьных манипуляций, числовых вычислений и графического представления результатов. Она была создана Уильямом Шелтером в 1967 году. Одна из ее современных версий занимает объем в распакованном виде около 22 Mb.

Maxima предоставляет возможность работы с объектами-списками, содержащими не только однотипные, но и разнотипные элементы. В частности, элементами списков могут быть числа, константы, переменные, выражения и даже сами списки. Списки используются для конструирования более частных типов данных – массивов, матриц и векторов.

Часто математические или иные объекты содержат множество данных, которые желательно объединять под общим именем. Например, под объектом с именем M можно понимать квадратную матрицу 10×10 с общим

числом элементов, равным 100. Человека с именем Victor, например, можно охарактеризовать целым списком разных данных – символьным (фамилией, именем и отчеством), целочисленным (годом рождения), вещественным (ростом, объемом груди) и т.д.

На языке данной системы список – это совокупность произвольных данных указанных в квадратных скобках.

Для генерации списков с элементами, являющимися вещественными и целыми числами, а также выражениями, используется функция **makelist**.

Списки относятся к данным сложной структуры. Поэтому при работе с ними возникает необходимость контроля за структурой, иначе применение списков может привести к грубым ошибкам, как явным, сопровождаемым выдачей сообщения об ошибке, так и неявным.

Стек является частным случаем однонаправленного списка. Он относится к системам хранения данных динамического типа, его размеры непрерывно меняются по ходу вычислений.

В *Maxima* для работы со списками существуют следующие функции:

Listp(exp) – exp список?

Length(exp) – число элементов списка exp.

Member(exp,list) – выводит TRUE, если переменная exp входит в список list.

Cons(exp,list) – добавляет в начало списка list элемент exp.

Endcons(exp,list) – добавляет в конец списка list элемент exp.

Last(list) – возвращает последний элемент списка list.

First(list) – возвращает первый элемент списка list.

Rest(exp,n) – удаляет из списка n элементов.

А также, *Maxima* предоставляет ряд расширенных возможностей для работы со списками, выходящих за рамки обычных стековых операций.

Так, например, для расширения списка путем включения в него новых элементов используются следующие функции:

Append(list1,list2,list3,...) – объединяет списки в указанном порядке.

Delete(exp1,exp2) – удаляет элемент exp1 из списка exp2.

Reverse(list) – возвращает список с обратным порядком расположения элементов.

Sort(list) – сортирует элементы списка list в каноническом порядке.

Комбинирование списков позволяет создавать сложные структуры данных из более простых структур. Это может быть полезно при построении очередей, деревьев и иных структурных построений.

Для изучения основных приемов работы со списками студентам предлагается задание приведенное ниже, аналог которого взято из [1]:

Список содержит информацию о наличии автобусов в автобусном парке, сведения о каждом автобусе содержат:

– номер автобуса;

– фамилию и инициалы водителя;

– номер маршрута;
– признак того, где находится автобус – на маршруте или в парке.
Программа должна предусматривать реализацию следующего:
– начальное формирование данных обо всех автобусах в виде списка;
– выдаются сведения об автобусах, находящихся в парке, или об автобусах, находящихся на маршруте.

Создаем список из трех элементов. Признак наличия автобуса в парке – 1, на маршруте – соответственно 0.

Создаем первоначальный список, состоящий их трех элементов.

```
(D1) [[1, petrov, 0], [2, sidorov, 0], [5, ivanov, 1]]
```

Далее в цикле перебираем все элементы списка и ищем необходимые нам автобусы. Если сравнение найдено, мы выводим этот элемент на экран и удаляем его из списка, иначе – просто удаляем элемент из списка.

```
(C2) FOR A:1 THRU length(parkn) STEP 1 DO( if member(0,first(parkn)))  
then ( display( first(parkn) ), parkn: delete( first(parkn), parkn) ) else (parkn:  
delete( first(parkn), parkn));
```

В итоге на экран выводятся следующие результаты.

```
FIRST([[1, petrov, 0], [2, sidorov, 0], [5, ivanov, 1]]) = [1, petrov, 0]
```

```
FIRST([[2, sidorov, 0], [5, ivanov, 1]]) = [2, sidorov, 0]
```

Списки используются в следующих ситуациях: 1) в случае удаления и вставки элементов в середину; 2) если производится управление данными меняющегося размера, особенно, если доступ к ним происходит по принципу стека; 3) они используют память эффективнее, чем массивы, при наличии нескольких стеков, которые независимо друг от друга растут и уменьшаются; 4) использование в случае, когда информация внутренне связана в цепочку неизвестного заранее размера, например как последовательность слов в документе.

На основе приведенных функций можно также изучить алгоритмы работы с очередями.

Литература:

1. Павловская Т.А. С\С++ программирование на языках высокого уровня – СПб.: Питер, 2002 – 464 с.
2. Леонова Н.А. Использование пакета символьной математики *Mathima* в курсе “Компьютерные технологии в научных исследованиях”. // Теорія та методика навчання математики, фізики, інформатики: Зб. наук. праць. – Кривий Ріг: Видавничий відділ КДПУ, 2002.

ИЗЛОЖЕНИЕ РАЗДЕЛА «КОМПЬЮТЕРНЫЕ СЕТИ» ДИСЦИПЛИНЫ «ИНФОРМАТИКА И КОМПЬЮТЕРНАЯ ТЕХНИКА»

С.Н. Чаплыгина

г. Днепропетровск, Приднепровская государственная академия
строительства и архитектуры
marinapn@ukr.net

По какой бы специальности ни осуществлялась подготовка студентов в учебном заведении, умение владеть компьютером является неотъемлемой частью этой подготовки. Не возникает сомнения также и в том, что следующей ступенью после освоения основных приемов работы на персональном компьютере является умение работы в компьютерной сети. Обусловливается это двумя основными причинами:

- практически во всех организациях, где есть компьютеры, они объединены в сеть, что дает возможность сотрудникам использовать ресурсы не только своего компьютера, но и всех других компьютеров, куда разрешен доступ;
- Internet сейчас повсеместно используется как источник информации любого характера и как средство связи.

В настоящее время имеется достаточное количество литературы по компьютерным сетям. Однако для специальностей, непосредственно не связанных с вычислительной техникой и программированием, в планах отводится немного времени на этот раздел, в связи с чем встает вопрос об отборе тем для лекций и лабораторных работ. От преподавателя требуются значительные усилия по организации локальной сети, обеспечению ее работоспособности, подбору индивидуальных заданий и написанию методических указаний к выполнению лабораторных работ.

Излагая этот раздел, мы делим его на две части: локальные вычислительные сети и Internet. При этом основные сведения общего характера излагаются на лекциях, а лабораторные работы проводятся по следующим темам.

Работа в одноранговой сети

В аудитории была создана одноранговая сеть, объединяющая двенадцать компьютеров. Выполняя эту лабораторную работу, студенты знакомятся с локальной вычислительной сетью, ее компонентами, службами, установленными в сети. Каждый студент должен получить эту информацию непосредственно со своего компьютера. Согласно индивидуальным заданиям создаются общедоступные папки с разными типами доступа, подключаются сетевые диски, производится поиск информации на других компьютерах и работа с ней. Особый интерес у студентов вызывает помещение созданного ими документа (сообщения) в общедоступные папки, находящиеся

на компьютерах их сокурсников.

Электронная почта в локальной сети

Работа электронной почты осуществлялась с помощью программы OFMSRV, запущенной на одном из компьютеров, и Outlook Express, установленной на остальных компьютерах. При выполнении этой работы нужно добиться четкого понимания основных элементов, лежащих в основе функционирования электронной почты, таких как учетные записи, почтовые адреса, почтовые ящики и др. Выполняя эту работу, студенты посылали деловые письма, соответствующие своей специальности, на несколько компьютеров и получали ответы на них, а также изучали возможности Outlook Express. Обычно эта работа проходит оживленно. В заключение студентам предлагается сравнить электронную почту с обычной, отметить их общие свойства и преимущества первой.

Работа с обозревателем Internet Explorer

Значительная часть работы с сетью Internet осуществляется с помощью браузеров и поисковых систем. Одним из наиболее широко используемых обозревателей является Internet Explorer.

Целью этой работы является ознакомление с основными приемами работы с этим обозревателем, приобретение практических навыков в поиске нужной информации. Особое внимание уделяется таким понятиям, как адресация, Web-страница, Web-узел, гиперссылка.

Студенты получают задания на поиск конкретной информации с помощью разных средств поиска, сохранение найденной информации и работу с ней в автономном режиме. При этом осуществляется знакомство с различными поисковыми службами и их особенностями. Задания включают также использование Журнала, панели Ссылки, Избранного и других возможностей обозревателя.

Электронная почта в Internet

При выполнении этой работы использовалась служба Ukr.net. Для того, чтобы стать пользователем этой службы, нужно в ней зарегистрироваться и получить почтовый ящик. Это можно сделать со стартовой страницы этой службы, которая имеет адрес <http://www.ukr.net>. Загружая эту страницу, студенты получают пошаговую инструкцию, следуя которой, создают свой почтовый ящик на Ukr.net. После создания своего почтового ящика студенты посылают сообщения на адрес преподавателя.

Кроме своей основной функции, почтовые службы в настоящее время имеют ряд других, например, новостные. При выполнении работы студентам предлагается ознакомиться с этими разделами.

По поводу этой работы можно отметить, что когда скорость работы в Internet-классе бывает низкой и не все студенты успевают полностью выполнить задания и послать сообщения, те, кто не успел это сделать, по своей инициативе во внеурочное время приходят в Internet-класс или посылают свои сообщения с других компьютеров, не дожидаясь следующего занятия.

Создание Web-страниц

В этой работе по выбору преподавателя студентам предлагается один из способов создания Web-страницы: с помощью языка HTML, текстового редактора Word или программы FrontPage Express.

В методических указаниях приведено описание создания основных элементов Web-страниц любым из указанных способов. Индивидуальные задания содержат обязательные элементы и оставляют возможность проявить инициативу исполнителей. При выполнении этой работы студенты по своему желанию используют различные источники информации, в том числе и полученные недавно навыки поиска информации в Internet, обсуждают возникающие вопросы и свои намерения между собой и с преподавателем, стараются внести в оформление своей Web-страницы что-нибудь оригинальное.

Как известно, интерес является мощным стимулом познавательной деятельности. В связи с широким распространением в профессиональной деятельности и в быту получения информации из Internet и возможности общения через Internet интерес к этой области достаточно велик, что, в частности, проявляется в желании создать свою Web-страницу непохожей на другие.

После выполнения каждой лабораторной работы в рабочей тетради студентами составлялся отчет по этой работе. Он состоял из задания, краткого описания его выполнения и полученных результатов. Составление такого отчета дает возможность мысленно повторить выполненные действия, систематизировать их и способствует сознательному запоминанию проработываемого материала.

В заключение хотелось бы отметить, что при изучении этого раздела у студентов наблюдался большой интерес и активность при выполнении лабораторных работ по сравнению с другими разделами курса «Информатика и компьютерная техника».

ОСОБЛИВОСТІ ВИКЛАДАННЯ ДИСЦИПЛІНИ “ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ”

С.І. Шаповалова^а, О.Є. Круш^б

м. Київ, Національний технічний університет України “КПІ”

^а lana@aprodos.ntu-kpi.kiev.ua

^б krush@aprodos.ntu-kpi.kiev.ua

Постановка проблеми. Освітньо-професійна програма підготовки бакалавра з напрямку підготовки 0804 “Комп’ютерні науки” містить рекомендований перелік навчальних дисциплін і практик, до якого входить дисципліна “Об’єктно-орієнтоване програмування”. Ця дисципліна належить до циклу природничонаукової підготовки і є обов’язковою для вивчення студентами всіх спеціальностей, що відносяться до комп’ютерних наук. Володіння знаннями і практичними навичками з побудови об’єктно-орієнтованих систем є необхідним для сучасного спеціаліста.

Аналіз останніх досліджень. Серед величезної кількості навчальної літератури, що тим чи іншим чином висвітлює об’єктно-орієнтований підхід до програмування, досить багато літератури, в якій у доступній формі формулюються його основні принципи та визначення, але мало уваги приділяється прикладам, практичним завданням на створення об’єктної ієрархії та її використання. Автори зосереджуються на, власне, створенні об’єктів та класів тією чи іншою мовою програмування, мало або зовсім не висвітлюючи призначення та основні методи створення об’єктно-орієнтованих систем. Як результат, студенти знають, як створити клас чи об’єкт, але не знають, навіщо це потрібно і які саме класи потрібно створювати для більш гнучкої системи.

Існує також інший напрямок у літературі з теорії об’єктно-орієнтованого програмування, в якому висвітлюється його суть, але мало приділяється уваги практичному застосуванню побудованих моделей у тій чи іншій мові, що робить подібні книжки не популярними серед студентів. Наслідком цього є майже повна відсутність розуміння процесу проектування у студентів.

З учбових посібників, що розкривають суть процесу об’єктно-орієнтованого проектування, можна виділити [1] та [2]. Вони безумовно можуть стати базовими для постановки задачі на проектування та чіткого формулювання цілей побудови об’єктних моделей. Основними перевагами цих підручників є:

- чітке формулювання основних термінів і визначень;
- визначення концепції об’єктно-орієнтованого проектування та її специфіки;
- постановка проблеми об’єктно-орієнтованого проектування на різних рівнях абстракції та відповідні методи вирішення;

- аргументація та обґрунтування об'єктно-орієнтованого моделювання, чітке визначення цілей.

Недоліком підручників [1] та [2] є те, що автори не ставили собі за мету надання практичних навичок використання існуючих мов програмування для створення об'єктно-орієнтованих систем. При викладанні дисципліни “Об'єктно-орієнтоване програмування” підручники [1] і [2] доречно застосовувати для теоретичної частини постановки проблеми проектування та його цілей.

Для більш широкого розуміння об'єктно-орієнтованого програмування невід'ємною частиною дисципліни є практичні приклади на створення об'єктно-орієнтованих систем. Цьому повинна приділятися найбільша увага та виділятися найбільше часу практичних занять.

Оскільки на сьогодні найбільш розповсюдженими є такі мови програмування, як Pascal та C++, викладання дисципліни доцільно орієнтувати на обидві мови. Саме тому, для розгляду практичної реалізації рекомендуємо навчальні посібники [3], [4] та [5]. Крім того, різні мови програмування мають особливості реалізації об'єктів, тому всебічний розгляд об'єктно-орієнтованого програмування неможливий у межах однієї мови.

З вищезазначеного видно, що на самостійне вивчення курсу об'єктно-орієнтованого програмування витрачається досить багато часу, але це не означає, що матеріал буде охоплено всебічно і, зазвичай, залишаються без уваги досить важливі грані об'єктно-орієнтованої технології створення програмних систем.

