

Міністерство освіти та науки України  
Національна металургійна академія України

Теорія та методика  
навчання математики,  
фізики, інформатики

*Збірник наукових праць  
Випуск 3*

Том 3

Кривий Ріг  
Видавничий відділ НМетАУ  
2003

**Теорія та методика навчання математики, фізики, інформатики:** Збірник наукових праць. Випуск 3: В 3-х томах. – Кривий Ріг: Видавничий відділ НМетАУ, 2003. – Т. 3: Теорія та методика навчання інформатики. – 381 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 та застосування засобів нових інформаційних технологій 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

*О.В. Сергеев*, 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 5-7490-0093-3

## СТВОРЕННЯ ІНТЕРАКТИВНИХ НАВЧАЛЬНИХ ПОСІБНИКІВ ЗА ДОПОМОГОЮ WEBEXE

О.В. Бабич

м. Полтава, Полтавський державний педагогічний університет  
ім. В.Г. Короленка  
taurus@kardzs.com

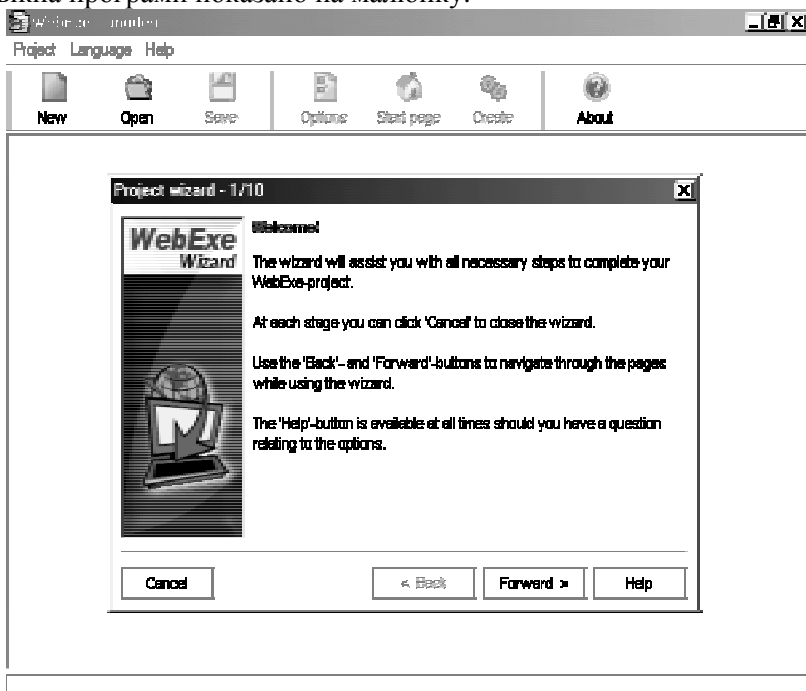
Інтерактивні посібники під час вивчення фундаментальних дисциплін у вищій школі застосовуються вже досить довгий час. У найпростішому випадку вони являють собою просто конспект лекцій з даної дисципліни, що має меню для вибору лекції з конкретної теми, яке одночасно виконує функції змісту. Більш складні системи можуть також містити секцію контролю, тобто тест з даної дисципліни.

Під час створення таких посібників певну увагу слід приділити питанню вибору інструментальних засобів. Досить часто для цього використовуються середовища візуальної розробки додатків типу Delphi. В такому випадку викладач витрачає час більше на програмування і створення інтерфейсу, ніж на написання самого контенту посібника. Застосування MS Power Point також не є ефективним, оскільки переглянути готовий продукт можна лише на машині, на якій встановлено MS Office або MS Power Point Viewer. Досить непоганим вибором при створенні електронного посібника є Demo Shield, оскільки він має в своєму складі Demo Shield Player для перегляду створюваного продукту, але цей вельми, безперечно, потужний пакет має певні особливості, і починаючи роботу з ним деякий час слід витратити на освоєння базових прийомів роботи з ним.

На думку автора, ідеальним інструментальним засобом для розробки інтерактивного навчального посібника є зв'язка WebEXE з будь-яким HTML-редактором. На користь цієї думки свідчать такі факти. По-перше, HTML – мова дуже проста, а для неспеціалістів її знання навіть не вимагається – ви можете просто зберегти текст своїх лекцій, набраний у MS Word в форматі HTML. Зміст – меню з гіперпосиланнями – можна також створити в тому ж MS Word. По-друге, використавши WebEXE, ви зможете захистити свою інтелектуальну власність, заборонивши

копіювання, друк документу, збереження ілюстрацій або використавши захист паролем чи обмеження строку користування програмою. І по-третє, готові продукти, створені в WebEXE, використовують MSIE у якості додатку-сервера, а він, як відомо, є складовою всіх операційних систем від Microsoft, тож встановлений практично на кожній машині.

Механізм роботи WebEXE дуже простий: програма просто об'єднує кілька взаємопов'язаних Web-сторінок в один архів, а потім генерує оболонку для перегляду вмісту цього архіву з досить сучасним інтерфейсом. При створенні проекту в WebEXE можна налагодити як зовнішній вигляд продукту, так і опції захисту від несанкціонованого копіювання. Також можна вибрати, ще які типи файлів, крім .htm, слід включати в проект. Вигляд вікна програми показано на малюнку.



Такий підхід вже було використано викладачами Харківського національного університету радіоелектроніки [1]. Автор цих рядків використав WebEXE для створення посібників з дисциплін “Мережі ЕОМ” та “Бази даних та інструментальні системи”.

Також WebEXE можна з успіхом використовувати для створення оболонок автозапуску для CD з матеріалами конференцій, семінарів і т.ін.

Звичайно, можливості HTML обмежені, тож певно деяких специфічних задумів втілити в життя не вдасться, але для потреб навчального процесу у вищій школі, а саме для створення інтерактивних навчальних посібників, дана програма є чудовим вибором. До речі, скопіювати пробну версію WebEXE можна з [www.aw-soft.com](http://www.aw-soft.com).

#### Література

1. Методы и средства компьютерных информационных технологий. Лабораторный практикум. Кафедра программного обеспечения ЭВМ Харьковского национального университета радиоелектроники. ЕАЛ – 2002.
2. WebExe Online Manual. © Andreas Wulf, 2002.

## К ВОПРОСУ О КОНЦЕПЦИЯХ СОВРЕМЕННОЙ ИНФОРМАТИКИ

А.Н. Бакал

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

В наше время повсеместного распространения персональных компьютеров (ПК) человеческие знания о природе информации приобретают общекультурную ценность. Этим объясняется интерес исследователей и практиков всего мира к относительно молодой и быстро развивающейся научной дисциплине – информатике. На сегодняшний день информатика выделилась в фундаментальную науку об информационно-логических моделях, и она не может быть сведена к другим наукам, даже к математике, очень близкой по изучаемым вопросам. Объектом изучения информатики являются структура информации и методы ее обработки. Появились различия между информатикой как наукой, с собственной предметной областью, и информационными технологиями.

Технология, в отличие от науки, предполагает созидательную деятельность для получения определенных результатов человеком, который использует данную технологию. Можно выделить некоторые информационные технологии, базирующиеся на применении компьютера, без которых человек не может обойтись в современном мире. Это универсальные технологии, основанные на работе с текстовыми и табличными процессорами, СУБД, графическими редакторами, а также мультимедиа и системы телекоммуникаций.

Помимо этого, существует множество специализированных компьютерных технологий (например, компьютерная верстка, бухгалтерский учет, проектирование в архитектуре, и многое другое). Еще один важный раздел компьютерных технологий – это компьютерные технологии обучения, представляющие значительные возможности для повышения эффективности занятий, придания им интереса и динамичности. Очевидно, что базовые знания этих технологий должна давать школа. По мере распро-

странения компьютерной техники вся совокупность взаимодействующей с ней людей все более четко разделяются на три большие (относительно) группы:

- системные программисты;
- прикладные программисты, разрабатывающие пакеты прикладных программ для решения больших классов задач из самых различных отраслей;
- широкий круг пользователей.

Для представителей первой группы написание программ на тех. или иных языках или непосредственно в машинных кодах является обязательной составной частью их профессиональной деятельности. Более того, продуктом деятельности системных программистов являются средства общения с ПК, предназначенные для широкого круга пользователей. Общение пользователей с ПК протекает в рамках специализированных, заранее разработанных программных систем, использующих привычную для них профессиональную и бытовую терминологию.

Таким образом, для значительного большинства людей, использующих ПК в своей работе дома или в офисе, знание конкретных языков и систем программирования не является обязательным.

В то же время, с точки зрения пользователя, существуют «глубокие» и «неглубокие» уровни применения компьютера. Если человек использует компьютер для набора какого-либо текста, то компьютеру ничего неизвестно о содержательной стороне решаемой проблемы и используется он только как «умная» пишущая машинка с проверкой орфографии. Если же работающий за компьютером специалист сообщает ему сведения о содержании задачи, о ее понятиях и объектах (разработка нового бизнес-плана предприятия), то при таком глубоком применении компьютера возможна значительно более высокая эффективность его использования. А для этого необходимо нечто большее, чем «общий язык с компьютером», – нужна общая мысль, понятия, отдельный стиль мышления и навыки умственных действий. В настоящее время такой набор необходим практически каждому специалисту в любой из отраслей науки и техники. Рассмотрим кратко наиболее существенные из них.