Формулювання цілей статті (постановка завдання). Метою статті є формування розподілу теоретичного матеріалу дисципліни “Об'єктно-орієнтоване програмування” по розділах, обґрунтування вибору тем і завдань лабораторних робіт до них таким чином, щоб встигнути за семестр висвітлити основні задачі об'єктно-орієнтованого програмування та методи їх вирішення. Метою курсу є звернення уваги власне не на мови програмування як такі, а на якісне проектування об'єктно-орієнтованих систем. Вважається, що до вивчення цієї дисципліни студенти вже прослухали такі курси, як “Основи програмування та алгоритмічні мови”, “Структура та організація даних в ЕОМ” і тому набули певного досвіду з програмування мовами високого рівня.

Основа частина. Дисципліна “Об'єктно-орієнтоване програмування” повинна ознайомити студентів з технологією проектування об'єктно-орієнтованих систем та програмними засобам її реалізації.

Основними аспектами, що залишаються поза увагою при несистематичному вивченні матеріалу дисципліни є:

- по-перше, втілення об'єктної орієнтації всієї програмної системи. Без цього студенти створюють об'єктно-орієнтовані програми так, ніби вони просто пишуть процедурною мовою;
- по-друге, створення грамотної об'єктної ієрархії. Зазвичай студенти

або створюють дуже мало класів, що не дозволяє скористатись усіма перевагами об'єктно-орієнтованого засобу укладання програм, або створюють їх так багато, що всі переваги знищуються через громіздкість системи;

- по-третє, набуття практичних навичок застосування найпоширеніших мов програмування при створенні програмної реалізації об'єктно-орієнтованої системи. Студенти, що вже вміють програмувати мовами високого рівня, мають свій власний стиль програмування. При створенні програмної реалізації об'єктно-орієнтованої системи вони використовують саме його, і, якщо об'єктна модель вимагає суттєвих змін у стилі програмування, намагаються змінити модель, а не стиль програмування. Це призводить до некоректного використання можливостей мови програмування.

Для того, щоб запобігти виникненню вищезазначених проблем курс поділяється на два розділи з відповідними темами:

1. Проектування об'єктно-орієнтованих систем
 - 1.1. Основні визначення об'єктно-орієнтованого програмування.
 - 1.2. Призначення та функції об'єктно-орієнтованих систем.
 - 1.3. Принципи формалізації предметної області при об'єктно-орієнтованому підході.
 - 1.4. Засоби формалізації предметної області при об'єктно-орієнтованому підході.
2. Практична реалізація у розповсюджених мовах програмування C++ та Pascal
 - 2.1. Особливості створення класів і об'єктів у середовищах C++ та Turbo Pascal.
 - 2.2. Методи створення об'єктів. Звернення до складових об'єкту.
 - 2.3. Особливості наслідування. Реалізація поліморфізму: абстрактні, віртуальні, динамічні методи та перекриття методів у класах-спадкоємцях.
 - 2.4. Створення складних структур об'єктів за допомогою одно- і двохрівневих списків. Методи звільнення пам'яті.
 - 2.5. Особливості реалізації класів і об'єктів у середовищах Delphi та Visual C++.

Перша частина призначена для засвоєння принципів об'єктної орієнтації систем і навчання грамотному будуванию об'єктної моделі середовища, що проектується. Тому доцільно розпочати лекційний матеріал з коротких історичних відомостей щодо виникнення та розвитку об'єктно-орієнтованого програмування, його переваг перед процедурним програмуванням. Важливим також є чітке формулювання основних термінів та визначень, що дає можливість у наступних розділах розкривати особливості їх реалізації. Знайомство з різними методиками побудови об'єктних моделей,

порівняльний аналіз різних напрямків проектування допомагає студентам грамотно проектувати та орієнтуватися у розмаїтті об'єктно-орієнтованих систем і новітніх технологіях.

Матеріал першої частини дає велику кількість теоретичних відомостей щодо нового для студентів погляду на програмування, які дуже важко сприйняти і зрозуміти без прикладів. Тому матеріал необхідно супроводжувати прикладами проектування об'єктних моделей для різних предметних середовищ. Цю функцію виконують практичні заняття, передбачені навчальним планом.

Лекції другого розділу присвячені розкриттю особливостей реалізації об'єктно-орієнтованих моделей у мовах програмування C++, Pascal і середовищах проектування програмного забезпечення Visual C++, Delphi.

Мета циклу практичних занять полягає саме в тому, щоб студенти навчилися будувати об'єктні моделі різних предметних середовищ, ознайомились з основними помилками та проблемами, які при цьому виникають. Завдання на практичні заняття мають бути орієнтовані саме на процес проектування, а не на мову програмування.

Теми практичних занять:

1. Сімейство об'єктів, які описують прості геометричні фігури. Методи графічного зображення та переміщення фігур на екрані монітора.
2. Об'єктні моделі новітніх графічних середовищ.
3. Сімейство об'єктів, які описують арифметичні вирази.
4. Структури об'єктів-списки, їх динамічний розподіл та звільнення.
5. Об'єктні моделі однорівневих списків новітніх об'єктно-орієнтованих середовищ.
6. Дворівневі списки. Взаємодія автомобілів на перетині доріг. Потоки сніжинок.

Мета циклу лабораторних робіт полягає у тому, щоб студенти закріпили теоретичний матеріал і отримали практичні навички, самостійно опрацьовуючи весь процес проектування об'єктно-орієнтованої програмної системи.

Теми лабораторних робіт:

1. Створення об'єктів. Порівняльний аналіз з записами. Методи ініціалізації.
2. Створення сімейства об'єктів за заданою предметною областю. Методи графічного зображення та переміщення фігур на екрані монітора.
3. Реалізація поліморфізму за допомогою віртуальних методів.
4. Створення статичних та динамічних екземплярів об'єктів. Складні приклади взаємодії об'єктів.
5. Створення структур об'єктів-списків, їх динамічний розподіл та звільнення.
6. Реалізація складних прикладів з використанням списків.

7. Реалізація дворівневих списків.

8. Створення сімейства об'єктів за заданою предметною областю у об'єктно-орієнтованому середовищі.

Перші сім лабораторних робіт студенти виконують у двох мовах програмування C++ та Pascal за індивідуальним завданням, а восьму – у об'єктно-орієнтованому середовищі Delphi або Visual C++. Для виконання восьмої лабораторної роботи студентам рекомендується ознайомитись з роботою [6].

Висновки. Метою вивчення дисципліни “Об'єктно-орієнтоване програмування” є придбання знань, достатніх для орієнтування у багатогранному світі об'єктно-орієнтованих систем, засвоєння студентами базових принципів та отримання основних навичок створення об'єктно-орієнтованих програм.

Оскільки в дисципліні викладається багато матеріалу, видається велика кількість завдань і використовується різноманітне програмне забезпечення, доречно створити електронний посібник з наступними розділами: глосарій, структурований теоретичний матеріал, завдання лабораторних робіт, приклади їх вирішення, викликання та завантаження зовнішніх спеціалізованих програмних засобів, приклади їх застосування.

Література:

1. Буч Г. Объектно-ориентированное проектирование с приложениями: Пер. с англ. – К.: Диалектика, 1993.
2. Шлеер С., Меллор С. Объектно-ориентированный анализ: моделирование мира в состояниях: Пер. с англ. – К.: Диалектика, 1993. – 240 с.
3. Зуев Е.А. Язык программирования Turbo Pascal 6.0, 7.0. – М.: Веста, Радио и связь, 1993. – 384 с.
4. Шилдт Г. Теория и практика C++: Пер. с англ. – СПб.: BHV–Санкт-Петербург, 2000. – 416 с.
5. Дьюхарст С, Старк К. Программирование на C++: пер. с англ. – К.: Диасофт, 1993. – 272 с.
6. Фаронов В.В. DELPHI 5. Учебный курс. – М.: HOЛИДЖ, 2000. – 640 с.

ВИКОРИСТАННЯ ЕЛЕКТРОННИХ ПІДРУЧНИКІВ В НАВЧАЛЬНОМУ ПРОЦЕСІ

С.В. Шаров

м. Мелітополь, Мелітопольський державний педагогічний університет
seg@mpu.melitopol.net

В наш час багато організацій займаються розробкою та впровадженням дистанційних курсів. В основі будь-якого курсу знаходиться, як правило, електронний підручник, що дає можливість тим, хто навчається, одержати необхідний обсяг навчальної інформації, а також пропонує додатковий матеріал, різноманітні завдання для контролю та самоконтролю, перелік рекомендованої літератури, посилання на тематичні ресурси Інтернет тощо.

Використання електронного підручника в навчальному процесі різноманітне:

– підручник може використовуватися для самостійної роботи учнів та студентів, тому що він полегшує розуміння навчального матеріалу за рахунок інших, ніж у друкованій навчальній літературі, засобів подання матеріалу: індуктивний підхід, вплив на слухову та емоційну пам'ять. Він виконує роль терплячого наставника, надаючи велику кількість роз'яснень, повторень, підказок, надає великі можливості для самоперевірки на всіх етапах роботи. До того ж правильно спроектований електронний підручник адаптується відповідно до потреб та рівня підготовки того, хто навчається;

– використання електронного підручника корисно і на практичних заняттях, тому що він дозволяє використовувати комп'ютерну підтримку для рішення великої кількості задач, звільняє час для аналізу отриманих рішень та їхньої графічної інтерпретації. Це дозволяє викладачу проводити заняття у формі самостійної роботи за комп'ютерами, залишаючи за собою роль керівника та консультанта; швидко й ефективно контролювати знання учнів; вирішувати зміст та рівень складності завдань;

– електронний підручник дозволяє викладачу виносити на лекції і практичні заняття матеріал у меншому, але найбільш істотному по змісту обсязі, залишаючи для самостійної роботи те, що виявилось поза рамками аудиторних занять;

– такий вид підручника звільняє від стомлюючої перевірки домашніх завдань, типових розрахунків і контрольних робіт, передоручаючи цю роботу комп'ютеру, дозволяє оптимізувати співвідношення між кількістю прикладів і задач, розглянутих в аудиторії та завданнями, що задаються як домашнє завдання, дозволяє індивідуалізувати роботу зі студентами стосовно теоретичного матеріалу, домашніх завдань та контрольних заходів.

Для розробки електронного підручника є певні методичні рекомендації, які складаються з декількох пунктів.

На першому етапі доцільно підібрати друковані та електронні видання,

що найбільш повно відповідають програмі курсу або спецкурсу та які містять велику кількість прикладів і задач.

На другому етапі розробляється власний зміст електронного підручника, тобто робиться розбивка матеріалу на розділи та підрозділи, що складаються з модулів, мінімальних за обсягом, але замкнених за змістом, а також складається перелік понять, необхідних та достатніх для оволодіння навчальним предметом.

На третьому етапі переробляються тексти джерел відповідно до змісту і структури модулів; розробляється система контекстних довідок (допомога); визначаються зв'язки між модулями та інші гіпертекстні зв'язки. Таким чином, підготовляється проект гіпертексту для комп'ютерної реалізації підручника.

На четвертому етапі підручник реалізується у електронній формі. У результаті створюється примітивне електронне видання, що вже може бути використане в навчальних цілях.

На п'ятому етапі розробляється комп'ютерна підтримка: визначається, які дії в кожному конкретному випадку буде виконувати комп'ютер; розробляються інструкції для користувачів по застосуванню електронного підручника.

У результаті створюється працюючий електронний підручник, що має властивості, які роблять його корисним для застосування на аудиторних та самостійних заняттях і зручним для викладачів.

На шостому етапі змінюються способи пояснення окремих понять і стверджень та відбираються тексти для заміни мультимедійними матеріалами.

На сьомому етапі розробляються тексти звукового супроводу окремих модулів з метою розвантаження екрану від текстової інформації і використання слухової пам'яті учня для полегшення розуміння та запам'ятовування матеріалу, який вивчається. Розроблені тексти звукового супроводу записуються на диктофон і реалізуються на комп'ютері.

На восьмому етапі розробляються сценарії візуалізації модулів для досягнення більшої наочності, максимального розвантаження екрану від текстової інформації та використання емоційної пам'яті учнів або студентів для полегшення розуміння і запам'ятовування матеріалу. Створюється візуалізація текстів, тобто комп'ютерне втілення розроблених сценаріїв з використанням малюнків, графіків та анімації.

Електронний підручник має додержуватися певних вимог, а саме: повинен містити систематизований матеріал за робочою програмою навчального курсу, забезпечувати творче та активне оволодіння знаннями, уміннями та навичками в цій області; відрізнятися високим рівнем виконання і художнього оформлення, повнотою інформації, якістю методичного інструментарію та технічного виконання, наочністю, логічністю і послідовністю викладу.

Опираючись на ці вимоги та етапи створення, ми зробили спробу створити свій електронний підручник, який є допоміжним засобом для самостійного вивчення спецкурсу “Нові інформаційні технології”, який вивчається в Мелітопольському державному педагогічному університеті студентами гуманітарних спеціальностей, що не мали змогу відвідати окремі лекційні та практичні заняття.

Структура підручника складається з тематичних розділів відповідно до навчальної програми курсу. В кожному розділі подано:

- лекційний матеріал;
- матеріал для самостійного вивчення;
- запитання для самоперевірки;
- практичні роботи.

При виконанні практичної роботи студент в залежності від рівня своєї підготовки до даної теми може вибрати один з трьох рівнів складності.

На рівні А студентам пропонується тільки приклад кінцевого результату або коротка інструкція до виконання даної роботи.

Рівень складності Б передбачає приклад завдання та методичні вказівки до виконання цієї роботи (повна інструкція).

На рівні С студентам пропонується повна інструкція виконання практичної роботи та перелік відеофрагментів найбільш складних пунктів даного завдання.

Після проходження кожного розділу студенти відповідають на тестові завдання (програмний комплекс тематичного оцінювання знань учнів, який є окремим програмним засобом), до отриманих балів додається підрахована певним чином кількість балів, отриманих за виконання практичних робіт в залежності від рівня складності.

Робота над даним електронним підручником ще не завершена. Є певні результати, окремі модулі вже готові, але продовжується робота над створенням відеоматеріалу та звукового супроводження до практичних робіт. Також передбачається створити журнал індивідуального проходження студентами тематичних розділів, який буде зберігатися на особистій дискеті кожного студента. До цього журналу ввійдуть кількість опрацьованих практичних робіт та рівень, на якому вони були пройдені. Це дозволить викладачу простежити індивідуальну криву проходження студентом всіх розділів даного курсу.