Навыки планирования структуры действий, необходимых

для достижения заданной цели при помощи фиксированного набора средств часто называют умением алгоритмически мыслить, хотя термин «структура действий» несколько шире классического определения алгоритма.

Деятельность пользователя по описанию алгоритма решаемой задачи состоит в том, что, видя перед собой конечную цель – результат, он конструирует программу (в широком смысле слова), план действий, представляющий последовательность отдельных более или менее стандартных операций. То, что в обиходе называют иногда программистскими способностями, определяется в первую очередь умением представить сложное действие в виде определенной последовательности простых операций. При этом пользователь должен спланировать не только сами действия, но и используемые при этом информационно-технические ресурсы.

Человек, использующий накопленные машинными системами банки информации, даже будучи освобожденным, от описания структуры данных, должен, тем не менее, отдавать себе отчет о классах используемых величин и их взаимосвязях. При этом весьма важен навык представить себе структуры информационных объектов в формализованном виде.

Значение этого навыка в настоящее время повышается в связи с распространением реляционных СУБД, информационных и экспертных систем, САД-систем, основу которых составляют информационные модели. При работе с такими моделями необходимо отразить в них все существенные для решения поставленной задачи свойства объектов и их взаимодействия, иначе решение, предложенное машиной, может сослужить пользователю плохую службу.

Решение задачи может быть эффективным только в том случае, когда правильно определен объем данных, необходимых для ее решения, и правильно организован их поиск. При этом вырабатываются навыки формирования поисковых механизмов и пользования ими. По мере роста емкости и снижения стоимости запоминающих устройств ПК становится реально перспектива передачи машинам всего информационного фонда, накопленного человечеством. Это дает возможность пользователю оперативно получить практически любые необходимые ему сведения. Тем



более важным оказывается умение правильно определить, какие именно сведения необходимы, и по каким признакам организовать их поиск.

Системный и прикладной программист должен приобрести навык общения с машиной или системой на различных уровнях в зависимости от их «обученности». С системами низшего уровня (имеющие незначительное математическое обеспечение) общение возможно только при помощи указаний о «микродействиях» – машинных операциях. В системах с высоким уровнем математического обеспечения программист должен строить свою программу из крупных блоков – операторов языка высокого уровня и готовых программ, в том числе и разноязыковых. Пользователь далекий от подобного глобального подхода к решаемым задачам, тем не менее, обязан, если он хочет рационально общаться с машиной, уметь превратить в процедуры, часто используемые конструкции для того, чтобы в последствии применять их как элементарные предписания.

Для рационального использования вычислительной техники пользователь должен иметь привычку своевременно обращаться к ПК при решении задач из любой области. Если такая привычка не сформирована, нельзя гарантировать, что даже опытный пользователь, хорошо владеющий перечисленными выше навыками, догадается обратиться в нужный момент к ПК, если такая задача ему прямо не поставлена. В результате может возникнуть такая ситуация, когда пользователь выполняет вычисления карандашом на бумаге сидя рядом с терминалом ПК, позволяющей сделать эти вычисления гораздо быстрее и точнее.

Также пользователь должен уметь работать клавиатурой и другими периферийными устройствами, такими как принтер, сканнер, модем, факс и т.д.

Информатика как образовательная дисциплина быстро развивается. Если 5–6 лет назад базовый курс информатики состоял из изучения основ алгоритмизации и программирования, основ устройства и применения вычислительной техники, то сегодня целью курса информатики в школе является повышение эффективности применения человеком компьютера как инструмента. Компьютерная грамотность определяется не только умением программировать, а, в основном, умением использовать готовые

программные продукты, рассчитанные на пользовательский уровень. Эта тенденция появилась благодаря широкому рассмотрению «мягких» продуктов, ориентированных на неподготовленных пользователей. Разработка таких программно-информационных средств является весьма дорогостоящим делом в силу его высокой наукоемкости и необходимости совместной работы высококвалифицированных специалистов: психологов, компьютерных дизайнеров, программистов. Однако она окупает себя благодаря тому, что доступ к компьютеру сегодня может получить практически каждый человек даже без специальной подготовки.

# **ПРОБЛЕМА ВВЕДЕНИЯ В СОДЕРЖАНИЕ КУРСА “ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧ НА ЭВМ” (ДЛЯ ТЕХНИЧЕСКИХ ВУЗОВ) МАТЕРИАЛА О МЕТОДЕ ГРАНИЧНЫХ ЭЛЕМЕНТОВ**

Д.Е. Бобылев

г. Кривой Рог, Криворожский экономический институт  
Киевского национального экономического университета

В последнее время среди специалистов, которые занимаются расчетами напряженно-деформируемых состояний твердых тел, становится все более популярным метод граничных элементов (МГЭ). Такая тенденция вызвана стремительным развитием вычислительной техники. Долгий период МГЭ был не задействован в научных расчетах по нескольким причинам:

- была недостаточно проработана теоретическая база данного метода;

- фундаментальные положения МГЭ трудно усвоить без хорошей математической подготовки;

- в научной литературе по МГЭ предмет обычно описывается в контексте частной научной или инженерной проблемы, поэтому не всегда очевидно, что метод, разработанный для решения задач одного типа, применим также и к иным задачам;

- построение расчетной схемы для каждой задачи требует значительных усилий.

Если же первая причина уже устранена, то последние – остаются открытой проблемой. Во многом решение этих проблем зависит от содержания дисциплин математического цикла в технических вузах. А оно остается стабильным вот уже на протяжении тридцати лет. Например, в курсе “Численные методы решения задач на ЭВМ”, который изучается в Криворожском техническом университете, рассматриваются только методы конечных разностей и конечных элементов [3].

В классических учебниках по численным методам отсутствие МГЭ можно объяснить еще тем, что до появления ЭВМ решение задач данным методом было связано с большими вычислительными трудностями [4]. Современные же учебники и учебные пособия по численным методам для технических вузов не

включают в себя материал по МГЭ, скорее всего из-за боязни авторов того, что большинство студентов технических специальностей не в состоянии понять данный метод.

Но данная причина не должна лишать возможности будущих инженеров ознакомиться с МГЭ. Поскольку этот метод является мощным вычислительным инструментом нахождения характеристик напряженно-деформируемого состояния твердых тел. МГЭ превосходит по возможностям методы конечных разностей и конечных элементов, благодаря двум его решающим преимуществам – сокращение на единицу геометрической размерности задачи (и соответствующему снижению затрат на подготовку информации, память, время и стоимость вычислений) и легкость исследования бесконечных областей. Кроме того, МГЭ позволяет естественным образом отразить достаточно сложные условия взаимодействия на соприкасающихся границах тел.

Специфичность данного метода накладывает некоторые условия на учебный материал. Главное из них – прагматический характер учебного пособия, т.е. материал должен содержать уже готовые модули МГЭ. Это условие позволяет студентам сразу применять методы граничных элементов на практике, прямо перенося приведенные в приложениях программы на свои ЭВМ, комбинируя вычислительные модули для создания новых программ и даже проводя сложные изменения в модулях для повышения точности и (или) для решения нелинейных, динамических и т.п. задач. Таким образом, будущие исследователи и инженеры получают благоприятную возможность легко и быстро включиться в процесс использования и развития МГЭ.

Вторая условие – чрезвычайная простота и доступность изложения материала, ее высокий методический уровень, т.е. ясность и умелая подача материала, “модульная” структура, акцент на физической стороне обсуждаемых вопросов, обход математических трудностей, подробность необходимых выкладок, представление всех главных формул в развернутом, годном для прямого включения в программы виде.

Третье условие – проявление внимания к сложным контактным задачам, связанным с учетом не только упругого взаимодействия на соприкасающихся границах тел, но и с необратимым деформированием на контактах. Метод граничных элементов в

варианте разрывных смещений по самой своей природе идеально приспособлен к решению соответствующих проблем.

Четвертое условие – изложение материала должно быть построено на решении задач, которые решаются теми специалистами, для которых читается курс численных методов.

Проанализировав литературу, в которой излагается МГЭ [1, 2] можно сделать вывод, что наиболее всего этим условиям удовлетворяет вторая. Это не только делает книгу ценной для исследователей и инженеров, но и позволяет ей служить полезным пособием в преподавании.

Но к введению материала о МГЭ в содержание курса “Численные методы решения задач на ЭВМ” должно носить системный характер. Поскольку в курсе высшей математики для технических вузов в недостаточной мере излагаются темы “Комплексный анализ” и “Интегральные уравнения”, необходимо также пересмотреть содержание курса высшей математики.

Необходимо также предусмотреть несколько лабораторных работ на прямой и непрямой методы граничных элементов. В них уже должны быть основные блоки для реализации данных методов, поскольку создание программы МГЭ с “нуля” требует очень много времени.