ВИКОРИСТАННЯ ТЕРМІНАЛЬНИХ ІНФОРМАЦІЙНИХ СИСТЕМ НА ЗАНЯТТЯХ З ІНФОРМАТИКИ

О.В. Шматко

м. Харків, Академія цивільного захисту України

fd.apbu@list.ru

Ні для кого ні секрет, що комп'ютерна техніка коштує досить дорого і не завжди знаходяться гроші на відновлення парку комп'ютерів у навчальних закладах, у той час як практично в будь-якому вузі можна знайти морально застарілі, але проте ще цілком працездатні комп'ютери з найпростішими материнськими платами, процесорами 386, 486 і 8-16 МВ оперативної пам'яті. Встановити сучасні програмні продукти на такий комп'ютер неможливо, а списати шкода. Виходом у такій ситуації можуть бути забуті в нашій країні термінальні системи. Дійсно, у часи «великих комп'ютерів» виконання всіх додатків і збереження даних на сервері було справою досить таки простою. Потім з появою персональних ПК, користувальницькі робочі місця почали здобувати все більші можливості, ставати незалежними – наступила ера децентралізованого комп'ютинга. Але термінальні системи не зникли, і сьогодні застосування термінальних інформаційних систем (ТІС) у навчальних закладах дозволяє зручним грамотним способом побудувати роботу інформаційної структури ВНЗ, забезпечивши високий ступінь безпеки, зручне адміністрування і можливість у майбутньому безболісно нарощувати масштаби і потужності ІС (а також змусити студентів займатися справою в комп'ютерному класі).

Прикладом ефективного використання терміналів у навчальному процесі може служити система, встановлена на кафедрі інформатики й обчислювальної техніки Академії цивільного захисту України (м. Харків). Співробітники кафедри зштовхнулися з проблемою застарілого устаткування, а також виключення дій студентів, не пов'язаних з навчальним процесом.

У якості хоста тут виступає сервер на базі процесора AMD Sempron 2300+ socket-A BUS=333 Mhz 3D Now Pro, оснащений 512 МВ оперативної пам'яті, накопичувачем 40 GB. Локальна мережа поєднує 12 морально застарілих комп'ютерів із процесорами Pentium 133-433, з 16-32 МВ оперативної пам'яті. Програмний базис рішення складається з ОС Windows 2000 Server SP4. Ресурсами мережі викладачі кафедри користуються на заняттях з предметів «Інформатика і комп'ютерна техніка», «Основи інформатики й обчислювальної техніки» та інших дисциплін кафедри. У системі прекрасно працюють такі програми, як MSOffice 2000 Resource Kit, WinRAR, WinCOM, Opera, Maple 6, Draft Choice. Немаловажну роль зіграла дешевина такої мережі. Кафедра одержала повноцінний працездатний комп'ютерний клас із сучасним ПО, витративши близько 300\$ на придбання сервера, а також мережних карт і кабелю.

З огляду на витрати на модернізацію, технічну підтримку й адміністрування термінальної системи можна з упевненістю сказати, що ТІС є прогресивними засобами навчання у вищих навчальних закладах.

ПРИМЕНЕНИЕ МУЛЬТИМЕДИА ТЕХНОЛОГИЙ ПРИ ПРЕПОДАВАНИИ ИНФОРМАТИКИ

И.А. Яковлева

г. Харьков, Академия гражданской защиты Украины

fd.apbu@list.ru

Восприятие учебного материала часто во многом зависит от качества использованных в учебном пособии иллюстраций. Особенно это касается такого предмета, как информатика. При чтении лекций очень трудно, как говорится, «на пальцах» объяснить назначение кнопок, элементов окна или приемов управления мышью. Использование мультимедийных технологий кардинально меняет ситуацию. В этом случае любой объект может быть представлен не только в строго определенной, зафиксированной форме, но его можно сделать «живым». Возможно то, что называется интерактивным общением с объектом. Технология интерактивного представления объектов позволяет производить необходимые действия не «в голове», а прямо на экране, и тут же видеть результат, а не представлять, опять же, его в уме.

Для создания интерактивных мультимедийных презентаций на кафедре информатики и вычислительной техники использовалась стандартная программа электронных презентаций MS PowerPoint, язык гипертекстовой разметки HTML 4.0, а также программа создания анимационных рисунков Adobe Image Ready.

Программа PowerPoint дает очень наглядное представление о элементах диалоговых окон Windows, позволяет разнообразить методику подачи материала, в игровой и занимательной форме закрепить и обогатить знания, ранее полученные студентами.

PowerPoint позволяет создавать мультимедийный презентации, которые кроме текста могут включать в себя рисунки, диаграммы, анимационные объекты.

Система электронных презентаций позволяет создавать интерактивные презентации, содержащие гиперссылки как внутри презентации так и на другие презентации.

При чтении курса использовались мультимедиа презентации и мультимедиа проектор. Для лучшего усвоение материала, каждая лекция сопровождается удобным мультимедийным электронным учебником, который содержит материал лекции, а также задания для практической работы. Учебник написан на языке HTML 4.0 и содержит интуитивно понятный интерфейс, а также анимационные рисунки, которые позволяют слушателю легко выполнить задание. Для оценки знаний и итогового контроля использовалась система контроля знаний Test2000, которая содержит базу данных вопросов по дисциплине. Система тестирования позволяет задать время на выполнения теста, количество вопросов, а также критерии оценки. Все ин-

формация о результатах тестирования, количестве попыток сохраняется в базе данных системы.

Опыт применения такого комплекса современных информационных технологий показывает, что компьютерные презентации являются одним из самых эффективных методов представления и изучения любого материала. При представлении материала в графиках, картинках, таблицах, тезисах, виртуальных моделях включаются механизмы не только звуковой, но и зрительной и ассоциативной памяти. Компьютерные презентации позволяют сделать преподавание информатики содержательнее, интереснее, зрелищнее, эмоциональнее, нагляднее, эффективнее.

*Захист інформації
та інформаційна безпека*

ПРОГРАМНА РЕАЛІЗАЦІЯ СИМЕТРИЧНИХ ТА АСИМЕТРИЧНИХ КРИПТОСИСТЕМ ТА ЇЇ ВИКОРИСТАННЯ У НАВЧАЛЬНОМУ ПРОЦЕСІ

Х.О. Засадна¹, Р.Б. Попович²

¹ м. Львів, Львівський банківський інститут НБУ

² м. Львів, Національний університет “Львівська політехніка”
zasadna@lbi.wubn.net, kt@lbi.wubn.net

У Львівському банківському інституті НБУ та Національному університеті “Львівська політехніка” викладається дисципліна “Основи криптології”. Програма дисципліни передбачає вивчення таких тем.

Тема 1. Базові поняття захисту інформації. У цій темі основна увага приділяється законодавству України в галузі криптографії; використанню комп’ютерних технологій у банках України, послугам служби захисту інформації в інформаційно-обчислювальній мережі НБУ; опису криптології як науки та областям її застосування.

Тема 2. Симетричні криптосистеми. У рамках даної теми викладаються початкові засади криптографії та класичні симетричні криптосистеми: шифри підстановки і перестановки, поліалфавітні шифри, шифр одноразового блокноту. Поряд з ними розглядаються сучасні симетричні криптосистеми: потокові шифри та блокові шифри (стандарти шифрування DES та AES).

Тема 3. Асиметричні криптосистеми. Присвячена вивченню криптосистеми RSA з відкритим ключем, процедури утворення та перевірки цифрового підпису, їх застосуванню в системі електронних банківських послуг.

Для поглиблення знань з теоретичної частини предмету студентам рекомендуються посібники та монографії [1–5], методичні рекомендації [6], нормативні та законодавчі акти [7], [8].

З метою закріплення теоретичних знань на практичних заняттях з теми 2 студентам пропонується освоєння низки програм, а саме:

- програма “Шифр зсуву”, призначена для шифрування та дешифрування тексту українською та англійською мовами за допомогою шифру зсуву. Вхідна інформація вводиться вручну або з текстового файлу. На рис. 1 зображено робоче вікно програми;
- програма “Шифр Віженера” для шифрування та дешифрування тексту за допомогою шифру зсуву з аналогічним інтерфейсом;
- програма “Одноразовий блокнот”, яка виконує генерування ключа як випадкової послідовності бітів (користувач задає довжину ключа), шифрування та дешифрування повідомлень. Програма реалізує теоретично абсолютно надійний шифр;
- програма для реалізації симетричної потокової криптосистеми. Ключ для шифрування й дешифрування у цьому разі утворюється з використанням

каскаду Голмана [4]. Каскад Голмана складається з послідовності зсувових регістрів з лінійним зворотним зв'язком (рис. 2). На сьогодні вважається, що такий генератор ключа забезпечує прийнятну стійкість до зламування, коли число регістрів у каскаді приблизно рівне 15–20.

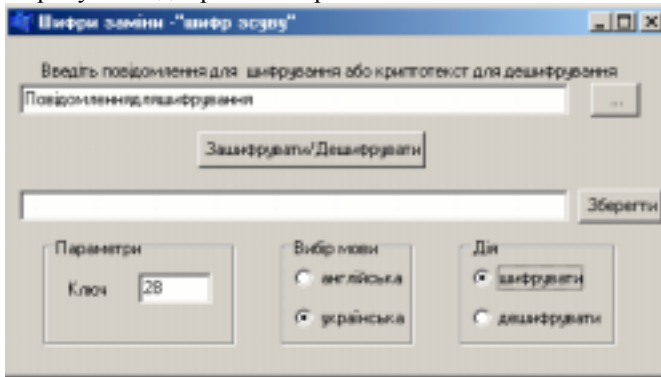


Рис. 1. Робоче вікно програми “Шифр зсуву”

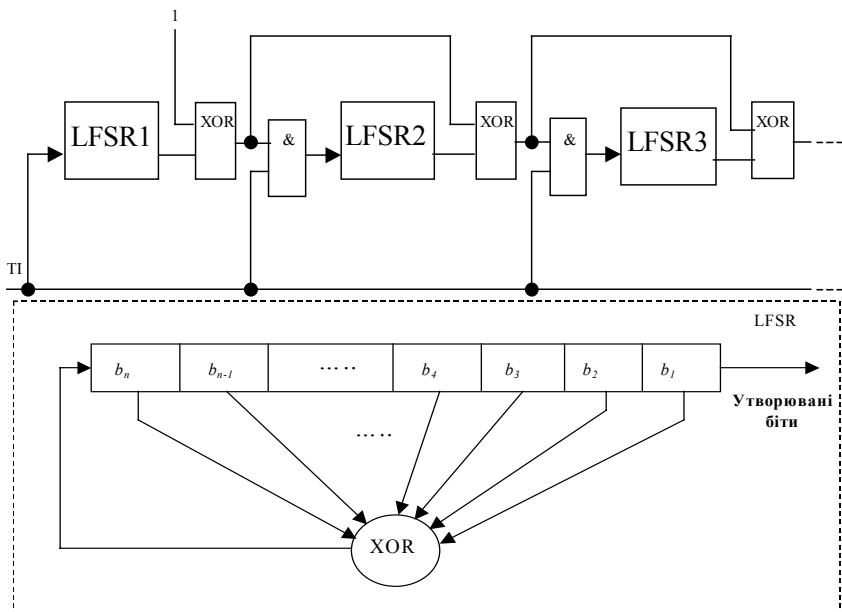


Рис. 2. Структурна схема генератора ключа шифрування/дешифрування для симетричної потокової криптосистеми

Слід зауважити, що обмін криптотекстами під час практичних занять виконується в межах різних аудиторій по локальній мережі, при цьому тає-

мний ключ пересилається через командний рядок MS DOS.

Згадані перші три програми реалізують нескладні алгоритми. Вони написані силами студентів Національного університету “Львівська політехніка” та інженерів лабораторії Львівського банківського інституту.

Програма для реалізації симетричної потокової криптосистеми на основі каскаду Голмана створена викладацьким персоналом.

При вивченні теми 3 використовуються програми для побудови системи RSA, створені інженерним та викладацьким персоналом.

Зокрема, студентам пропонується створити шаблон для генерації ключів у криптосистемі RSA засобами редактора MS Excel (рис. 3, 4).

	A	B	C	D	E	F	G	H	I	J
1	p	67					Відкритий ключ			
2	q	47					e	2627		
3	n	3149					n	3149		
4	phi	3036		НОД(e, phi)			Тайний ключ			
5	e	2627		1			d	2843		
6										
7										
8	Прямий хід алгоритму Евкліда					Обернений хід алгоритму Евкліда				
9	Ділене	Частка	Дільник	Остача	n	q	Pn	Qn		
10	2627	0	3036	2627	-2	q	0	1		
11	3036	1	2627	409	-1	q	1	0		
12	2627	6	409	173	0	0	0	1		
13	409	2	173	63	1	1	1	1		
14	173	2	63	47	2	6	6	7		
15	63	1	47	16	3	2	13	15		
16	47	2	16	15	4	2	32	37		
17	16	1	15	1	5	1	46	52		
18	15	15	1	0	6	2	122	141		
19					7	1	167	193		
20					8	15	2527	3036		

Рис. 3. Шаблон для генерації ключа у системі RSA

	A	B	C	D	E	G	H	I	J	
1	p	67				Відкритий ключ				
2	q	47				n	=B5			
3	n	=B1*B2				n	=B3			
4	phi	=(B1-1)*(B2-1)		НОД(e, phi)		Тайний ключ				
5	e	2627		=НОД(B5, B4)		d	=ОСТАТ(-J19;J20)			
6										
7										
8	Прямий хід алгоритму Евкліда					Обернений хід алгоритму Евкліда				
9	Ділене	Частка	Дільник	Остача	n	q	Pn	Qn		
10	=B5	=ЦЕЛОЕ(A10/C10)	=B4	=ОСТАТ(A10, C10)	-2	q	0	1		
11	=C10	=ЦЕЛОЕ(A11/C11)	=D10	=ОСТАТ(A11, C11)	-1	q	1	0		
12	=C11	=ЦЕЛОЕ(A12/C12)	=D11	=ОСТАТ(A12, C12)	0	=B10	=I10+I11*H12	=J10+J11*H12		
13	=C12	=ЦЕЛОЕ(A13/C13)	=D12	=ОСТАТ(A13, C13)	1	=B11	=I11+I12*H13	=J11+J12*H13		
14	=C13	=ЦЕЛОЕ(A14/C14)	=D13	=ОСТАТ(A14, C14)	2	=B12	=I12+I13*H14	=J12+J13*H14		
15	=C14	=ЦЕЛОЕ(A15/C15)	=D14	=ОСТАТ(A15, C15)	3	=B13	=I13+I14*H15	=J13+J14*H15		
16	=C15	=ЦЕЛОЕ(A16/C16)	=D15	=ОСТАТ(A16, C16)	4	=B14	=I14+I15*H16	=J14+J15*H16		
17	=C16	=ЦЕЛОЕ(A17/C17)	=D16	=ОСТАТ(A17, C17)	5	=B15	=I15+I16*H17	=J15+J16*H17		
18	=C17	=ЦЕЛОЕ(A18/C18)	=D17	=ОСТАТ(A18, C18)	6	=B16	=I16+I17*H18	=J16+J17*H18		
19					7	=B17	=I17+I18*H19	=J17+J18*H19		
20					8	=B18	=I18+I19*H20	=J18+J19*H20		
21										

Рис. 4. Формули шаблону для генерації ключа у системі RSA

Хоча наявні у шаблоні числа p , q , n , phi , e (зокрема відкритий і таємний ключі) у цьому разі є невеликими й не є такими, які реально використовуються на практиці, проте вони дозволяють студентам добре зрозуміти принципи побудови криптосистеми RSA. Якщо б зразу бралися числа з сотнями десяткових розрядів, то таке розуміння було б суттєво ускладнене.

Разом з тим студентам слід показати реальні числа, які можуть використовуватися в криптосистемі RSA. Для цього викладацьким персоналом створена програма для тестування великих натуральних чисел (сотні десяткових розрядів) на простоту (рис. 5).

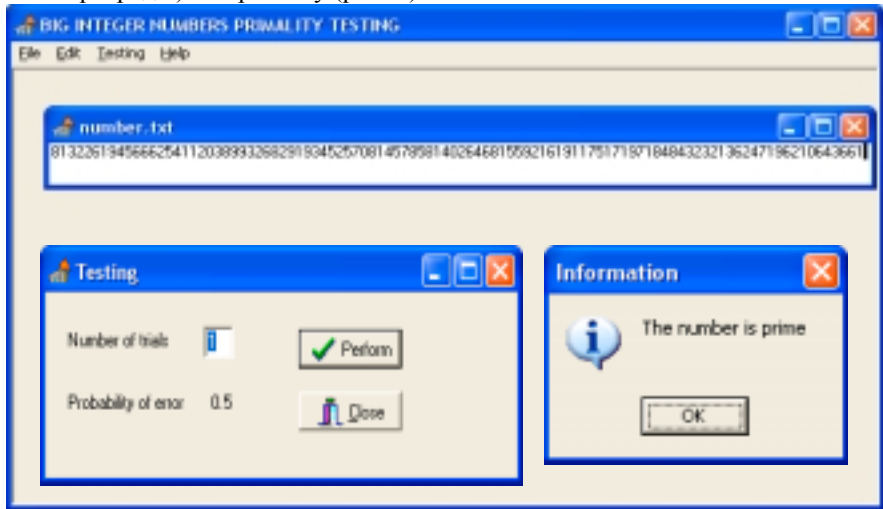


Рис. 5. Вікно програми тестування великих чисел на простоту

Дамо для ілюстрації такий приклад. Використовуючи описану програму, під час практичних занять спільними зусиллями викладача та студентів, показано, що наведені далі числа

$p=8132261945666254112038993268291934525708145785814026468155921619117517197184843232136247196210643661$

та

$q=6751592284822506657951467840983533389258228499577326266824366847039368903297074786737395335288189391$

зі 100 десятковими розрядами є простими. Ці числа можна використати для побудови сучасної стійкої до зламування криптосистеми RSA (генерації її публічного та таємного ключів). Зауважимо, що число n у цьому разі буде мати 200 десяткових розрядів. Зламування криптосистеми RSA зводиться до спроби його розкладу на два прості множники p та q . Разом з тим найбільше розкладене на сьогодні число мало лише 174 десяткових розряди, а для виконання цього розкладу були задіяні значні людські та обчислювальні ресурси [8].

Усі програми, які використовуються в навчальному процесі, написані на мові програмування Object Pascal в середовищі візуального програмування Delphi 7.0, що дозволило швидко створювати необхідні інтерфейси.

Література:

1. Методи захисту банківської інформації: Навчальний посібник/ Задірака В.К., Олексюк О.С., Недашковський М.О. – Київ: Вища школа, 1999. – 261 с.
2. Бабичев С.Г., Гончаров В.В., Серов Р.Е. Основы современной криптографии. – Москва: Горячая линия-Телеком, 2001. – 120 с.
3. Введение в криптографию / Под общей ред Яценко В.В. – Санкт-Петербург: Питер, 2001. – 288 с.
4. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си: – Москва, Триумф, 2003.
5. Смець В., Мельник А., Попович Р. Сучасна криптографія. Основні поняття. – Львів: БАК, 2003. – 144 с.
6. Методичні рекомендації щодо виконання практичних завдань із дисципліни “Комп’ютерний практикум” (криптографічний захист інформації) / Уклад. Р.Б. Попович, Х.О. Засадна. – Львів: ЛБІ НБУ, 2004. – 60 с.
7. www.rada.gov.ua
8. www.rsasecurity.com

ПРАКТИКУМ ПО ОСНОВАМ ЗАЩИТЫ ИНФОРМАЦИИ

Е.А. Кононова, Н.В. Моисеенко

г. Кривой Рог, Криворожский государственный педагогический университет

В курсе «Основы защиты информации», как правило, основное внимание уделяется криптографическим методам защиты. В криптографии сообщение шифруется таким образом, что непосвященные люди прочесть его не могут. Они могут получить зашифрованную информацию, но прочесть ее сможет только тот, кто имеет соответствующий ключ. Однако существуют и другие методы. Один из них (ненаучно-фантастический) – создание абсолютно надежного защищенного канала, второй – сокрытие факта передачи информации (стеганография, которую иногда ошибочно причисляют к криптографии). Программную реализацию одного из вариантов стеганографии мы предлагаем студентам в качестве лабораторной работы.

Стеганография служит для передачи секретов в других сообщениях, так что спрятано само существование секрета. Как правило, отправитель пишет какое-нибудь неприметное сообщение, а затем прячет секретное сообщение на том же листе бумаги. История знает такие приемы, как невидимые чернила, невидимые простому глазу пометки у букв, плохо заметные отличия в написании букв, пометки карандашом машинописных символов, решетки, покрывающие большую часть сообщения, кроме нескольких символов и тому подобное.

Важнейшее условие стеганографии – никто не должен заподозрить, что носитель информации скрывает какую-то тайну. Для того, чтобы спрятать сообщение в другом тексте, требуется определенная ловкость. Эта задача существенно облегчается при использовании компьютеров. В настоящее время можно незаметно и быстро спрятать информацию в компьютерном файле.

Принцип стеганографии

Для того, чтобы сокрытая информация осталась необнаруженной ни визуально, ни по объему файла, не должно быть заметно, что над файлом были проведены какие-то операции. Наиболее популярный способ, позволяющий реализовать такое сокрытие, – это LSB-алгоритм (Least Significant Bit). При этом скрываемый файл бит за битом включается в отдельные пиксели, причем в самую последнюю позицию двоичного числа. LSB-алгоритм оказывает наименьшее влияние на значение двоичного числа.

Например, двоичное представление десятичного числа 64 имеет следующий вид:

$$01000000=2^6=64.$$

Если изменить стоящий справа бит (это и есть LSB), то получится:

$$01000001=2^6+2^0=65.$$

Таким образом, десятичное число изменилось всего на 1. Если такую же

операцию выполнить над слева стоящим битом, т.е. Most Significant Bit (MSB), то получим:

$$11000000=2^7+2^6=128+64=192.$$

Как видно, результат существенно отличается от первоначального значения.

Поскольку сходные цвета в цветовой палитре имеют близлежащие числовые значения, то при использовании LSB-алгоритма цветовая информация изменяется незначительно.

Приведем пример. Букву А, в двоичном представлении 10000011, можно спрятать в 8 пикселях. Если последним 8 битам соответствующих двоичных чисел придать значения, приведенные ниже, то при перезаписи буквы А двоичная последовательность будет иметь следующий вид:

Пиксель 1: 00100111→00100111

Пиксель 2: 10101001→10101000

Пиксель 3: 01011000→01011000

Пиксель 4: 00111101→0011110

Пиксель 5: 11001000→11001000

Пиксель 6: 01011001→01011000

Пиксель 7: 11001000→11001001

Пиксель 8: 00100111→00100111

Изменились только четыре подчеркнутых бита. Если цветовая палитра широка, то искажение цветов минимально и невооруженным глазом незаметно.

Выбор носителя

Для стеганографии важную роль играет выбор носителя. Графические файлы форматов BMP и GIF, использующие простейшие алгоритмы сжатия, сохраняют изображение без потери его качества, и поэтому более пригодны в качестве носителей информации, чем, например, JPEG-файлы. Конечно, JPEG-сжатие позволяет сэкономить на жестком диске больше места, однако с потерей качества изображения. Поэтому, если после сокрытия информации в JPEG-файле будет выполнено его сжатие, а затем – разуплотнение, то нельзя будет полностью восстановить скрываемую информацию.

Мы спрятали текст в графическом изображении (рис. 1), заменяя младший значащий бит изображения битом сообщения. Графическое изображение при этом изменилось совсем незаметно (рис. 2). Большинство графических стандартов определяют больше цветовых градаций, чем способен различить человеческий глаз.

Кроме того, для целей стеганографии глубина цвета графического файла в 24 бита более приемлема, чем в 8 бит. Если при 256 цветах изменить цветовой код пикселя на единицу, то можно получить совершенно другой цвет. При этом в изображении будут искажены цвета, поэтому наиболее приемлемым решением будет использование изображений с градацией серого на 256 оттенков. Так, в черно-белой картинке 1024x1024 пиксела мож-

но спрятать сообщение в 64 Кбайт. Многие общедоступные программы могут проделывать подобный фокус.

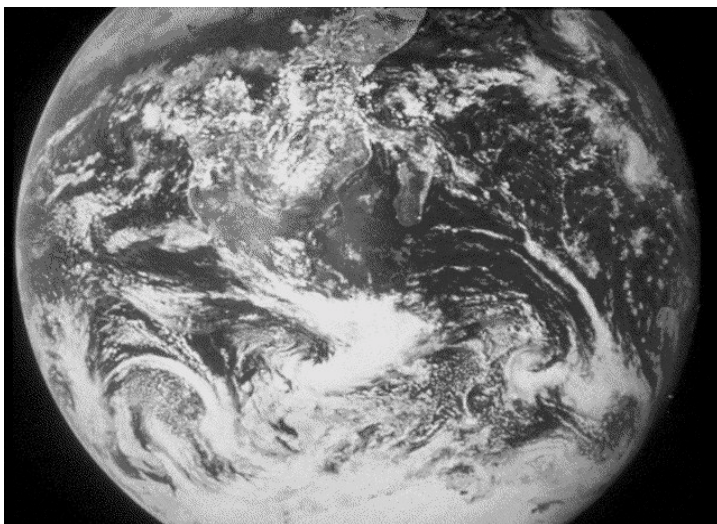


Рис. 1. Рисунок без секретов

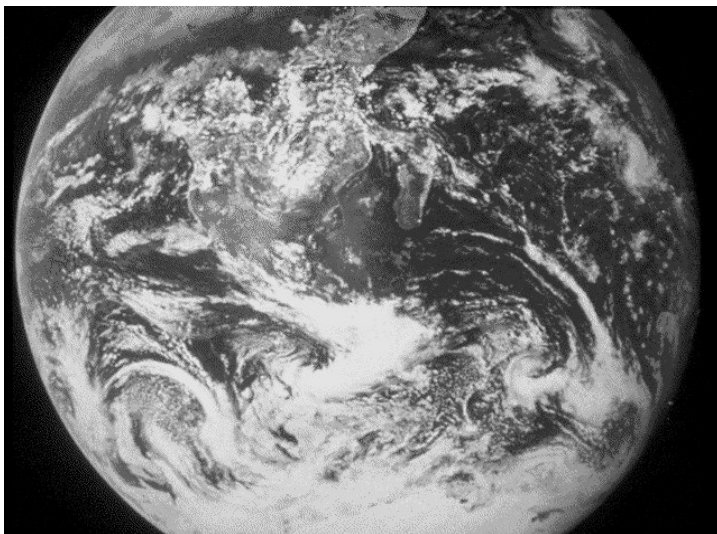


Рис. 2. Рисунок, в котором спрятано текстовое сообщение

Звуковые файлы в WAV- или VOC- формате также можно использовать в качестве носителей информации. Эти файлы тоже относительно не-

чувствительны к изменениям, если использовать LSB-алгоритм. При этом сделанные изменения на слух неразличимы.

О лабораторной работе

Задача, которую должны решить студенты в ходе выполнения лабораторной работы, состоит из двух частей. Первая разрабатываемая программа прячет текстовый файл в графическом. Здесь обязательно нужно обращать внимание студентов на формат файла, поскольку в начале файла располагается заголовок, который не должен принимать участия в сокрытии информации. Если заголовок будет модифицирован, то результат будет непредсказуемым. Вторая программа должна восстановить спрятанный текстовый файл.

Как правило, эта работа вызывает живой интерес студентов, поскольку реализуется достаточно быстро и есть возможность увидеть результаты своей работы в достаточно наглядной форме.

ИНФОРМАЦИОННЫЙ ТЕРРОРИЗМ: ПОНЯТИЕ, ТЕРМИНОЛОГИЯ, ПРОТИВОДЕЙСТВИЕ

Г.Ю. Маклаков

г. Севастополь, Севастопольский национальный технический университет

Всемирная федерация ученых в августе 2000 года первой в списке угроз человечеству в XXI веке поставила угрозу информационной безопасности [1]. Таким образом констатировался факт о том, что угрозы в информационной сфере преобладают над угрозами в экологии, энергетике, демографии. Индустрия информационных услуг является сегодня одной из наиболее динамичной сфер мировой экономики, сравнимой по доходности с топливно-энергетическим комплексом, автомобилестроением, производством сельскохозяйственной продукции. На долю информационных технологий приходится до 15% от общего объема мировой торговли [2].

Стремительное развитие информационных технологий породило новый вид преступности – компьютерную преступность, а переход на методы электронного управления технологическими процессами способствовал появлению нового вида терроризма. Такой вид терроризма часто называют кибертерроризмом [3], но встречаются и другие понятия – компьютерный терроризм [5], электронный терроризм [4], гипертерроризм [2]. Многообразие определений затрудняет выработку стратегии борьбы с терроризмом, связанным с информационными технологиями. За основу возьмем определение терроризма, сформулированное начальником кафедры уголовного права и криминологии Национального университета внутренних дел МВД Украины Емельяновым В.П. [6, с. 153]: «терроризм, то есть совершение или угроза совершения взрыва, поджога или иных общеопасных деяний, могущих повлечь гибель людей или иные тяжкие последствия и направленных на устрашение населения в целях понуждения государства, международной организации, физического или юридического лица или группы лиц к совершению или отказу от совершения какого-либо действия». Далее примем во внимание, что в настоящее время активизируется информационное противоборство, постоянно ведутся информационные войны.