#### Литература

1. Бенерджи П., Баттерфилд Р. Метод граничных элементов в прикладных науках: Пер. с англ. – М.: Мир, 1984. – 496 с.
2. Крауч С., Старфилд А. Методы граничных элементов в механике твердого тела: Пер. с англ. – М.: Мир, 1987. – 328 с.
3. Методические указания к лабораторным работам по курсам “Численные методы решения задач на ЭВМ”, “Математические методы и модели в расчетах на ЭВМ”, “Вычислительная техника и программирование” / Сост. Н.Х. Сайтгареев. – Кривой Рог: КТУ, 2002. – 114 с.
4. Хемминг Р.В. Численные методы для научных работников и инженеров: Пер. с англ. – М.: Наука, 1972. – 400 с.

# ОРГАНІЗАЦІЯ НАВЧАННЯ ОПЕРАТИВНОГО ПЕРСОНАЛУ НА ОСНОВІ МЕТОДИКИ АЛГОРИТМІВ ДІЙ

М.А. Богданевич

м. Київ, Київський національний торговельно-економічний університет  
bna@ksteu.kiev.ua

Ефективне функціонування багатьох галузей господарства залежить від надійності роботи складних людино-машинних систем. Такі системи містять складні технологічні об'єкти (енергоблоки електростанцій, диспетчерські системи й ін.), у контур керування якими включені люди. Діяльність оперативного персоналу має ряд особливостей: великі потоки інформації, обмежений час на прийняття рішення, велика відповідальність за результати діяльності та інше.

У діяльності оператора по керуванню технічним обладнанням можна умовно виділити п'ять основних етапів:

- прийом інформації;
- оцінка ситуації;
- пошук причин відхилення (діагностика);
- планування дій;
- реалізація плану.

Прийом інформації проходить в процесі нагляду за режимом роботи обладнання. При цьому оператор періодично приймає рішення про нормальний розвиток процесу, чи про появу відхилень від норми. У випадку відхилень від нормальної роботи від повинен провести діагностику, тобто виявити причину відхилення. Після визначення даної причини необхідно розробити план операцій по її усуненню та приступити до реалізації цього плану.

Ефективність і успішність діяльності по керуванню людиною-оператора залежать від наступних чинників:

- рівня знань, що дозволяють приймати осмислені рішення при керуванні обладнанням;
- рівня навичок оперативного мислення, тобто навичок побудови причинно-слідчих зв'язків між показаннями приладів і ходом технологічного процесу; навичок взаємодії з моторними й

інформаційними елементами, включаючи різні системи відображення інформації, засоби обчислювальної техніки, різні системи автоматичного регулювання і дистанційного керування.

Всі ці чинники вимагають реалізації комплексного підходу для забезпечення якісної підготовки персоналу. Вони є основою при побудові структури навчання і тренажу оперативного персоналу. Ігнорування або недооцінка якогось чинника діяльності оператора і відповідно тієї складової частини в структурі навчання і тренажу приводить до утруднень у процесі керування технологічним об'єктом. Тому в системі підготовки персоналу можемо виділити ряд підсистем розвитку комплексу знань, навичок і умінь.

Підсистема навчання і контролю знань призначена для початкового придбання знань і поточного контролю рівня знань. Можна виділити наступні типи знань: 1) знання загальнотеоретичного характеру, зв'язані з фізичними процесами роботи об'єкта управління, конструкцією устаткування, режимами роботи, правилами експлуатації і 2) знання елементів діяльності по спостереженню, оцінці ситуації, плануванню дій.

Використання знань у процесі практичної роботи досягається за рахунок тренування. При цьому в оператора повинні бути розвиті навички спостереження і виявлення відхилень, упізнання відхилень, ухвалення рішення про метод ліквідації відхилення і побудові плану дій.

Одним із найважливіших етапів оперативної діяльності є реалізація плану (алгоритму) дій для досягнення поставленої мети. Алгоритми дій являють собою об'єднання операцій і умов, що регламентують послідовність здійснення операцій. В них задається початкова ситуація і ставиться кінцева мета побудови плану. Методика алгоритмів дій ефективна при переведенні устаткування з одного стану в інший при нормальній роботі, а також при усуненні аварійних ситуацій. Крім того дану методику можна застосовувати при вивченні тих дисциплін, які вимагають знання будь-якої послідовності дій, наприклад, знання технологічних карт виготовлення певної продукції, послідовність операцій виконання певних процесів та інше. Користувачу необхідно послідовно сформулювати з деякої, випадковим чином отриманої, початкової послідовності правильну послідовність дій. При цьому,

як правило, задається певний час для виконання завдання та обмеження на кількість помилок.

Моделі функціонування обладнання створюються відповідними спеціалістами по роботі з даним обладнанням в режимі діалогу з комп'ютером, який дозволяє організувати введення даних в відповідну базу та забезпечує при необхідності редагування даних.

Важливе значення має спосіб наочного описування моделі функціонування алгоритму дій. Він повинний відбивати причинно-слідчі зв'язки між операціями, бути простим і зручним як для поняття тим, хто навчається, так і для безпосереднього введення в комп'ютер текстового файлу плану дій.

Відомі різні способи описання алгоритму функціонування плану дій. Серед них можна виділити описання у виді текстів, описання за допомогою блок-схем, описання з використанням спеціально розробленої методики.

Текстові описання годяться для простих планів. У випадку складної логіки функціонування плану дій робота з текстом великого розміру стає важкою. Крім того, словесний опис не дозволяє реалізацію виконання даного плану дій на комп'ютері.

Останнім часом плани дій описувалися за допомогою блок-схем. Для цього використовуються стандартні елементи такі як початок, закінчення, виконання операції, перевірка умови.

Аналіз технологічного процесу при роботі з методикою планування дій дозволив виділити наступні типи елементарних операцій:

- виконання дії;
- перевірка умови;
- виконання однієї з групи альтернативних дій;
- виконання всіх дій у групі в довільному порядку (рівнозначні дії).

Важливу роль при заповненні бази алгоритмів дій відіграє підсистема автора. Вона повинна організувати перетворення описання плану дій в комп'ютерну базу даних, а також забезпечити простоту внесення змін до алгоритму дій. В відповідності зі структурою інформації роботу автора по введенню інформації на основі блок-схеми можна розбити на наступні етапи:

- введення номерів та текстового опису дій та умов;



- введення зв'язків між операціями;
- формування груп альтернативних дій;
- формування груп рівнозначних дій;
- допустимий час формування алгоритму дій;
- допустима кількість помилок.

Всі операції в алгоритмі повинні бути пронумеровані. При цьому необхідно притримуватись певних обмежень, наприклад, першою операцією в алгоритмі завжди повинна бути операція під номером 1. Крім того операція перевірки умови повинна мати всього два виходи. Вже на етапі введення інформації необхідно забезпечити перевірку на її правильність.

За бажанням автора кожній дії може призначена певна вага, яка буде враховуватись при допущенні помилки при виборі даної дії. Тобто, чим більша вага помилки, тим вона “грубіша”. За замовчуванням всі помилки рівнозначні при аналізі результату формування алгоритму дій.

Бувають складні алгоритми дій, які містять велику кількість операцій, що відразу не зможуть бути виведені на екран. В цьому випадку у автора є можливість розбити план на певну кількість фрагментів, кожен з яких буде автоматично завантажуватись в режимі контролю після закінчення роботи з попереднім.

При роботі студента з системою планування дій йому на екран дисплею виводяться дії, які розташовані в випадковому порядку та пронумеровані в послідовному порядку. Займають дані операції ліву частину екрану з назвою “Початкова послідовність”. Дії, які виводяться студенту, можуть бути двох типів: 1) дії, метою яких є включення, вимкнення, переключення та т.п. і 2) дії, метою яких є перевірка умов (показань).

Права частина екрану під назвою “Правильна послідовність” послідовно формується за рахунок перенесення дій з лівої у випадку правильної роботи студента. Якщо вибирається при роботі дія перевірки умови, то в цьому випадку видається випадковий результат умови (так, ні) і перевірка побудови плану йде в подальшому відповідно до даного результату. При неправильній відповіді двічі підряд на вибір правильної операції система робить підказку правильної операції. Кожна помилка фіксується та виводиться на екран.

Вихід з режиму планування дій проходить в наступних ви-

падках :

- алгоритм дій сформовано, отримання оцінки;
- час, відведений на формування, вийшов; незадовільна оцінка;
- кількість помилок перевищила допустиму; незадовільна помилка.

По завершенні роботи з планом на дисплей виводиться статистична інформація, яка дозволяє визначити хід побудови алгоритму, допущені при побудові помилки, затрачений час та виводить загальну отриману оцінку.

Оцінка може бути, як по основній системі оцінок (відмінно, добре, задовільно, незадовільно) так і так по так званій бальній системі. Бальна система оцінок вимагає від автора додаткової інформації, по внеску кожної правильної або неправильної дії в загальну оцінку під час формування плану. В подальшому студенту на початку роботи видається певна кількість балів, яка збільшується при кожній правильній відповіді, та зменшується при неправильній і відповідно до кількості затраченого часу.

Для надання інформації керівнику навчання передбачається наявність статистичної бази даних. До неї включено: дані про студентів, які повинні були працювати з підсистемою алгоритмів дій; перелік вивчених алгоритмів; дати роботи; статистична інформація про хід роботи.

## **ПРОБЛЕМИ ВИВЧЕННЯ ДИСЦИПЛІНИ “ІНФОРМАТИКА ТА ОБЧИСЛЮВАЛЬНА ТЕХНІКА”**

М.А. Бондаренко

м. Харків, Українська інженерно-педагогічна академія

Наша сучасність характерна стрімким розвитком комп'ютерної техніки, отже величезним потоком програмного забезпечення. Комп'ютерні технології проникають у всі сфери людської діяльності. Зараз неможливо уявити висококваліфікованого спеціаліста без навичок роботи з комп'ютером. Сучасна людина повинна мати знання з інформаційних технологій та повинна бути дійсно підготовленою як користувач комп'ютерної техніки.

Життя примушує суміщати два різних напрямки :

- вивчати основи інформатики, фундаментом якої є алгоритмізація та елементи логіки;
- вдосконалювати напрямок користувача, пов'язаний з вивченням комп'ютерних технологій.

Аналізуючи знання студентів після закінчення навчання можна заключити, що ігнорувати будь-яким із цих двох напрямків було б недоцільним.

Відомо, що сучасна література з інформатики бере напрямок на підготовку “чистих” користувачів. Зрозуміло що зараз для розв'язання більшості задач існує готове програмне забезпечення. Але цей прагматичний підхід до вивчення зовнішньої атрибутики комп'ютера не виправдовує себе.

Такий підхід не завжди дозволяє розв'язувати поставлені життям задачі.

Це питання пов'язане з відсутністю звичайних елементарних знань та умінь основам наук, логіки та алгоритмізації мислення, уміння абстрагуватися та відверто мислити.

Тому обидва напрямки при вивченні інформатики та комп'ютерних технологій повинні йти паралельно, збагачуючи та доповнюючи один одного.

Сучасному користувачеві недостатньо вміти використовувати програмне забезпечення, він повинен опиратися на знання основних принципів роботи комп'ютера. Алгоритмічне та логіч-

не мислення допомагає у свою чергу максимально використовувати можливості конкретного комп'ютера.

Слід також відмітити, що еволюція технічних засобів персональних комп'ютерів (ПК) привела до витіснення старої операційної системи (ОС) MS-DOS значно більш потужнішими системами WINDOWS, програмування для яких є більш складним за програмування для MS-DOS. Розробники систем програмування, і перш за все такі корпорації як Microsoft і Borland вже в 1991 році, тобто одразу після появи Windows 3.1, випускають свою продукцію. Borland випускає Turbo Pascal for Windows, а в 1992 р. – удосконалену версію цієї системи програмування Borland Pascal with Objects 7.0. Ці перші спеціалізовані інструменти потребували знання Windows та були складними при їх вивченні. В 1993 р. Microsoft випускає перше візуальне середовище програмування Visual Basic. Програмування для Windows стало більш простим, ніж програмування для MS DOS. У відповідь на це Borland в 1995 році випустила першу версію об'єктно-орієнтованої системи програмування в Delphi. А з інтервалом в один рік – ще такі версії: 2, 3, 4, 5, 6. Не секрет, що система Turbo Pascal є кращою системою для MS DOS. Delphi продовжила версію Паскаль-орієнтованих середовищ програмування, та є найбільш зручним інструментом для Windows-програмування.

Практично всі сучасні мови програмування (в тому числі C++ і Visual Basic) підтримують методикау об'єктно-орієнтованого програмування (ООП), яка дозволяє різко скоротити терміни розробки програм та підвищити їх якість. Таким чином, процедурне програмування, основою якого є алгоритм, процедура обробки даних, відійшла на задній план. Об'єктно-орієнтоване програмування – це методика розробки програм, в основі якої лежать поняття об'єкту, як деякої структури, описуючої об'єкт реального всесвіту, його поведінки. Задача, розв'язана з використанням методики ООП, описується в термінах об'єктів та операцій над ними, а програма в такому разі являє собою набір об'єктів та зв'язків між ними.

Немає сумніву, що практично всі вищі навчальні заклади перейшли до вивчення та застосування системи об'єктно-орієнтованого програмування. До таких ВНЗ відноситься і Українська інженерно-педагогічна академія. Студенти першого

курсу деяких спеціальностей енергетичного та електромеханічного факультетів активно працюють в системі об'єктно-орієнтованого програмування Delphi 7. Це дає змогу студентам йти в ногу з сучасними середовищами програмування, працювати з комп'ютерами типу Pentium та готуватись кваліфіковано виконувати курсові та дипломні проекти. Середовище Delphi 7 – це складний механізм, що забезпечує високоефективну роботу програміста. Візуально вона реалізується декількома одночасно розкритими на екрані дисплею вікнами. Вікна можуть переміщатись по екрану, частково чи повністю перекриваючи одне одного, кожне вікно несе на собі деяку функціональність, тобто призначене для вирішення певних задач.

Основним інструментом Delphi є мова Object Pascal, на якій ведуться розробки проектів. Адже система Delphi 7 – це в першу чергу система програмування і без знання базової мови, яка лежить в її основі, вивчити та зуміти перетворити в дійсність реалізовані в системі технології неможливо.

Сучасний програміст – не письменник або вчений, а кваліфікований робітник. Проїшов той час, коли місяцями доводилось вивчати пристрої та принципи її віртуальних таблиць об'єктного Паскалю, забиратися до надер початкових текстів оригінально створених компонентів. Звичайно невеликому відсотку розробників необхідні глибокі знання (це в основному системні адміністратори та спеціалісти по захисту корпоративної інформації), та все ж від більшості програмістів сьогодні потребується, перш за все, вміння писати прикладні програми максимально швидко та без помилок. При цьому чим менше в проекті буде використано маловідомих можливостей Windows і Delphi, тим краще – бо з текстами програм доведеться розбиратися (та покращувати їх) другим програмістам різної кваліфікації. В таких умовах для керівника проекту на перший план з'являється вміння організувати злагоджену роботу колективу, а для рядових програмістів – здібності вірно розуміти, що від них потребується (але це не так просто, як з початку здається), та виконувати роботу вчасно.

Комерційні компанії досить жорстко потребують від своїх працівників отримувати прибуток. Вони створюють хороші умови праці та платять велику заробітну плату, але робочий день в таких компаніях розписано похвилинно, тому часу на вивчення

нових компонентів, або принципів функціонування нових технологій неможливо знайти. Ось чому вся світова індустрія середовищ розробки сучасних програм йде в напрямку максимального спрощення процесу розробки програм, переводячи його на візуальний рівень. Це дає змогу програмісту сконцентрувати свої зусилля на логіці задачі, що розв'язується. Тому система Delphi 7 являє собою середовище, близьке до ідеального. Комфортне місце розробника поєднується з безліччю технологій, інтегрованих в середовище.

Чотири роки майбутні інженери-педагоги енергетичного та електромеханічного факультетів вивчають систему ООП Delphi 7. Досвід ще невеликий, але з впевненістю можна сказати – вони знаходяться на вірному шляху. Майбутнім фахівцям непотрібно буде тратити час на вивчення систем об'єктно-орієнтованого програмування, їм зараз потрібно сумлінно виконувати лабораторні роботи, які ведуться на комп'ютерах типу Pentium, вивчати Delphi 7 та використовувати цю систему програмування при виконанні курсового та дипломного проектування. Студенти розуміють, що в наш час стрімко розвивається комп'ютерна техніка та програмне забезпечення. Комп'ютерні технології проникають до всіх сфер людської діяльності. Зараз досить трудно уявити собі висококваліфікованого фахівця, не володіючого сучасними системами об'єктно-орієнтованого програмування. Це дуже не просто, бо потребує нової техніки, яку академія має в недостатній кількості, підручників, які треба створити, бо крім підручників, переведених з англійської мови, практично ніяких немає. Але всі ці питання добре розуміють в УПА, бо знають на якій техніці та яким новим системам об'єктно-орієнтованого програмування треба навчати.

# ОСОБЛИВОСТІ ВИВЧЕННЯ ПРИКЛАДНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ У ВИЩОМУ УЧБОВОМУ ЗАКЛАДІ

О.М. Боско

м. Кривий Ріг, Інститут ділового адміністрування

Економічні реформи, що провадяться в Україні, пов'язані з ринковими відносинами. В умовах ринку для ефективного управління підприємствами та установами велике значення має інформація, яка повинна бути оперативною та об'єктивною. Використання засобів сучасної електронної обчислювальної техніки, комп'ютерних мереж та нових інформаційних технологій, дає реальну можливість забезпечити фахівців усіх рівнів такою інформацією, яка їм потрібна для виконання своїх функціональних обов'язків.