Совсем недавно был отмечен такой специфический вид терроризма, как «ядерный шантаж». В начале 1999 года по электронной почте в адрес правительств более чем 20 стран (США, Великобритании, Израиля, Австрии и др.) были направлены сообщения от имени офицеров российской воинской части, расположенной в городе Козельске Калужской области и имеющей на вооружении стратегические ракеты шахтного базирования. В этих письмах информировалось, что командный состав части недоволен «унизительным положением России», и содержалась угроза «самовольно произвести пуски ракет по целям, расположенным в столицах и промышленных центрах западных стран». Кроме того, анонимы традиционно требовали выпла-

ты крупной денежной суммы [2]. В этой связи правительства ряда ведущих стран выразили МИД России серьезную обеспокоенность случившимся и попросили оказать содействие в розыске вымогателей. В результате проведенной ФСБ России расследования преступники были задержаны. Ими оказались два жителя Калуги, не являющиеся военнослужащими [2]. Можно согласиться с мнением Малышкина В.Г. и Коновалова А.М. [2] о том, что такой вид терроризма, можно классифицировать как специфический вид кибертерроризма. Вместе с тем, если проанализировать определение кибертерроризма, данное Голубевым А.А. [3, 4 с.50]: «кибертерроризм – преднамеренная, политически мотивированная атака на информацию, обрабатываемую компьютером, компьютерную систему и сети, которая создает опасность для жизни или здоровья людей или наступления других тяжких последствий, если такие действия были содеянные с целью нарушения общественной безопасности, запугивания населения, провокации военного конфликта». В приведенном выше примере «атака на информацию, обрабатываемую компьютером» не осуществлялась. Рассмотрим еще один пример.

19 декабря 1997 года по национальному телевидению Японии демонстрировался анимационный фильм, содержащий контаминацию цветовой гаммы, мигания визуальной информации, от просмотра которого десятки людей получили психофизические расстройства различной тяжести [2]. Налицо ситуация, «которая создает опасность для жизни или здоровья людей или наступления других тяжких последствий» однако «атака на информацию, обрабатываемую компьютером» не осуществлялась. Таким образом, подобный случай также не вписывается в определение кибертерроризма, приведенное выше.

Не претендуя на окончательную классификацию явления терроризма в эпоху информационного века, хотелось бы подчеркнуть, что для выработки адекватных мер противодействия терроризму, совершаемому с помощью информационных технологий необходимо дать достаточно точные определения исходных понятий.

Не вдаваясь в тонкости юридического анализа существующих понятий терроризма, считаю целесообразным ввести понятие «информационного терроризма» (за основу взято определение терроризма, сформулированное Емельяновым В.П. [6, с. 153]): «информационный терроризм – есть совершение или угроза совершения с помощью информационных технологий и/или информационного оружия взрыва, поджога или иных общепасных деяний, могущих повлечь гибель людей или иные тяжкие последствия и направленных на устрашение населения в целях понуждения государства, международной организации, физического или юридического лица или группы лиц к совершению или отказу от совершения какого-либо действия». Не трудно заметить, что такое определение включает в себя понятия «кибертерроризма» и «электронного терроризма», т.е. является более общим. Данное определение является рабочим и служит, прежде всего, для

разработки тактики и стратегии борьбы с терроризмом, связанным с использованием информационных технологий.

Проблема информационного терроризма может быть решена только комплексным путем.

Во-первых, это правовые методы обеспечения информационной безопасности государства, т.е. разработка нормативных правовых актов, регламентирующих отношения в информационной сфере, и нормативных методических документов по вопросам обеспечения информационной безопасности.

Во-вторых, это организационно-психологические методы, направленные на совершенствование организационной структуры государственных и коммерческих предприятий (включая сертификацию средств защиты информации, лицензирование деятельности в области защиты и стандартизация средств защиты информации), и самое главное, формирование высоких нравственно-этических норм у сотрудников, обслуживающих информационные системы критических предприятий и производств.

В-третьих, это создание и совершенствование систем обеспечения информационной безопасности и предупреждения нападения.

Рассмотрим более подробно третью составляющую. В плане совершенствования системы обеспечения информационной безопасности можно выделить следующие направления:

- разработка, использование и совершенствование средств защиты информации и методов контроля эффективности этих средств, развитие защищенных телекоммуникационных систем, повышение надежности специального программного обеспечения;

- создание систем и средств предотвращения несанкционированного доступа к обрабатываемой информации и специальных воздействий, вызывающих разрушение, уничтожение, искажение информации;

- выявление технических устройств и программ, представляющих опасность для нормального функционирования информационно-телекоммуникационных систем, предотвращение перехвата информации по техническим каналам, применение криптографических средств защиты информации при ее хранении, обработке и передаче по каналам связи, контроль за выполнением специальных требований по защите информации;

- формирование системы мониторинга показателей качества информационной защиты.

Одной из самых больших угроз для защищаемых компьютерных систем является атака посредством использования программных закладок. Программная закладка – это специальная скрытно внедренная в защищенную систему программа (или специально дописанный фрагмент пользовательской программы), позволяющая злоумышленнику путем модификации свойств системы защиты осуществлять несанкционированный доступ (НСД) к тем или иным ресурсам системы (в частности, к конфиденциальной

информации).

Если программная закладка написана грамотно, то после того, как она внедрена в систему, обнаружить ее стандартными средствами администрирования практически невозможно, она может функционировать неограниченно долгое время, и на протяжении всего этого времени внедривший ее злоумышленник имеет практически неограниченный доступ к системным ресурсам. Как правило, программные закладки выявляются либо из-за неаккуратного использования злоумышленниками полученной с их помощью информации, либо из-за допущенных при их программировании ошибок, либо чисто случайно.

Для организации надежного предохранения от программных закладок необходимо, чтобы система защиты была устойчива к ошибкам программного обеспечения и администраторов: даже если злоумышленнику удалось внедрить в нее программную закладку, он не должен иметь возможности причинить значительный ущерб конфиденциальности и целостности хранимой информации. Можно предложить следующие меры защиты, повышающие устойчивость системы к ошибкам:

- контроль целостности кода системного программного обеспечения (СПО);
- контроль целостности конфигурации системы защиты и ее интерфейсных связей, регистрация их изменений;
- мониторинг потоков данных.

Контроль целостности кода СПО. Если внедренная в систему программная закладка модифицирует СПО, она будет выявлена при ближайшей проверке целостности кода (поскольку программная закладка может модифицировать СПО не только на диске, но и в оперативной памяти, необходим контроль не только исполняемых файлов, но и их образов, загруженных в память).

Контроль целостности конфигурации системы защиты и ее интерфейсных связей и регистрация их изменений. Если программная закладка внедрена в систему защиты путем модификации ее конфигурации или интерфейсных связей, закладка также будет выявлена при ближайшей проверке их целостности.

Мониторинг потоков данных, которые могут быть использованы программными закладками в процессе функционирования и которые практически не реализуются легальным программным обеспечением.

Создание в системе объектов-ловушек (объектов, доступ пользователя к которым невозможен без использования программных закладок). Любые успешные обращения пользователей к объектам-ловушкам должны регистрироваться. Например, если подсистема аудита ОС Windows NT зафиксировала успешное обращение к файлу с атрибутами защиты Everyone – No Access, то это означает, что подсистема разграничения доступа изменена программной закладкой.

Эти рекомендации носят общий характер. При использовании конкретных систем защиты могут понадобиться и другие меры защиты от программных закладок, зависящие от особенностей защищаемой системы. Выявить программную закладку любого известного типа сегодня не представляет особых трудностей. Основную же опасность представляют закладки ранее неизвестных типов, для которых разработанные прежде средства защиты неэффективны.

Большую роль в борьбе информационным терроризмом должен сыграть активный мониторинг сетевого пространства. На микроуровне – это системы мониторинга самой компьютерной системы (персонального компьютера) и локальной сети. В частности, для мониторинга локальной сети используются специализированные программы – снифферы, позволяющие перехватывать переписку по локальной сети (Kerio Network Monitor 2.1.1., Winsniffer 2.0, Smb File Sniffer 1.0.0.1 и др.). На макроуровне – это системы мониторинга региональных компьютерных сетей (в пределе – фрагмента сети Интернет).

Следует подчеркнуть, что сниффинг информации является, по сути дела, нарушением тайны переписки и наряду с прослушиванием линий является противозаконным. Поэтому такой съем информации должен производиться в строгом соответствии с требованиями законодательства Украины.

Экспоненциальный рост компьютерной преступности обязывает силовые ведомства любого государства предпринимать соответствующие контрмеры, в частности осуществлять разведывательные и контрразведывательные операции. С этой точки зрения является оправданным принятие законодательных актов регламентирующих проведение оперативно-розыскных мероприятий. В России, например, развиваются и совершенствуются системы типа СОРМ-2 (анализ Интернет-трафика – накопление и обработка информации по ключевым словам). Федеральным правительством Германии в январе 2002 года утверждено «Распоряжение по технической и организационной реализации мер контроля за телекоммуникациями», которое имеет силу закона [7]. Введение таких законодательных актов целесообразно еще потому, что, как показывает мировая практика, из десяти разоблаченных контрразведкой агентов девять проваливаются на связи [7]. Системы мониторинга как раз и создаются на условиях, которые обеспечивают возможность доступа к любому сеансу связи абонента во время пользования услугами сети телекоммуникаций. Такие системы позволяют в реальном времени гарантированно идентифицировать и копировать содержание сеансов связи с использованием современных технологий.

Службой безопасности Украины разработан проект Закона Украины «О мониторинге телекоммуникаций» [8]. Этот документ призван в законодательном порядке закрепить и усовершенствовать механизм проведения правоохранительными органами оперативно-технических мероприятий на каналах связи, в частности, таких, как телефонная связь или электронная почта.

та. Разработка данного законопроекта вызвана необходимостью усовершенствования правовых и технологических условий обеспечения национальной безопасности в информационной сфере. Законы Украины «Об оперативно-розыскной деятельности», «О борьбе с терроризмом», «О разведывательных органах» и «О контрразведывательной деятельности» определяют снятие информации с каналов связи лишь как одну из функций правоохранительных органов. Проект закона «О мониторинге телекоммуникаций» детализует процедуру и условия создания интегрированной системы мониторинга, а также порядок его законного проведения субъектами ОРД [7].

Следует подчеркнуть, что новый закон не предусматривает предоставления правоохранительным органам дополнительных полномочий. Он необходим для того, чтобы урегулировать отношения специально уполномоченного государственного органа и операторов телекоммуникаций при внедрении и применении новых современных систем мониторинга. Закон определяет технические, организационные и режимные мероприятия для обеспечения мониторинга и контроля над его осуществлением в соответствии с Конституцией Украины и действующим законодательством. Законопроект «О мониторинге телекоммуникаций» внесен на рассмотрение Верховного Совета Украины.

Информационное оружие становится идеальным средством для террористов, а информационный терроризм может стать угрозой существованию целых государств, что делает вопрос информационной безопасности важным аспектом национальной и международной безопасности. Опасность от использования информационных средств в преступных целях одинаково велика как для высокоразвитых стран, так и для стран со слаборазвитой экономикой.

Литература:

1. Информационные вызовы национальной и международной безопасности. Под общ. ред. А.В. Федорова. – М.: ПИР-Центр, 2001.
2. Малышкин В.Г., Коновалов А.М. Реалии «информационного апокалипсиса»: киберпреступность, гипертерроризм, гипероружие. – Доклады Института Европы РАН. – 2003. – № 111. – М.: Изд-во «Квадрат-С», 2003.
3. Голубев В.А. Кибертерроризм как новая форма терроризма. – http://www.crime-research.org/library/Gol_tem3.htm. – 10.04.02
4. Голубев В.А. Электронный терроризм – новое лицо терроризма. // Компьютерная преступность и кибертерроризм: Сб. научн. тр. Центр исследования компьютерной преступности. Вып. 1., 2004. – Запорожье. Изд-во гуманитарного университета «ЗИГМУ», 2004. – С.49-56.
5. Соколов А.В., Степанюк О.М. Защита от компьютерного терроризма. Справочное пособие. – Спб.: БХВ-Петербург, 2002.
6. Емельянов В.П. Терроризм и преступления с признаками терроризма:

уголовно-правовое исследование. – Спб.; Изд-во «Юридический центр Пресс», 2002.

7. Кого и как «слушает» спецслужба. Эксклюзивное интервью газете «2000» заместителя Председателя СБ Украины генерал-лейтенанта А. Герасимова. Газета «2000». – 17.10.2003 (<http://www.sbu.gov.ua/pres/>)
8. Проект закона Украины «О мониторинге телекоммуникаций». – <http://www.sbu.gov.ua/base/promon1-4.shtml>. – 15.05.04
9. Чирило Дж. Защита от хакеров для профессионалов. – Спб.: Питер, 2002.

РОЛЬ ЧЕЛОВЕЧЕСКОГО ФАКТОРА ПРИ ПОДГОТОВКЕ СПЕЦИАЛИСТОВ В ОБЛАСТИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ В ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЯХ

Г.Ю. Маклаков

г. Севастополь, Севастопольский национальный технический университет

Одним из важных условий повышения качества подготовки специалистов в области информационной безопасности является формирование высоких нравственных качеств у студентов. Недостаточное внимание к человеческому фактору, как правило, представляет собой более значительную угрозу, чем использование новейших технических средств для добывания конфиденциальной информации. Под понятием «человеческий фактор» психологи понимают «совокупность свойств человека оператора, влияющих на эффективность системы "человек-машина"» [1]. Представляется целесообразным расширить определение – это интегральная характеристика личности, определяющая надежность защиты информации при ее получении, хранении и переработке в автоматизированных технико-биологических системах. Невзирая на разнообразие и постоянное совершенствование специальной техники для защиты информации, люди остаются самым слабым звеном в человеко-машинных системах, одним из самых вероятных источников утечки информации.

Проблему человеческого фактора при подготовке специалистов в области информационной безопасности целесообразно решать в двух направлениях: совершенствование технологии профотбора на специальности, связанные с защитой информации и оптимизации воспитательной работы в процессе обучения. В учебных заведениях, связанных с подготовкой специалистов в области информационной безопасности, целесообразно создавать специальные подразделения (лаборатории, группы, службы и т.п.), которые смогли бы заниматься изучением мотивации студентов к совершению противоправных действий в области информационных технологий и выработкой рекомендаций для оперативной корректировки учебно-воспитательной работы среди молодежи. Одним из главных направлений деятельности таких подразделений должно быть проведение профориентационной работы среди молодежи и обязательного тестирования абитуриентов на их профессиональную пригодность. Такие подразделения смогли бы решать еще одну достаточно важную задачу – использование влияния лидеров социальных групп для борьбы и предотвращения компьютерных преступлений. Такое направление успешно развивается в США. По мнению Д.Л. Шиндера [2], одним из способов сокращения количества киберпреступлений является использование влияния лидеров социальных групп. Как показали социологические исследования, проведенные в США [2], влияние авторитетов в социальных группах действует на поведение людей. Так,

снижение количества курильщиков в США в значительной степени связано с социальным «клеймом» курильщика.

Таким образом, одним из важных факторов повышения качества подготовки специалистов в области информационной безопасности должны быть меры по ужесточению режима отбора на специальности, связанные с защитой информационных технологий. Можно предложить следующие пути развития технологии профотбора:

1) создание эталонных моделей студента и специалистов в многомерном пространстве профессионально важных качеств;

2) отражение в содержании профессионально важных качеств познавательных способностей личности, адаптационных возможностей в профессиональной направленности кандидата;

3) разработка алгоритма оценки близости реальных и эталонных образцов кандидата с расчетом как обобщенного интегрального показателя, так и уровней развития составляющих каждого показателя.