Автоматичне і автоматизоване забезпечення об'єктивно необхідною інформацією фахівців звільняє їх від рутинної роботи, пов'язаної зі збиранням, обробкою та елементарним аналізом цієї інформації, і дає змогу займатися творчою працею, розробляти нові, продуктивніші методи управління й т. ін. [1]. Насамперед, зростає оперативність і якість роботи в цілому за рахунок своєчасно отриманої інформації, підвищується продуктивність праці, зменшується ймовірність помилок, покращується якість управління.

Але, на жаль, технічне забезпечення на більшості підприємств та установ України не відповідає загальносвітовому рівню в галузі електронної обчислювальної та інформаційної техніки. В деяких регіонах матеріально-технічна база відстає навіть від загальноукраїнського рівня. Особливий клопіт викликає ситуація, що склалася в загальноосвітніх середніх школах та професійно-технічних училищах. Адже, майбутні студенти вищих учбових закладів не мають можливості на належному рівні підготуватись до сприйняття інформації більш специфічного, професійного характеру. Вирішення цих питань досить актуальне в сучасних умовах.

Тому мета дисципліни “Прикладне програмне забезпечення” – надати майбутнім фахівцям основні теоретичні відомості та

практичні навички і вміння щодо раціонального використання програмних продуктів, які є найпоширенішими на даному етапі технічного забезпечення в конкретному регіоні і в Україні в цілому, а також дати основний напрямок для розвитку розуміння, як саме можна швидко навчитися працювати у нових програмах. Не секрет, що після закінчення вищих учбових закладів молоді фахівці не в змозі продуктивно використовувати комп'ютери. На жаль, картина, коли “користувачі ПК” з гордовитим виглядом заповнюють на екрані розроблені форми лише набиваючи цифри з клавіатури, а основні маніпуляції з розрахунками проводять на калькуляторі, є типовим [2].

Дисципліна “Прикладне програмне забезпечення” має бути тією провідною ниткою, що з'єднає майбутніх фахівців з реальними технічного та інформаційного забезпечення сьогодення. Викладання можна розбити на три етапи (рис. 1), де зображена структурна схема курсу прикладного програмного забезпечення.

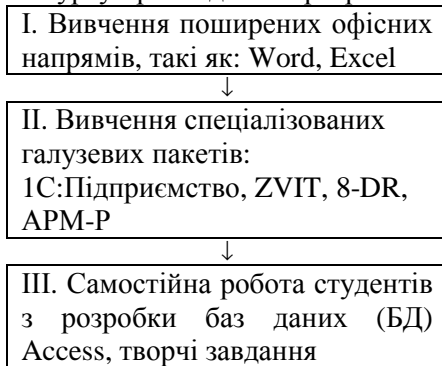


Рис. 1.

На першому етапі на конкретних практичних матеріалах розкрити методи оперативного забезпечення необхідною інформацією певної категорії працівників. На цьому етапі раціональніше використовувати найбільш поширені офісні програми Microsoft Excel та Microsoft Word, оскільки вивчення їх поширилось у розділах дисципліни “Інформатика”. Лабораторні роботи повинні бути підібрані так, щоб вони відповідали професійній специфіці факультету, де вивчається ця дисципліна. Для їх розробки можна порекомендувати застосовувати поширені в даній галузі періодичні видання, адже тоді вони будуть відповідати



сучасним професійним вимогам.

На другому етапі необхідно ознайомити з тим програмним продуктом, який є найбільш поширеним в цієї галузі виробництва на ринку України. Доцільно розробити основну глобальну задачу, на основі якої можна продемонструвати як поетапне її рішення розвивається наочно при застосуванні програми. При можливості деякі етапи розвитку рішення бажано розглянути в альтернативних програмних продуктах. Наприклад, 1С: Підприємство, X-DOOR, БЭСТ та інше.

На третьому етапі вивчення дисципліни необхідно продемонструвати самостійний творчий підхід до організації інформаційних систем на автоматизованому робочому місці (АРМ). Насамперед, це розробка баз даних (теоретичні методи побудови програмного продукту, загальні вимоги до програмних виробів) [3]. В Інституті ділового адміністрування практикується підготовка індивідуальних та творчих завдань за допомогою пакету Access. Крім того, практикується спільне проведення та підготовка занять з викладачами, які ведуть професійно-орієнтовані дисципліни.

На думку автора, впровадження даної системи вивчення ППЗ у вищих навчальних закладах дозволить покращити якість підготовки фахівців у потрібному напрямку, дозволить їм використовувати набуті знання в курсах професійно-орієнтованих дисциплін, курсовому та дипломному проектуванні.

Практичні навички роботи з сучасним прикладним програмним забезпеченням, отримані студентом у вищому навчальному закладі нададуть молодому фахівцю безумовну перевагу при влаштуванні на роботу та у подальшій виробничій діяльності.

#### Література

1. Твердохліб М.Г. Інформаційне забезпечення менеджменту: Навч. посібник. – К.: КНЕУ, 2000. – 208с.
2. Пикуза В. EXCEL: возможности для бухгалтера. – ж. Бизнес/пр. Бухгалтерия № 46. – 1999
3. Молодцова О.П. Прикладне програмне забезпечення / Навчально-методичний посібник для самостійного вивчення дисципліни. – Київ, 2000. – 263с.

## ПРОГРАМУВАННЯ В X WINDOW МОВОЮ ПАСКАЛЬ

В.К. Буряк, Є.М. Ракова

м. Кривий Ріг, Криворізький державний педагогічний університет

X Window (чи просто X) – це система для створення графічного користувальницького інтерфейсу на комп'ютерах, що працюють під керуванням UNIX-подібних операційних систем. X була створена в Массачусетському Технологічному Інституті (США) в 1984 р., і за свою майже 20-річну історію добре зарекомендувала себе у системі освіти. На сьогодні уже випущено версію 11.6 (X Window System Version 11 Release 6 чи X11R6).

Особливістю системи є те, що вона підтримує роботу як на окремії ЕОМ, так і в мережі. Це означає, що програма, що працює на одному комп'ютері, може за допомогою X Window спілкуватися з користувачем, що сидить за іншою машиною. Система забезпечує виведення графічної інформації на екран машини, сприймає сигнали від зовнішніх пристроїв, таких як клавіатура і миша, і передає їх програмам. Зауважимо, що пристрій виведення може мати кілька екранів, і X забезпечує малювання на кожному з них. Усе це: екран (чи екрани), а також пристрій введення (клавіатура чи миша) називаються в термінах X Window *дисплей*.

X дозволяє користувачу спілкуватися з багатьма програмами одночасно. Щоб виведення з них не змішувався, система створює на екрані дисплея віртуальні підекрани – *вікна*. Кожен додаток, як правило, малює лише у своєму чи вікні вікнах. X надає набір засобів для створення вікон, їхнього переміщення по екрану і зміни їхніх розмірів.

Як правило, програми мають набір конфігураційних параметрів – *ресурсів*. Це може бути колір вікна, тип шрифту, яким малюється текст, і багато чого іншого. Система стандартизує спосіб задання ресурсів додатків і містить ряд процедур для роботи з ними. Ця сукупність функцій називається *менеджером ресурсів* (X resource manager чи скорочено Xrm). «Сховище» параметрів програми називається *базою даних ресурсів*.

Особливістю X Window є те, що вона організує спілкування між самими програмами і між програмами та зовнішнім середо-

вищем шляхом розсилання подій. *Подія* – це одиниця інформації, що ідентифікує дії, що відбуваються в системі чи зміні, і утримуюча додаткові зведення про їх.

Система X Window представляє сукупність програм і бібліотек. Серцем її є окремих UNIX-процес, що існує на комп'ютері, до якого приєднаний дисплей. Саме *сервер* знає особливості конкретної апаратури, знає, що треба почати, щоб зафарбувати піксель на екрані, намалювати лінію чи інший графічний об'єкт. Він також уміє сприймати сигнали, що надходять від клавіатури і миші.

Сервер спілкується з програмами-клієнтами, посылаючи чи приймаючи від них порції (пакети) даних. Якщо сервер і клієнт працюють на різних машинах, то дані посылаються по мережі, якщо ж комп'ютер один, то для передачі даних використовується внутрішній канал. Наприклад, якщо сервер виявляє, що натиснуто кнопку миші, то він готує відповідний пакет і посилає його тому клієнту, у чийм вікні знаходиться курсор миші. І навпаки, якщо програмі треба що-небудь вивести на екран дисплея, то вона створює необхідний пакет даних і посилає його серверу.