В качестве начального приближения к эталонной модели кандидата можно предложить следующее ее содержание: перечень основных профессиональных качеств, которые являются значимыми для успешной профессиональной подготовки в вузе; диапазон изменения уровня сформированности качеств, в пределах которого не снижается успешность подготовки; уровень развития каждого качества.

В учебном заведении учебно-воспитательная работа должна быть поставлена таким образом, чтобы активно вовлечь специалистов по компьютерным технологиям в борьбу с киберпреступностью или хотя бы изолировать их от преступной среды. В учебных заведениях, занимающихся подготовкой специалистов по информационным технологиям, целесообразно преподавать «Кодекс компьютерной этики», делая упор на то, что уважение к собственности других лиц в виртуальном мире столь же важно, как в мире физическом. Конечно, есть те, кто совершит преступление независимо от общественного мнения, но влияние авторитета – ценный инструмент против многих правонарушений, совершаемых только из-за ошибочной веры в то, что «все это делают».

В идеале необходимо, чтобы специалисты по информационной безопасности символически давали клятву соблюдать «Кодекс компьютерной этики» (типа «Клятвы Гиппократ», как это принято у выпускников медицинских вузов).

Литература:

1. Психологический словарь / Под. ред. В.В. Давыдова, А.В. Запорожца, Б.Ф. Ломова и др. – М.: Педагогика, 1983.
2. Шиндер Д.Л. Киберпреступность / перевод Т. Тропиной – Crime.vl.ru

УЧЕБНО-МЕТОДИЧЕСКИЕ АСПЕКТЫ СОВЕРШЕНСТВОВАНИЯ ПОДГОТОВКИ СПЕЦИАЛИСТОВ В ОБЛАСТИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Г.Ю. Маклаков

г. Севастополь, Севастопольский национальный технический университет

Проблема совершенствования подготовки специалистов в области информационной безопасности является многоплановой. Успешно решить её возможно только комплексно, на основе системного подхода.

Концепция совершенствования процесса обучения в учебно-методическом направлении предусматривает решение следующих задач: формирование правовой культуры в области информационных технологий, корректировка существующих учебных программ, введение новых учебных дисциплин.

Для отражения современных достижений в области защиты информации необходимо регулярно (не менее 1 раза в год) пересматривать содержание специальных учебных дисциплин. В настоящее время целесообразно дополнить учебные программы подготовки специалистов в области информационной безопасности дисциплинами по нормативно-правовой базе информационных технологий и по стратегии и тактике информационной войны. Главное в этом направлении – усиление разделов подготовки студентов по нормативно-правовой базе в области информационной безопасности. Такой подход предлагается реализовать следующим образом.

На первом курсе для студентов всех специальностей вводится дисциплина «Правовые основы компьютерной деятельности» (название условное). Цель изучения дисциплины: ознакомить студентов с правовыми аспектами эксплуатации вычислительной техники, показать неотвратимость наказания за совершение компьютерных преступлений. В процессе изучения дисциплины студенты должны познакомиться с основными положениями Закона Украины «Об авторском праве и смежных правах», Закона Украины «Об информации» и разделом 16 «Преступления в сфере использования электронно-вычислительных машин (компьютеров), систем и компьютерных сетей» Уголовного Кодекса Украины. Ориентировочный объем дисциплины 8–12 часов. Возможно изучение этой дисциплины путем включения ее разделов в дисциплину «Введение в специальность».

Введение дисциплины «Правовые основы компьютерной деятельности» представляется целесообразным не только для специалистов по информационной безопасности, но и во все учебные заведения, занимающиеся подготовкой специалистов по компьютерным наукам.

В учебные планы подготовки специалистов по защите информации предлагается ввести дисциплину: «Расследование компьютерных инцидентов». Цели дисциплины: ознакомление студентов с используемыми право-

охранительными органами методами предотвращения и пресечения попыток неправомерного доступа к компьютерной информации, составляющей коммерческую (служебную) тайну, а также существующими подходами к расследованию компьютерных преступлений. Основное содержание дисциплины:

Раздел 1. Понятие и классификация компьютерных преступлений (КП). Возможные последствия несанкционированного доступа к критически важной информации. Неизбежность КП как следствие невозможности создания абсолютной защиты. Основные стадии КП (подготовка, развитие, скрытие следов).

Раздел 2. Деятельность хакеров. Социальный состав хакеров. Цели и методы их деятельности. Возможные последствия. Некоторые мифы и суеверия о хакерах. Типичные атаки хакеров. Понятие о социальной инженерии.

Раздел 3. Основные предпосылки для возникновения КП. Принципы формирования службы безопасности. Работа с персоналом. Уязвимости в программном обеспечении. Наличие скрытых уязвимостей в используемых программных средствах вследствие использования пиратского ПО.

Раздел 4. Расследование КП на Украине и за рубежом. Компетентные государственные органы, осуществляющие расследование на Украине. Управление «К» МВД и ФСБ Российской Федерации. Секретная служба Министерства финансов США. ФБР США. Интерпол. Международное взаимодействие в расследовании КП. Доказательственное значение материалов, полученных из-за границы. Типичные ошибки при сборе информации за рубежом.

Раздел 5. Действия в случае возникновения КП. Неотложные действия. Рекомендации NIPС США и их адаптация к условиям Украины. Взаимодействие с государственными и негосударственными органами. Выявление и устранение предпосылок, способствовавших возникновению КП.

Раздел 6. Действия после обращения в государственные органы. Взаимодействие с правоохранительными органами в ходе расследования. Возбуждение уголовного дела и получение статуса потерпевшего. Оценка нанесенного злоумышленником ущерба с учетом морального вреда и упущенной выгоды организации. Сбор доказательств преступной деятельности злоумышленника. Недопустимость применения незаконных методов сбора доказательств. Различие между служебным расследованием в рамках организации и оперативно-розыскной деятельностью.

Раздел 7. Изъятие и исследование компьютерной техники и носителей информации. Правовые основы для изъятия и исследования компьютерной техники. Основные принципы изъятия имущества в ходе расследования уголовного дела (обыск, выемка, осмотр, добровольная выдача) и в административном порядке (осмотр). Методика изъятия компьютерной техники и носителей информации. Обеспечение доказательственного значения изъя-

тых материалов. Методика исследования компьютерной техники.

Компьютерные преступления является одним из способов ведения информационной войны с целью идеологического и психологического влияния на сознание отдельного индивида, социальных групп, разжигания этнической и религиозной вражды. Поэтому при подготовке специалистов в области защиты информации необходимо включать вопросы, связанные методологическими принципами ведения информационных операций. Специалист по защите информации обязан знать стратегию и тактику ведения информационной войны, поражающие факторы информационного оружия и приемы противодействия информационным операциям. Представляется целесообразным в учебные планы подготовки специалистов в области информационной безопасности ввести дисциплину «Информационно-психологическая безопасность информационных технологий». В дисциплине предлагается изучение следующих тем [1, 2]:

Тема 1. Информатизация общества и проблема защиты информации (ретроспективный анализ подходов к формированию множества угроз информации; информационная война: методологические основания; модель и принципы информационной войны; информационное пространство и общественное мнение, как объект воздействия).

Тема 2. Правовые основы информационной безопасности (правовое обеспечение информационной безопасности по защите прав и интересов личности, общества и государства; определение и содержание понятия угрозы информации в современных системах ее обработки; концепция национальной безопасности Украины; нормативно-правовые акты Украины; понятие об информационной безопасности человека, общества, государства).

Тема 3. Защита информации в биологических системах (информационная война как целенаправленное информационное воздействие информационных систем на человека; приемы информационного воздействия; общие принципы защиты информации в биологических системах).

Тема 4. Гомеостаз биосистемы (понятие об иерархической структуре системы управления психофизиологическими функциями человека; методы оценки стабильности управления; модель Парина-Баевского оценки устойчивости управления биосистемой; методы вариационной пульсометрии и реографии; энтропийно-информационные методы; ортогональные латинские квадраты и возможности их использования для выявления факторов информационного воздействия на человека).

Некоторые разделы дисциплины были апробированы в Севастопольском национальном техническом университете при подготовке бакалавров и магистров по специальности «Компьютерные системы и сети». С 2001 года часть разделов включена в дисциплину «Основы вычислительной техники» для аспирантов I года обучения всех специальностей [1].

Литература:

1. Маклаков Г.Ю. Система подготовки специалистов по компьютерной инженерии в области информационной безопасности. // Информационные технологии и безопасность. Материалы научно-практической конференции 9-11 октября 2001. Партенит. Украинский дом экономических и научно-технических знаний. – К., 2001. – С. 43-46.
2. Маклаков Г.Ю. Методологические принципы организации инженерного образования на основе концепции информационной безопасности. // Труды Таганрогского государственного радиотехнического университета. – 2003. – №4. – С. 380-382.

СИСТЕМА АНАЛИЗА ВОЗДЕЙСТВИЯ ИНФОРМАЦИИ НА ПОДСОЗНАНИЕ ЧЕЛОВЕКА В УСЛОВИЯХ ИНФОРМАЦИОННО-ПСИХОЛОГИЧЕСКОГО ПРОТИВОБОРСТВА

С.А. Сидченко¹, К.И. Хударковский¹, В.Л. Петров²

¹ г. Харьков, Харьковский университет Воздушных Сил

² г. Харьков, Объединенный научно-исследовательский институт

Вооруженных Сил

abbot@bk.ru

Постановка проблемы. Настоящий момент времени характеризуется усилением роли информационной безопасности (информационной борьбы), как ключевой составляющей в системе обеспечения национальной безопасности государства. Однако уровень развития информационной безопасности не в полной мере удовлетворяет потребностям общества и государства. Одним из факторов, влияющих на это, является “недостатки розвиненість науково-методичної бази забезпечення інформаційної безпеки, стратегії й тактики ведення інформаційної боротьби” [1].

Информационная борьба представляет собой комплекс мероприятий по воздействию и защите объектов информационной безопасности. Одним из них является информационно-психологическое противоборство. Важными приоритетами в обеспечении информационно-психологической борьбы является “встановлення методологічних засад і механізмів нейтралізації інформаційно-психологічних впливів проти держави” [1]. Эта задача становится актуальной еще и потому, что смежные с Украиной государства – Россия, Турция, Румыния и Польша – динамично развивают структуры для проведения информационно-психологического противоборства [1].

Анализ литературы. Общая характеристика форм и способов ведения информационной борьбы, направлений их развития на современном этапе рассмотрена в работах В.Б. Толубко, А.А. Рося, С.Я. Жука, И.С. Руснака, В.А. Фомина, С.Н. Гриняева и др. Аспектам информационно-психологического противоборства посвящены работы А.А. Рося, И.В. Замаруевой, Ч. Осгуда, А.П. Журавлева, А.А. Феклистова, О.Я. Лазаревой, Р.М. Фрумкина и др.

В работах не рассматривалась система информационно-психологического противоборства в формализованном виде. Методы поиска, анализа, составления и формирования текстов (информации) представлены очень широко. Однако системы, реализующие эти методы, не всегда являются пригодными для ведения информационно-психологического противоборства. Они реализуют методы либо лингвистического, либо фонетического анализа. Например, система «ВААЛ» [2] (<http://www.vaal.ru>), применяемая в России для анализа и формирования текстов с заданными параметрами воздействия, ориентирована на русскоязычные тексты и реализует методы фо-

нетического анализа. Еще одним недостатком большинства систем является отсутствие описания их математической базы и запрещением правительств большинства государств на экспорт данных продуктов в полном объеме.

Цель статьи. Рассмотреть в формализованном виде систему информационно-психологического противоборства и предложить специальное программное обеспечение анализа и воздействия информации на подсознание человека с элементами анализа ее смыслового содержания.

1. Формализованное представление системы информационно-психологического противоборства

При рассмотрении информационно-психологического противоборства необходимо учесть, что «информация – продукт скоропортящийся, дорогостоящий и не всегда съедобный, его надо правильно приготовить и подать к столу». Подходы к обоснованию ценности и старению информации рассмотрены в [3]. Информация является составляющей информационной сферы государства [1].

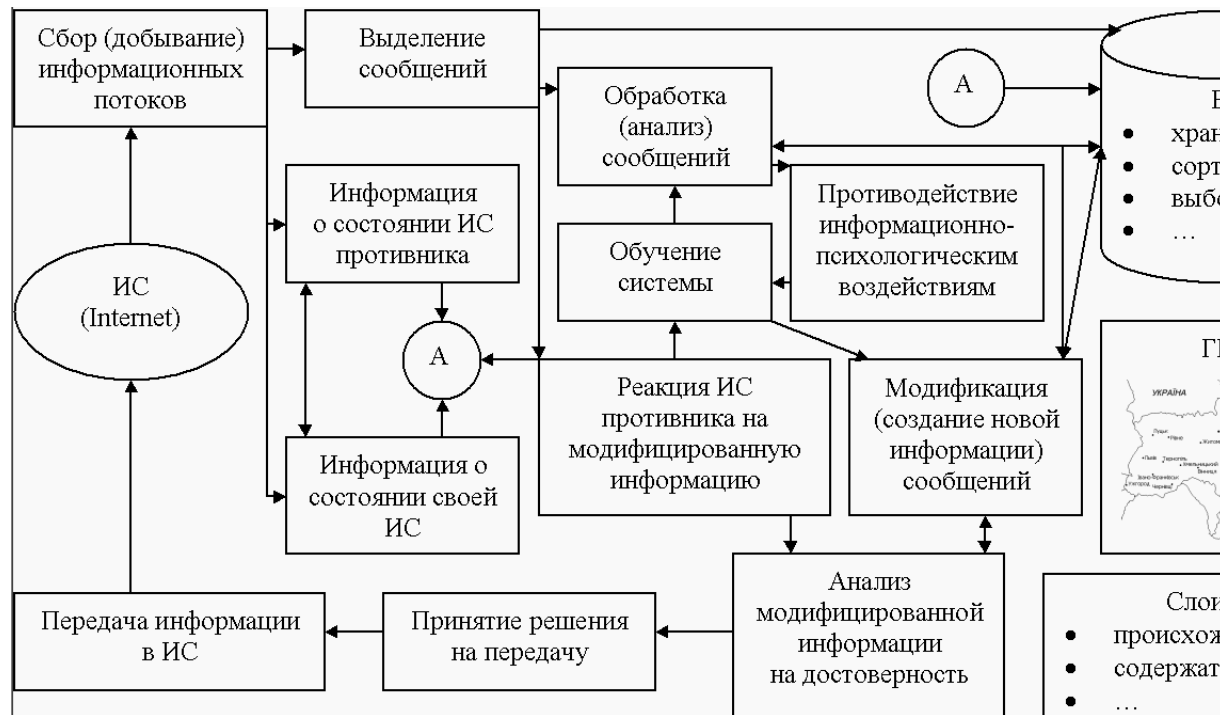
В формализованном виде информационно-психологическое противоборство (ИПП) представляет собой анализ (обработку) информационных потоков, поступающих в информационную сферу государства (в том числе, и информации полученной через систему разведки), противодействие информационно-психологическим воздействиям, составление документов с необходимыми параметрами воздействия, а так же защиту своих информационных потоков (рис. 1).

Рассмотрим две информационные системы (ИС) S_1 и S_2 , осуществляющие информационно-психологическое противоборство (ИПП).

Основной составляющей ИПП является сбор информационных потоков, выделение сообщений (выявления факта передачи сообщения, декодирование, криптоанализ, выявление стеганографических сообщений и т.д.) и противодействие информационно-психологическим воздействиям. Сбор информации и противодействие информационно-психологическим воздействиям должны удовлетворять требованиям оперативности, целенаправленности и достоверности.

Реализация стратегии сбора (добывания, разведки) информации ИС S_1 направлена на осуществление доступа (санкционированного и несанкционированного) к информации I (где $I = I_1 || I_2$; $I_1 = \sum_j I_{1j}$, где j – количество информационных ресурсов ИС S_1 ; $I_2 = \sum_i I_{2i}$, где i – количество информационных ресурсов ИС S_2) в заданном временном интервале времени $\Delta t_{\text{зад раз}}$.

Реализация стратегии противодействия информационно-психологическому воздействию I_2 ($I_2' = \sum_d I_{2d}'$, где d – количество ресурсов информационно-психологического воздействия ИС S_2) в заданном временном интервале времени $\Delta t_{\text{зад прот}}$ с высокой (заданной) вероятностью p_1 результата добывания и противодействия.



Добытое сообщение подвергается обработке (анализу). Основы обработки разведывательной информации представлены в [4]. Обработка информации должна отвечать требованиям своевременности, достоверности и полноты. Результаты обработки и само сообщение сохраняется в базе данных.

Следующей составляющей ИПП является информационно-психологическое воздействие на ИС противника. Оно включает в себя модификацию исходного сообщения (и/или создание нового) с необходимыми параметрами воздействия и передачу информации. Модифицированная информация должна подвергаться проверке на адекватность. При этом воздействие на ИС противника должно осуществляться с помощью информационного оружия конциентального и семантического воздействия [5].

Реализация стратегии информационно-психологического воздействия направлена на осуществление целенаправленного воздействия информацией I_1' ($I_1' = \sum_k I_{1k}'$, где k – количество ложных информационных ресурсов ИС S_1) с заданными показателями воздействия на ИС S_2 в заданном временном интервале времени $\Delta t_{зад\ воzd}$ с высокой (заданной) вероятностью p_2 результата осуществления воздействия.

Кроме наступательной функции ИПП обладает и оборонительной, которая заключается в защите своих информационных ресурсов от воздействия противника. При этом может применяться пассивное и активное (противодействия и превентивного воздействия) информационное оружие защиты.

Реализация стратегии защиты информации направлено на минимизацию осуществления НСД к информации I_1 ИС S_1 в заданном временном интервале времени $\Delta t_{зад}$ с высокой (заданной) вероятностью p_3 результата противодействию НСД.

Информационно-психологическое противоборство должно проводиться при максимально возможном осуществлении способа применения F_n и минимальных затратах C_p .

Формально это можно записать следующим образом:

$$\{S_1\} = \begin{cases} p_1 = \max_{\Delta t_{зад\ раз}} p(\text{access}(S_1 \rightarrow S_2, I)) \cup \max_{\Delta t_{зад\ пром}} p(\text{counter}(S_2 \rightarrow S_1, I_2')); \\ p_2 = \max_{\Delta t_{зад\ воzd}} p(\text{influence}(S_1 \rightarrow S_2, I_1')); \\ p_3 = \max_{\Delta t_{зад}} p(\text{noaccess}(S_2 \rightarrow S_1, I_1)); \\ F_n = \max_{\Delta t_{зад}} \sum_{j=1}^k (b_j \cdot f_j); \\ C_p = \min_{\Delta t_{зад}} \sum_{j=1}^k (a_j \cdot r_j); \end{cases}$$

где $\{S_1\}$ – множество стратегий системы информационно-психологического противоборства S_1 ;

p_1 – вероятность добывания информации (осуществления разведки) и противодействия информационно-психологическим воздействиям;
 p_2 – вероятность совершения информационно-психологического воздействия;
 p_3 – вероятность защиты информации;
 I_1 – множество защищаемых информационных ресурсов ИС S_1 ;
 I_2 – множество разведываемых информационных ресурсов ИС S_2 ;
 I_1' – множество информационных ресурсов воздействия ИС S_1 ;
 I_2' – множество информационных ресурсов воздействия ИС S_2 ;
access – событие, заключающееся в получении доступа к информации;
noaccess – событие, заключающееся в отказе доступа к информации;
counter – событие, заключающееся в информационно-психологическом противодействии;
influence – информационно-психологическое воздействие, как отображение ложной информации I_1' из ИС S_1 в S_2 ;
 F_n – числовое значение суммы приведенных способов ведения ИПП;
 f_j – j -й вид ИПП: $f_j = 1$, если ИПП имеет место, $f_j = 0$ – в противном случае;
 b_j – весовой коэффициент, отражающий существенность j -го способа ИПП;
 $j = 1..k$, k – количество способов ведения ИПП;
 C_p – числовое значение суммы приведенных затрат;
 r_j – значение затрат j -го уровня системы S_1 ;
 a_j – весовой коэффициент, отражающий существенность j -го уровня затрат для системы S_1 .

Вся информация, собранная в базе данных, должна выводиться на единую географическую информационную систему (ГИС) по происхождению, областям знаний (политическая, дипломатическая, научная, экономическая, военная, техническая, административная и т.д.), содержанию и т.д.

Функция обучения системы заключается в анализе реакции ИС противника на представленную ей ложную информацию и уточнение методов анализа и воздействия.

Система ИПП должна удовлетворять требованиям рефлексивного управления, которые заключаются в принятии решений на основе анализа своих возможностей, возможностей противника и знаниях обеих систем о возможностях противоборствующей стороны. При этом управление будет эффективным тогда и только тогда, когда наша ИС будет находиться на один шаг впереди по информированности о состоянии ИС противника.

Любая система хороша только в том случае, когда хорошее программно-аппаратное обеспечение, из которого она состоит, и персонал, который его обслуживает.

2. Специальное программное обеспечение анализа и воздействия информации на подсознание человека

Естественный язык является средством влияния на подсознание чело-

века. Эффективность влияния в большой степени достигается путем формирования необходимого эмоционального отношения к сказанному. Соответствующим управлением объемом, последовательностью и направленностью фонетических и лексических значений могут быть сформированы неявные (скрытые, суггестивные) каналы влияния.

Реализуя обнаружение и анализ соответствующих фонетических значений слов с использованием семантического дифференциала [6, 7 и др.], можно провести анализ текстов и выступлений, определить их направленность (наличие суггестивных аспектов) и осуществить корректирование соответственно заданным характеристикам влияния. В целом, реализация технологии анализа текстов и выступлений позволяет оценивать “степень подготовленности” к эффективному восприятию и скрытую направленность информационно-психологического влияния.

Реализуя данную технологию, нами предлагается специальное программное обеспечение (СПО) для оценки воздействия слов и текстов на подсознание человека с элементами анализа их смыслового содержания.

Основные методы, которые в нем используются:

- 1) методы фонетического анализа на основе формирования семантического дифференциала в применении к звукам языка (как основной язык выбран русский);
- 2) метод лингвистического анализа позитивного и негативного влияния отдельных слов на смысловое значение текста целиком;
- 3) метод цвето-звукового анализа;
- 4) методы нейро-лингвистического программирования для определения нагрузки на основные сенсорные каналы восприятия человека;
- 5) методы контент-анализа и др.

Для эффективного воздействия на большие массы людей важны не только и не столько логичность и аргументированность в употреблении языка, а скорее эмоциональное воздействие на слушателя или читателя. Ибо через формирование эмоционального отношения к сказанному можно быстрее всего добиться его притяжения или неприятия. Эмоциональная окраска слова определяется в большей мере его звучанием и формируется на уровне первой сигнальной системы. Поэтому большее внимание необходимо уделить фонетическому анализу текстов и связь его с лингвистическим (смысловым) анализом структуры документа.

Для оценки эмоционального воздействия можно использовать либо 20 шкал для фонетического анализа текстов, либо 25 шкал для анализа текстов методом семантического дифференциала. По этим шкалам сопоставлены оценки, которые не осознаются людьми, но оказывают довольно сильное воздействие на подсознание людей.

Фонетический анализ текстов проводится на основе звукового анализа сообщения (текста) целиком. Для проведения динамического анализа нами предлагается метод фонетического анализа текста по словам и струк-

турам, что позволяет эффективнее проводить анализ и дает представление о фонетико-смысловой нагрузке отдельных его элементов.

Лингвистический анализ позитивных и негативных слов позволяет оценить смысловую нагрузку текста.

Все виды анализа могут проводиться статически и динамически. Статический анализ дает только конечный результат, в то время как динамический анализ дает представление о картине в целом (изменение воздействия на разных участках информации).

Целенаправленное использование характеристик позволяет преобразовать исходные тексты или составлять новые тексты с аналогичным воздействием.

Анализ и целенаправленное автоматическое редактирование текстов предлагается проводить с помощью специально размеченного словаря Microsoft Word, что позволяет сделать СПО мобильной (и применять, как мобильное наступательное информационное оружие ИПП) и, в дальнейшем, адаптированное к разным языковым группам.

В силу того, что русский и украинский языки относятся к славянской группе, сделано предположение, что качественные оценки для украинского языка остаются без существенных изменений и СПО оптимизировано для работы с русскими и украинскими текстами. Ведутся дополнительные исследования для адаптации специального программного обеспечения для различных языковых групп.

Более подробное рассмотрение методов, предлагаемых для использования в СПО (как известных, так и их синтезов), будет продолжено в дальнейших публикациях.

Выводы. СПО может применяться для решения следующих задач:

- 1) оценивать эмоциональное воздействие отдельных слов на подсознание человека;
- 2) оценивать эмоциональное воздействие фонетической структуры текстов на подсознание человека;
- 3) оценивать уровень агрессивности текстов на основе анализа позитивного и негативного влиянием отдельных слов на смысловое значение текста целиком;
- 4) оценивать звуко-цветовые характеристики текстов;
- 5) задавать характеристики желаемого воздействия и создавать (корректировать) структуры соответствующей направленности;
- 6) настраивать тексты на лексически определенные социальные и профессиональные группы людей.
- 7) оценивать нагрузку на сенсорные каналы восприятия путем анализа используемой лексики.

Данное СПО необходимо при ведении ИПП и миротворческих операций, для создания благоприятной информационной обстановки.

Литература:

1. Толубко В.Б., Жук С.Я., Косевцов В.О. Концептуальні основи інформаційної безпеки України // Наука і оборона. – 2004. – №2. – С. 19-25.
2. Конкурентная разведка в Интернет / В.В. Дудихин, О.В. Дудихина. – М.: ООО «Издательство АСТ»: Издательство «НТ Пресс», 2004. – 229 с.
3. Петров В.Л., Сидченко С.А., Антонов Д.В. Подходы к обоснованию показателей стоимости системы защиты. // Сборник научных трудов ХВУ. – Харків: ХВУ. – № 7(37). – 2001. – С. 89-92.
4. Кудрявцев А.М. Обработка разведывательной информации. – Л.: ВАС, 1989. – 332 с.
5. Шолохов С.Н., Сидченко С.А. Информационное оружие – новый класс вооружения для дезорганизации автоматизированных систем управления войсками и оружием при проведении информационных наступательных операций. // Сборник научных трудов ХВУ. – Х: ХВУ. – № 1(39). – 2002. – С. 10-14.
6. Осгуд Ч. Приложение методики семантического дифференциала к исследованиям по эстетике и смежным проблемам // Семиотика и искусствоведение. – М., 1972.
7. Журавлев А.П. Фонетическое значение – Л.: ЛГУ, 1974.

МЕТОДИ РОЗРОБКИ МОДЕЛІ БЕЗПЕКИ АВТОМАТИЗОВАНОЇ СИСТЕМИ ДОКУМЕНТООБИГУ

Н.І. Соколянська

м. Київ, Інститут проблем реєстрації інформації НАН України
polevaya@yahoo.com

Вступ

Питання інформаційної безпеки є однією зі складових спільної задачі впровадження інформаційних технологій автоматизації документообігу у державних і приватних установах, на виробництві й у бізнесі, тобто в тих структурах, для яких проблема забезпечення конфіденційності і цілісності документованої інформації існувала завжди.

До застосування комп'ютерних технологій у будь-якій організації, для якої безпека документації мала певне значення, був установлений порядок роботи з інформацією (наприклад, система роботи із секретними документами), що регламентує документопотоки всередині організації й обмін інформацією з зовнішнім світом. Цей порядок містив, по-перше, схему документопотоків усередині організації, і, по-друге, набір правил керування цими потоками.

Досвід експлуатації існуючих систем документообігу показує, що проблема безпеки ще не вирішена, а запропоновані виробниками різних систем засоби захисту сильно розрізняються як по розв'язуваних задачах і використовуваних методах, так і за досягнутими результатами. Визначимо, що повинна являти собою захищена автоматизована система документообігу, які задачі необхідно вирішити, щоб її побудувати, які методи найбільш ефективні для їхнього вирішення.

Постановка проблеми

Під захищеною автоматизованою системою документообігу (СДО) пропонується розуміти систему, що володіє наступними трьома властивостями [1]:

- забезпечує автоматизацію процесу документообігу, включаючи всі аспекти цього процесу, пов'язані з забезпеченням цілісності і безпеки оброблюваних документів;

- успішно й ефективно протистоїть загрозам безпеки;

- відповідає вимогам і критеріям стандартів інформаційної безпеки.

Запропонований підхід до визначення поняття “захищена СДО” відрізняється від існуючих у першу чергу тим, що розглядає проблему безпеки СДО як проблему на перетині проблем автоматизації документообігу і загальної безпеки. Такий погляд на проблему дозволяє розглядати процеси проектування і розробки захищених автоматизованих систем документообігу одночасно в контексті вирішення задач автоматизації і забезпечення безпеки, що дає можливість застосовувати методи і технології, розроблені як в

одній, так і в іншій області.

Запропоноване визначення дозволяє сформулювати три задачі, які необхідно і досить вирішити, для того щоб створити захищену СДО, а саме:

– у ході автоматизації процесу обробки документів врахувати всі аспекти цього процесу, пов'язані з забезпеченням безпеки і цілісності оброблюваної інформації;

– забезпечити захист від загроз безпеки, що діють у середовищі експлуатації даної СДО;

– реалізувати вимоги відповідних стандартів інформаційної безпеки.