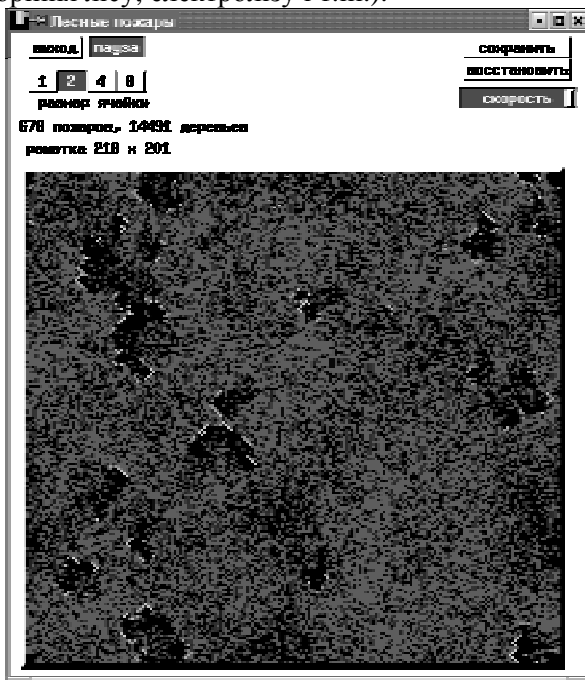
Склад пакетів і їхня послідовність визначаються спеціальним протоколом. Але щоб програмувати для X, зовсім не обов'язково знати деталі реалізації сервера і протоколу обміну. Система надає бібліотеку процедур, за допомогою яких програми здійснюють доступ до послуг X на високому рівні. Так для того, щоб вивести на екран точку, досить викликати процедуру `XDrawPoint()`, передавши їй відповідні параметри. Остання виконує всю чорнову роботу по підготовці і передаванню пакетів даних серверу. Згадана бібліотека називається *Xlib*. Вона міститься у файлі `l11.a (libl11.so)`, що, як правило, знаходиться в каталозі `/usr/X11R6/lib`. Прототипи функцій бібліотеки, використовувани нею структури даних, типи та інше визначається у файлах-заголовках з директорії `/usr/include/X11`.

X Window є основою, на якій побудовано сучасні графічні інтерфейси користувача – KDE, GNOME та інші. Бібліотека *Xlib* має інтерфейс до багатьох мов програмування – C, C++, CLOS, Tcl/Tk, Fortran тощо, проте реалізація інтерфейсу з мовою Паскаль, реалізована у компіляторі *FreePascal*, є неповною: у відповідних модулях (`x`, `xlib`, `xutil`) відсутні реалізації багатьох широ-

ковживаних функцій, що не давало можливості ефективно використовувати можливості X Window.

В зв'язку з цим нами була виконана робота по створенню модуля x11, що містить процедури та функції для взаємодії з віконним менеджером, регіонами, палітрами, менеджером ресурсів тощо. Застосування цього модуля в процесі експериментального навчання побудові інтерфейсу користувача в операційній системі Linux показало, що учні отримують можливість створювати інтерфейсні елементи графічних додатків (див. рис.) простими засобами, без вивчення об'єктного діалекту Паскалю та складних об'єктно-орієнтованих бібліотек візуальних компонентів (на кшталт VCL), що створює навіть для слабких учнів ситуацію успіху.

В процесі експерименту нами було створено методичні рекомендації з програмування в X Window мовою Паскаль та низку демонстрацій можливостей бібліотеки для побудови інтерфейсу користувача з простими імітаційними моделями (розмивання берегу, горіння лісу, електролізу і т.ін.).



## МІСЦЕ НОВІТНІХ ТЕХНОЛОГІЙ В СИСТЕМІ ПОСТІЙНОГО ІНТЕГРОВАНОГО ТРЕНІНГУ МАЙБУТНІХ СПЕЦІАЛІСТІВ

І.П. Вакалюк, В.М. Якимчук, І.Ю. Ванджура,  
О.В. Ткач-Мотуляк, Н.І. Скробач  
м. Івано-Франківськ, Івано-Франківська державна медична ака-  
демія  
vakal@phantom.pu.if.ua

Останнім часом в Україні широко впроваджуються методи оцінки знань студентів, ефективність яких підтверджена багатолітнім досвідом педагогічних колективів провідних навчальних закладів світу і упродовж десятка років використовується при сертифікації дипломів.

Одним з вже традиційних методів є тестовий контроль, який покладений в основу медичного ліцензійного іспиту “Крок-2-лікувальна справа”. Цей іспит базується на сучасній технології стандартизованого контролю рівня знань. Формування готовності випускника до складання ліцензійного іспиту є важливим завданням навчального процесу упродовж всього терміну навчання. Базуючись на тестових ситуаційних завданнях, які детально моделюють клінічні ситуації, діагностичні та лікувальні дії, лікувальний вплив немедикаментозних і медикаментозних засобів терапії, іспит може бути ефективно проведений лише за умов його сприйняття як випускником, так і педагогом, починаючи з фундаментальних дисциплін. Цьому сприяє проведення першого етапу ліцензійних іспитів, а саме “Крок-1”.

В той же час, при викладанні внутрішніх захворювань, наймасивнішого і найвагомішого розділу медицини, постає цілий ряд питань, що спрямовані на необхідність адаптації студентів до наступної задачі тестового іспиту. Більшість з них може бути вирішена шляхом послідовного, комплексного впровадження комп'ютеризації навчального процесу на кафедрах терапії.

Важливими принципами змісту комп'ютеризації навчання на кафедрах терапії є:

- створення послідовних навчальних, навчально-контролюючих і контролюючих програм;

- послідовність і єдність у викладанні матеріалу з врахуванням сучасних класифікаційних і діагностичних підходів до трактування кожної нозології;
- своєчасність адаптації створених програм до нових рекомендацій експертних груп та наукових форумів стосовно стандартів діагностики та лікування;
- розробка методів ефективного поєднання комп'ютеризації навчання та безпосередньої роботи студентів “біля ліжка хворого” (створення і використання фантомних класів, тематичних комп'ютерних ігор стосовно лікарської тактики);
- підготовка і видання відповідної методичної літератури;
- широке залучення студентів до елементів віртуалізації навчання з використанням лазерних носіїв та навчальних серверів мережі “Інтернет” (при цьому важливим постає знання англійської мови).

Реалізація вказаних принципів можлива при створенні мобільної матеріально-технічної бази навчального процесу із своєчасним її оновленням.

Важливим аспектом є методична освіченість викладацького складу стосовно підготовки та використання тестових завдань. Залучення викладачів до написання тестів, їх фахової експертизи (міжкафедральної і міжвузівської), та послідовне використання тестового контролю упродовж викладання дисципліни є одним з важливих принципів підвищення ефективності навчання.

Використання вказаних методичних підходів до забезпечення комп'ютеризації у викладання терапії, як і інших дисциплін, сприятиме оптимізації навчального процесу, повноцінному освоєнню студентами теоретичних знань, практичних навиків та вмій, що дасть можливість підготувати сучасного досвідченого спеціаліста.

## ЕЛЕКТРОННІ ПІДРУЧНИКИ У ВИВЧЕННІ КОМП'ЮТЕРНИХ НАУК

Л.В. Васильєва

м. Краматорськ, Донбаська державна машинобудівна академія  
apmath@dgma.donetsk.ua

Важливим моментом вивчення комп'ютерних наук є безперервність, можливість самостійної роботи на ПЕОМ. Для забезпечення цього процесу студентам пропонуються учбово-практичні посібники, що відрізняються від звичайних видань чіткою структурованістю і наочністю.

Вважається важливим питання використання електронних підручників. При роботі на сучасних комп'ютерах такі підручники можна «читати» паралельно з виконанням власного завдання. Вони містять тренувальні завдання, що сприяють більш міцному закріпленню вивченого матеріалу. Цей підручник має деякі переваги в порівнянні зі своїм друкованим аналогом. Одна з них – можливість посилань на пройдений матеріал. Електронний підручник дає можливість чітко і без зайвих відступів викласти основну ідею досліджуваної теми, додаткові ж пояснення можуть бути викликані, при необхідності, за допомогою системи гіперпосилань і використовуватися як для поглиблення знань, так і для додаткового вивчення недостатньо освоєних тем.

На кафедрі прикладної математики ДДМА при вивченні дисципліни «Чисельні методи при розв'язанні інженерних задач» (на базі пакета MathCad) використовується блокова структура електронного підручника. Матеріал підручника був підготовлений на основі досвіду викладання дисципліни студентам інженерних спеціальностей.

Основний – інформаційний – блок містить викладений у стислій формі навчальний матеріал (основні поняття, визначення, таблиці, графіки, що охоплюють усі розділи даного курсу). Кожен розділ інформаційного блоку закінчується спеціально підібраними прикладами, що дозволяють студенту самостійно виконати завдання по темі. Кожна тема оформлена у вигляді окремого уроку (рис. 1), що дозволяє досягти основної цілі навчання.

## Задавание функций, вычисленные значения функций.

Умение строить и понимать графики является обязательным для инженера. Во-первых, зачастую в инженерной практике график – единственная возможность отразить зависимость между двумя величинами. Такие графики выдают осциллографы, самописцы и другие приспособления аналогичного назначения. Во-вторых, при приближенных расчетах сравнение графиков точного и приближенного решения позволяет быстро и наглядно оценить точность применяемого приближенного метода.

Несмотря на кажущуюся простоту действий, связанных с построением графиков и вычислением значений функции, выполнение их без ЭВМ и специальных плакетов – довольно трудоемкая работа. Поэтому рассмотрим возможности пакета MathCad при решении таких задач.

### Сведения по пакету MathCad

Математические символы в MathCad задаются двумя способами: с помощью специальных панелей инструментов или с помощью так называемых кодов, набираемых на клавиатуре.

Панель инструментов **Арифметическая (Arithmetic Toolbar)** вызывается кнопкой  и имеет вид:



Рис. 1. Видяг сторінки підручника

Опис різних областей організовано у виді текстових модулів, доступних натисканням потрібних кнопок. Кожен такий модуль відповідає деякому класу аналітичних методів. Підручник також містить список літератури, використаної в ході викладу.

### Література

1. Ясинский В.Б. Каким должен быть электронный учебник в формате HTML. // Электронный журнал «Исследовано в России»/ – 2001. – №11. – С. 115-129 – <http://zhurnal.ape.relarn.ru/articles/2001/011.pdf>



# **СТРУКТУРНА ОРГАНІЗАЦІЯ СИСТЕМИ МОДЕЛЕЙ РЕПРЕЗЕНТАЦІЇ ПРОЦЕДУРНИХ ЗНАНЬ ДИСЦИПЛІНИ “ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ГАЛУЗІ”**

С.В. Вельма

м. Харків, Українська інженерно-педагогічна академія

Інтенсивний розвиток персональної комп'ютерної техніки та спеціального програмного забезпечення призвів до широкого їх використання спеціалістами різних галузей народного господарства. В процесі впровадження комп'ютерної техніки та технології не є виключенням і фармацевтична галузь. Майбутні спеціалісти фармації повинні не лише оволодіти основами роботи на персональному комп'ютері, тобто стати користувачами ПЕОМ, але й навчитися використовувати комп'ютер для розв'язання своїх професійних завдань

Розв'язанню саме цих задач присвячена дисципліна “Інформаційні технології в галузі”. Одним з центральних завдань комп'ютерної підготовки майбутніх спеціалістів є інтенсивне формування стійких практичних навичок роботи на персональному комп'ютері. Для викладання дисципліни “Інформаційні технології в галузі” було видано навчальний посібник “Практикум з інформаційних технологій у фармації” [1], основу якого складають інтенсивні педагогічні технології формування стійких практичних навичок. Зазначені вище педагогічні технології побудовані на основі використання системи дидактичних імітаційних моделей діяльності. Ці моделі являють собою ієрархічну систему структурно-логічних схем та описують діяльність студентів на дидактичному мікрорівні операцій, дій, рухів [2].

В якості основи для розробки інтенсивних педагогічних технологій професійної комп'ютерної підготовки спеціалістів фармації лежить реалізація основних положень теорії діяльнісного підходу та теорії поетапного формування розумових дій П.Я. Гальперіна та Н.Ф. Талізної [3]. У відповідності з цією теорією організується багаторазове повторення студентами завдань за допомогою тієї чи іншої дидактичної імітаційної моделі. В результаті послідовного, багаторазового виконання всієї сукупності завдань за допомогою моделей у студентів формується си-

стема знань, умінь та навичок за відповідною темою.

Інтенсивність навчання базується на виконанні студентом вірних, хоч спочатку і досить повільних дій. Такий підхід майже повністю виключає формування помилкових навичок, на корегування яких в традиційних технологіях навчання витрачається досить суттєвий обсяг часу. Причина полягає в тому, що перевчити набагато складніше і довше, ніж навчити.

Розглядаючи загальну структуру дисципліни “Інформаційні технології в галузі” (рис. 1), можна виділити чотири модулі, що базуються на задачах, які входять до професійної комп’ютерної підготовки спеціалістів фармації [4].



Рис. 1. Структура дисципліни “Інформаційні технології в галузі”

Для прикладу докладно розглянемо структурну ієрархію моделей, що були розроблені для викладання теми “Операційні оболонки”.

Під час вивчення теми “Операційні оболонки” студенти повинні навчитися працювати з файлами та директоріями (каталогами). На рис. 2 зображені основні задачі, які повинні вміти виконувати студенти, наведені разом з відповідними дидактичними імітаційними моделями.

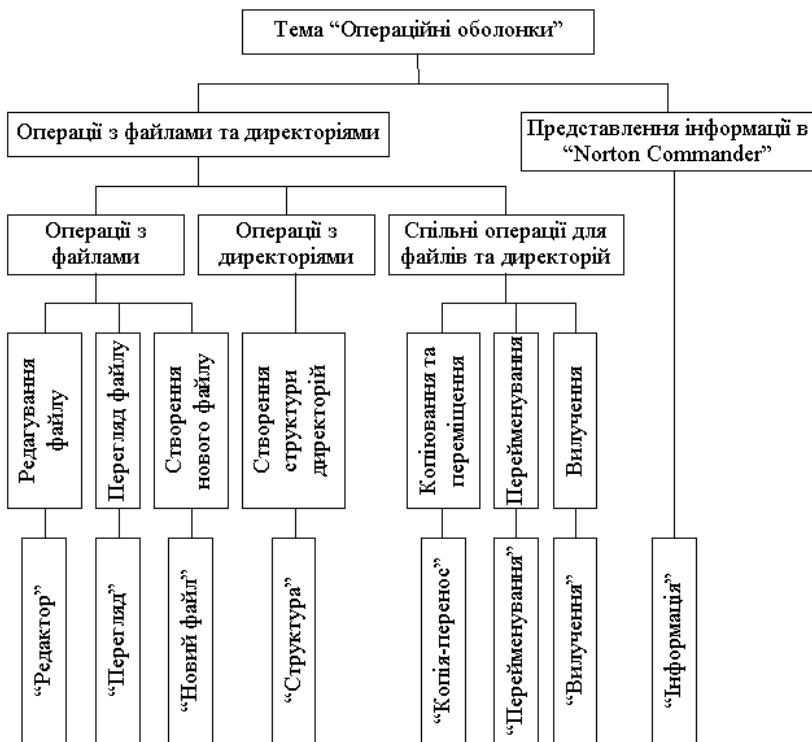


Рис. 2. Декомпозиція теми “Операційні оболонки”

Для вивчення основних задач необхідно засвоїти 8 імітаційних моделей дій. Взагалі, для теми “Операційні оболонки” було розроблено 18 дидактичних імітаційних моделей. На перший погляд це здається забагато. Але справа в тому, що окремі моделі були розроблені для елементарних дій та дій, що часто повторюються в інших програмних засобах, тобто є типовими. Наприклад, до імітаційних моделей елементарних дій можна віднести модель “Команда”, що стоїть на першому рівні ієрархії. Ця модель (рис. 3) не містить вкладених дій, для виконання дії за цією моделлю, взагалі, не потрібні будь-які специфічні знання. До другого рівня ієрархії належать моделі дій, виконання яких спирається на моделі попереднього, першого рівня ієрархії. До моделей другого рівня можна віднести дидактичну імітаційну модель “Курсор”, бо для вірного виконання дії за цією моделлю

необхідно спочатку навчитися виконувати дію за моделлю “Команда”.

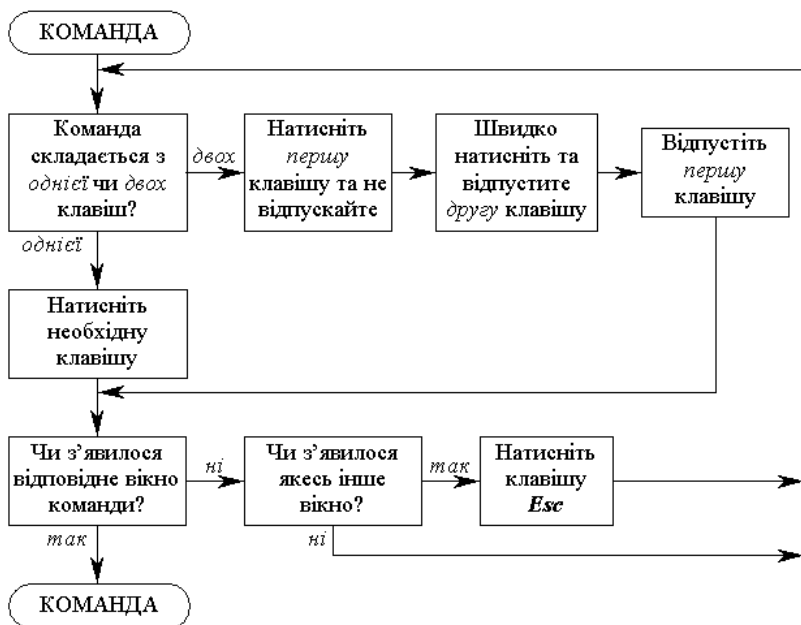


Рис. 3. Модель виконання команди за допомогою функціональних клавіш в Norton Commander

До третього рівня ієрархії належать моделі дій, виконання яких спирається на моделі попередніх, першого та другого рівнів ієрархії, наприклад, моделі “Меню”, “Розкрити директорію”, “Закрити директорію”. Імітаційні моделі дій, які зображені на рис.2 (наприклад, модель “Редактор”, що представлена на рис.4), відносяться до верхнього рівня ієрархії, бо саме ці моделі описують дії, засвоєння яких необхідне для студентів в процесі вивчення теми “Операційні оболонки”.