Якщо основна мета впровадження інформаційних технологій – автоматизувати процес обробки інформації, то, відповідно, часткова задача автоматизації процесів обробки документів – забезпечити адекватну реалізацію СДО схеми документопотоків і правил управління ними, що існувала до застосування автоматизованих засобів обробки. Тобто автоматизована СДО повинна, по-перше, реалізовувати потоки, що існували до її застосування, і не створювати нових і, по-друге, забезпечити можливість керування документопотоками відповідно до заданого набору правил.

Для вирішення цієї задачі необхідно виконати наступні етапи:

1. Визначення механізму, що адекватно виражає задану схему документопотоків і правила керування ними.

2. Побудова моделі безпеки, що відображає заданий порядок обробки документів, і формальний доказ її безпеки.

3. Реалізація СДО відповідно до запропонованої моделі.

4. Доказ адекватності реалізованих в автоматизованій СДО обробки документопотоків, правил контролю доступу до документопотоків і правил керування ними.

У ході здійснення даної послідовності дій розробки захищеної СДО виникають наступні проблеми:

1. Схема документопотоків і правила керування ними можуть бути неповні чи застосовувати важко формалізовані концепції. Це ускладнює створення моделі безпеки чи доказ її безпеки.

2. Схема документопотоків носить статичний характер і визначає поділ потоків тільки в штатному режимі роботи. Здійснення додавання і видалення компонентів визначається розроблювачами СДО у робочому порядку, що приводить до втрати доказовості і неадекватної реалізації моделі безпеки.

3. Поява нових об'єктів в схемі СДО, що не відповідають сутностям документообігу (наприклад, службові файли, ресурси) не відтворюється в схемі документопотоків і не враховуються правилами управління цими потоками. Однак, цілком очевидно, що контроль доступу до подібних об'єктів має ключове значення для безпеки всієї СДО в цілому.

4. У ході реалізації моделі безпеки можуть з'явитися неконтрольовані документопотоки, оскільки в комп'ютерній системі користувачі не можуть маніпулювати інформацією безпосередньо і використовують програмні за-

соби, що можуть незалежно від їхньої волі створювати небажані неконтрольовані потоки (атака за допомогою “троянського коня”).

Вирішення проблеми

Виходячи з визначеного переліку задач створення захищеної СДО, рішення яких необхідно для досягнення поставленої мети, спробуємо запропонувати методи забезпечення безпеки автоматизованих СДО з погляду автоматизації процесів документообігу і способів протидії загрозам безпеки.

Захищена СДО повинна складатися з наступних взаємозалежних блоків:

- блок технічних (програмних) засобів захисту електронного документообігу;
- блок технічних засобів захисту електронного документообігу, зв'язаних з нейтралізацією побічних електромагнітних випромінювань і наведень;
- блок методів захисту електронного документообігу, пов'язаних з людським фактором і рішенням кадрових питань;
- блок організаційних методів захисту електронного документообігу.

Блок засобів захисту електронного документообігу повинен бути багатокордонний, наприклад:

1 кордон – системи захисту інформації, передбачені програмним забезпеченням, на якій працює комп'ютерна мережа (засобу захисту Windows, Office);

2 кордон – системи захисту інформації, убудовані в саму систему електронного документообігу;

3 кордон – системи захисту інформації, додатково встановлені в комп'ютерній мережі на сервері і на робочих місцях користувача.

Блок засобів захисту електронного документообігу повинен передбачати:

- криптографічний захист комп'ютерної інформації на магнітних носіях (твердих дисках, дискетах, Zip, CD-ROM і так далі), у тому числі створення захищених «цифрових сейфів» користувачів на сервері;
- захист інформації (у тому числі криптографічними методами) при пересиланні по корпоративній мережі;
- захист інформації (у тому числі криптографічними методами) при передачі по електронній пошті;
- захист комп'ютерної інформації від несанкціонованого доступу;
- захист комп'ютерної мережі при роботі в глобальній інформаційній мережі Інтернет;
- мандатний принцип доступу користувачів до інформаційних ресурсів електронного діловодства.

Багаторівневий захист забезпечує розмежування доступу суб'єктів з різними правами доступу до об'єктів різних рівнів конфіденційності. Він ха-

рактерний для СДО зі складною територіально-розподіленою структурою.

Можна виділити три основних аспекти захисту в СДО з розмежуванням повноважень.

1. Розмежування доступу до документів усередині СДО.
2. Захист документів від несанкціонованого доступу ззовні СДО.
3. Захист документопотоків при передачі її по мережах зв'язку.

Аспект розмежування доступу є специфічним для багаторівневих систем, два інших – спільні проблеми всіх СДО. Розрізняють три основних методи побудови СДО з використанням багаторівневого ієрархічного доступу [2]:

- адміністративний розподіл ключів;
- криптографічний розподіл ключів;
- ієрархічне шифрування.

Враховуючи вищеперераховані аспекти безпеки з погляду автоматизації процесів документообігу і способів протидії загрозам безпеки удається досить ефективно будувати багаторівневі СДО, що задовольняють вимогам створення захищених СДО.

Висновки

Тільки вдале розв'язання проблем, що лежать на стику теорії інформаційної безпеки й автоматизації процесів документообігу, дозволить розроблювачам захищених СДО забезпечити адекватну автоматизацію існуючого в реальних системах порядку документообігу та коректно застосовувати на практиці формальні моделі безпеки.

Література:

1. Калінін М.О. Структура і застосування мови опису політик безпеки. // Проблеми інформаційної безпеки. Комп'ютерні системи. – 2002. – №1.
2. Min-Shiang Hwang. A Cryptographic Key Assignment Scheme in a Hierarchy for Access Control. // Mathl. Comput. Modelling. – Vol. 26. – №2 – P. 27-31.

Зміст

<i>Н.М. Аушева, А.Л. Гурін.</i> Геоінформаційні системи у сучасній підготовці спеціалістів з комп'ютерних наук	3
<i>А.В. Бабич.</i> Обзор CASE-средств для построения диаграмм UML	8
<i>А.Н. Бакал.</i> Использование брандмауэров для защиты Linux-серверов	12
<i>Н.В. Баловсяк.</i> Організаційно-педагогічні умови формування інформаційної компетентності	16
<i>В.Ю. Баранов.</i> Автоматизований контроль знань в системі управління навчально-пізнавальною діяльністю студентів	21
<i>Л.І. Білоусова, Т.В. Белявцева, О.Г. Колгатін, Л.С. Колгатіна.</i> Навчальні дослідження при вивченні методів обчислювальної математики	26
<i>Л.Р. Билялова.</i> Разработка компьютерных обучающих программ по ОС Linux и ее приложениям	31
<i>О.В. Бондар.</i> Застосування електронного підручника у навчальному процесі	37
<i>О.М. Боско, Н.В. Гринь.</i> Дисципліна “Прикладне програмне забезпечення” – ланка між теорією та практикою	41
<i>А.В. Владзимирский, Е.Т. Дорохова.</i> Разработка системы преподавания основ телемедицины в медицинском вузе	43
<i>Л.П. Воронець.</i> Міжпредметні зв'язки у структуруванні курсу “Інформатика і комп'ютерна техніка”	49
<i>О.О. Гагарін, В.І. Гайдаржи.</i> Підготовка розробника інформаційних систем в НТУУ “КПІ”	53
<i>О.В. Гладченко.</i> Походження терміну “інформація” та розуміння його сутності в рамках вузівського курсу інформатики	61
<i>Я.М. Глинський, В.А. Ряжська.</i> Підбір і послідовність вивчення тем у курсі інформатики	66
<i>П.М. Григоруک.</i> Роль методів багатомірного статистичного аналізу у фаховій підготовці економіста і методика їх практичного опанування	74
<i>О.М. Дреєв, З.Ю. Філер.</i> Частотно-амплітудний аналіз майже періодичних процесів та його застосування для прогнозів	78
<i>А.П. Забарна.</i> Педагогічні основи використання активних методів у навчанні інформатики	83
<i>З.Г. Зуїкова.</i> Применение численных методов к задачам общего курса физики со сложной зависимостью между параметрами	89
<i>О.Ф. Клименко, Н.Р. Головка.</i> Методика викладання дисципліни “Інформатика та комп'ютерна техніка” з урахуванням вимог Болонської декларації	91
<i>О.В. Клочко.</i> Становлення творчої професійної діяльності студентів вищих аграрних навчальних закладів у процесі вивчення дисциплін інформаційного циклу	96

<i>І.А. Колтаков, В.Г. Рябцев.</i> Метод проектирования алгоритмов для диагностирования микросхем памяти	103
<i>В.Н. Комличенко, С.А. Поттосина, Н.А. Кириенко.</i> О потенциале информационных технологий и математическом образовании специалиста по экономической информатике	108
<i>С.В. Кондратенко.</i> Зміст курсу «Нові інформаційні технології» для студентів факультету іноземних мов	114
<i>В.В. Кравченко.</i> Можливості Macromedia Flash в аспекті сучасних концепцій побудови інтерфейсу користувача.....	118
<i>Г.Т. Кравчук, Т.В. Шевчук.</i> Досвід впровадження інформаційних технологій у навчальний процес Львівського банківського інституту.....	124
<i>Ю.М. Красюк.</i> Особливості формування системи навчальних задач з інформатики	130
<i>І.А. Кривель, А.Н. Моргун.</i> Общие подходы к обучению пользованию программными средствами пакета Microsoft Office.....	133
<i>А.П. Кудін, Ю.А. Свистун.</i> Комп'ютерні системи контролю знань у дистанційному навчанні.....	138
<i>А.П. Кудін, Л.В. Харченко.</i> Дистанційний курс з інформатики для слабочуючих	142
<i>Г.И. Кулик.</i> Использование современных компьютерных технологий при подготовке и чтении различных разделов курса «Информатика»	146
<i>Е.А. Лавров, Н.Л. Барченко.</i> Нейронная сеть как элемент системы управления обучением	148
<i>О.П. Ліннік.</i> Модульна технологія вивчення методів обчислень	153
<i>Р.В. Ліхачов.</i> ‘Hello World’ як дидактичний атавізм	156
<i>І.М. Лукаш.</i> Організація самоконтролю при навчанні інформатики....	162
<i>І.В. Лупан, В.В. Копотій.</i> Навчальні проекти для майбутніх вчителів	165
<i>Т.Л. Мазурок, О.А. Логіна.</i> Інформаційна технологія підтримки прийняття рішень як складова формування інформаційної культури	170
<i>И.Е. Мазурок, Т.Л. Мазурок.</i> Использование мобильных коммуникационных устройств в образовательных целях	175
<i>Г.Г. Маклакова, Г.Ю. Маклаков, Е.А. Кожжаев.</i> Мультимедийный аппаратно-программный комплекс для отображения учебной информации на основе бытовых телевизионных приемников	180
<i>В.Л. Малорян.</i> Образная система алгоритмического языка	185
<i>Ю.С. Матвієнко.</i> Переваги ХМЛ при розробці освітнього порталу....	188
<i>А.Ю. Мельников.</i> Тестирующий программный комплекс «MATEST»	192
<i>О.І. Миронова.</i> Самостійна робота студентів у процесі вивчення курсу “Програмне забезпечення ЕОМ”	196
<i>А.А. Мясищев, В.М. Полозова.</i> Блокировка спама средствами транспортного агента sendmail под управлением UNIX	201

<i>А.А. Мясищев, В.М. Полозова.</i> Установка почтового сервера под UNIX FreeBSD с использованием транспортного агента – программы sendmail.....	209
<i>І.В. Настенко.</i> Використання інноваційних технологій на перших уроках з об'єктно-орієнтованого програмування в класах з поглибленим вивченням інформатики.....	216
<i>Е.А. Панина, Л.В. Гусева.</i> Особенности проведения модульных контрольных работ в форме теста.....	220
<i>Ю.М. Пилипенко, Л.П. Голубев.</i> Вивчення MS Excel у курсі «Інформатика».....	223
<i>В.О. Потієнко.</i> Формування змісту задач як засобу розвитку учнів під час вивчення графічних редакторів.....	229
<i>О.А. Синявская, Б.А. Железко.</i> Изучение технологии создания экспертных систем с помощью оболочки «XpertRule Knowledge Builder».....	234
<i>О.Г. Смілянець.</i> Методичні умови підготовки студентів економічних спеціальностей до розв'язання творчих фахових задач на заняттях з інформатики.....	240
<i>Ю.А. Супрунова.</i> Использование программы HyperCam для создания обучающих видео-роликов.....	245
<i>О.В. Тумашиова, О.М. Коссак.</i> Особливості методики викладання курсу “Чисельні методи” для студентів енергетичних спеціальностей.....	248
<i>О.Н. Туравинина.</i> Методическая разработка курса «Компьютерное делопроизводство» для специальности «Информатика и экономика».....	252
<i>І.Є. Фільо.</i> Удосконалення форм і методів контролю знань з інформатики у ВНЗ.....	258
<i>Н.А. Хараджян.</i> Использование пакета символьной математики Maxima при изучении линейных списков в курсе «Структуры данных и алгоритмы их обработки».....	262
<i>С.Н. Чаплыгина.</i> Изложение раздела «Компьютерные сети» дисциплины «Информатика и компьютерная техника».....	265
<i>С.І. Шаповалова, О.Є. Круш.</i> Особливості викладання дисципліни “Об'єктно-орієнтоване програмування”.....	268
<i>С.В. Шаров.</i> Використання електронних підручників в навчальному процесі.....	273
<i>О.В. Шматко.</i> Використання термінальних інформаційних систем на заняттях з інформатики.....	276
<i>І.А. Яковлева.</i> Применение мультимедиа технологий при преподавании информатики.....	278
Захист інформації та інформаційна безпека.....	280
<i>Х.О. Засадна, Р.Б. Попович.</i> Програмна реалізація симетричних та асиметричних криптосистем та її використання у навчальному процесі.....	281

<i>Е.А. Кононова, Н.В. Моисеенко.</i> Практикум по основам защиты информации	286
<i>Г.Ю. Маклаков.</i> Информационный терроризм: понятие, терминология, противодействие	290
<i>Г.Ю. Маклаков.</i> Роль человеческого фактора при подготовке специалистов в области информационной безопасности в высших учебных заведениях	297
<i>Г.Ю. Маклаков.</i> Учебно-методические аспекты совершенствования подготовки специалистов в области информационной безопасности	299
<i>С.А. Сидченко, К.И. Хударковский, В.Л. Петров.</i> Система анализа воздействия информации на подсознание человека в условиях информационно-психологического противоборства	303
<i>Н.І. Соколянська.</i> Методи розробки моделі безпеки автоматизованої системи документообігу	311

Наукове видання

**Теорія та методика навчання
математики, фізики, інформатики**

Випуск V

В 3-х томах

Том 3

Підп. до друку 06.03.2004
Папір офсетний №1
Ум. друк. арк. 16,81

Формат 80×84 1/16
Зам. №3-0603
Тираж 300 прим.

Жовтнева друкарня
50014, м. Кривий Ріг-14, вул. Електрична, 5
Тел. (0564) 664381

E-mail: cc@kpi.dp.ua