Центральною задачею побудови процесу навчання на основі використання дидактичних імітаційних моделей є розробка практичних завдань, які б, з одного боку, враховували необхідність багаторазового повторення початкових дій, а з іншого, не порушували логіку викладання навчального матеріалу.

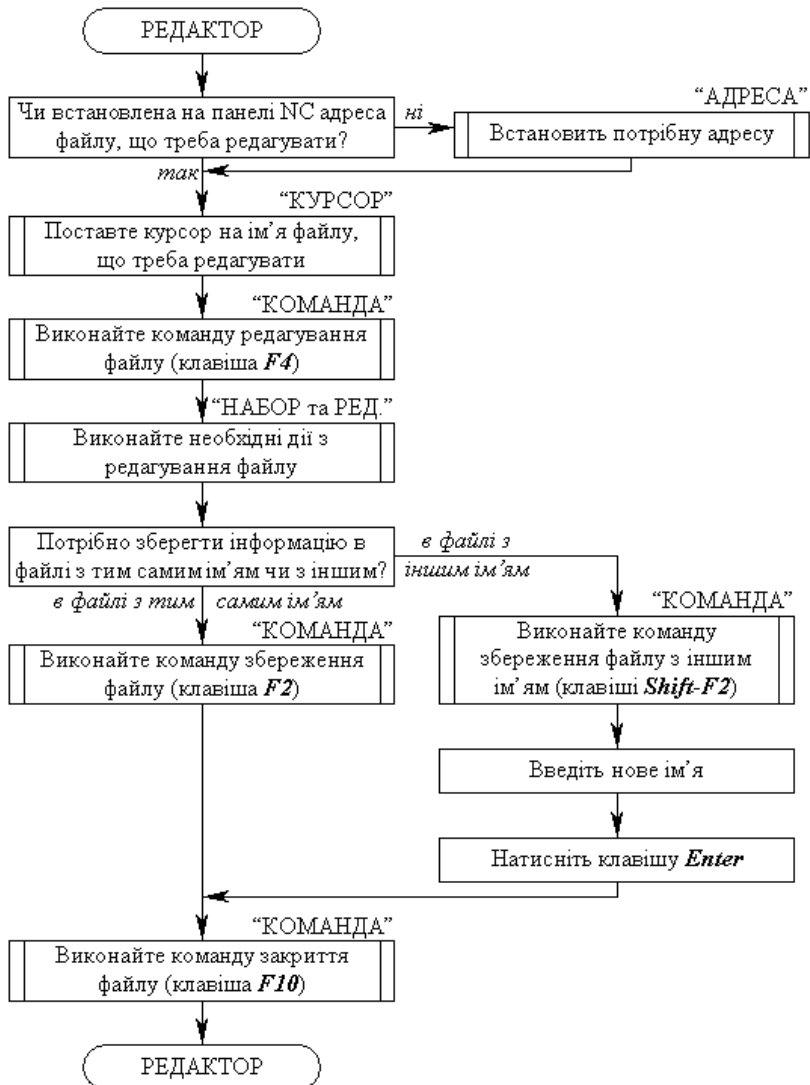


Рис. 4. Модель редагування файлів засобами Norton Commander

Методика та технологія навчання на основі використання дидактичних імітаційних моделей складаються з виконання наступних етапів:

– студент повинен прочитати та зрозуміти поставлене завдання;

– звернутися до відповідної моделі та повільно, без помилок виконати всі вказані в ній дії. При цьому дії, що їх виконує студент, рекомендується проговорювати вголос. Доцільно, щоб на цьому етапі студенти працювали групами по 2-3 людини за комп'ютером (взаємний контроль один одного на етапі повільної роботи дає позитивні результати);

– поступове збільшення швидкості виконання завдань та все більш рідке звертання до відповідних моделей, перехід зовнішнього приговорювання дій до внутрішнього (процес інтеріоризації дій). На цьому етапі студенти повинні працювати на комп'ютерах по одному;

– студенти виконують завдання без моделей взагалі, без помилок та досить швидко. Це свідчить про практичне засвоєння дій, про формування стійкої навички.

Для формування стійкої практичної навички необхідно виконати біля 7-10 завдань для кожної моделі. Кількість завдань, що потрібно виконувати для кожної моделі протягом одного заняття, повинна бути не менша за 4-5.

Практичне використання розробленої методики показало, що витрати навчального часу на формування стійких навичок зменшилося в середньому в 1,8-2,5 рази в порівнянні з традиційними методиками.

### Література

1. Лазарев Н.И., Вельма С.В. Практикум по информационным технологиям в фармации (на основе интенсивных методик обучения): Учеб. пособие для студентов фармац. вузов. – Х.: Изд-во НФАУ: Золотые страницы, 2002. – 264 с.

2. Лазарев М.І. Теоретичні основи розробки мікромодульних педагогічних технологій //Проблеми розробки та впровадження модульної системи професійного навчання: Збірник наук. праць. – К.: Науковий світ, 2001. – С. 131-137.

3. Талызина Н.Ф. Педагогическая психология. – М.: Академия, 1999. – 288 с.

4. Типова програма з курсу “Інформаційні технології в фармації” для студ. вищих фармацевтичних закладів освіти та фармацевтичних факультетів вищих медичних закладів освіти /М.І.Лазарев, В.С.Власов, О.С.Островський та ін. – К.: ЦМК МОЗ України, 2000. – 20 с.

## **ОРГАНІЗАЦІЯ СИСТЕМИ УПРАВЛІННЯ НАВЧАЛЬНИМ ПРОЦЕСОМ НА ОСНОВІ РЕЗУЛЬТАТІВ ПОТОЧНОГО КОНТРОЛЮ З ВИКОРИСТАННЯМ КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ**

В.М. Власенко<sup>1</sup>, В.І. Єфіменко<sup>2</sup>, Р.В. Ромадін<sup>2</sup>

<sup>1</sup> м. Київ, Національний педагогічний університет  
ім. М.П. Драгоманова

<sup>2</sup> м. Черкаси, Черкаський державний університет  
ім. Б. Хмельницького

Створення системи управління неможливе без використання об'єктивних даних про навчальний процес. Виходячи з теорії управління навчально-виховним процесом, можна відзначити, що контроль здійснюється з метою отримання інформації про реальне протікання процесу навчання, а в разі виявлення відхилень від визначеного напрямку, його регулювання з допомогою коригувальних впливів з метою приведення у відповідність з визначеним алгоритмом управління [3].

Ефективне керування навчально-пізнавальним процесом та організація замкненої системи управління можливі тільки на основі результатів поточного контролю. Поточний контроль знань, який пронизує весь навчальний процес, зумовлює підвищення рівня засвоєння навчального матеріалу. Його треба здійснювати в межах усіх організаційних форм навчання і старанно планувати. Головне завдання поточного контролю – допомогти учням організувати свою роботу, навчитися самостійно, відповідально і систематично опанувати навчальний матеріал. Цей вид контролю тісно пов'язаний з усіма видами навчальної роботи. Він привчає учнів готуватись до щоденної перевірки знань, а не лише наприкінці вивчення теми, його покладено в основу створення моделі регулярної самостійної навчально-пізнавальної діяльності. Поточний контроль як важливий компонент входить у навчальну діяльність вчителя.

Організація і проведення контролю – тривала і трудомістка робота. Полегшити та систематизувати її можна використовуючи тестування. Тести дозволяють розбити багатокomпонентні знання, уміння і навички на простіші складові, реєстрація яких фіксу-

ється в однозначній і відносно простій відповіді. Поєднання різних видів тестових завдань з однієї теми створює об'єктивну картину підготовленості групи, і кожного учня зокрема. Тестування є економною, цілеспрямованою і індивідуалізованою формою контролю. Воно дозволяє виявити конкретні прогалини в знаннях, перевірити наскільки усвідомлено учні володіють теоретичним матеріалом, як вони застосовують знання на практиці.

З розвитком інформаційних технологій тестування стає комп'ютерним, що дозволяє:

- застосовувати нові адаптивні алгоритми тестового контролю;
- використовувати в тестах мультимедійні можливості комп'ютерів;
- зменшити об'єм паперової роботи і прискорити підрахунок результатів;
- підвищити оперативність тестування;
- оперативно втручатися в хід навчального процесу з метою його корекції.

Проблема реалізації пов'язаних з контролем функцій при комп'ютерному тестуванні розпадається на три складові – функції підготовки до контролю, функції проведення контролю і функції забезпечення зворотного зв'язку в процесі навчання. При цьому комп'ютер та програмне забезпечення для тестування та обробки його результатів виступають як засоби реалізації системи комп'ютерного контролю.

Підготовка до контролю при комп'ютерному тестуванні полягає насамперед в складанні й оформленні тестових завдань, що вимагає значних затрат часу вчителя. Складаючи тести, підбирають такі запитання, щоб елемент вгадування був мінімальним, а усі правильні відповіді учень обирає тільки на основі знань, які він здобув на заняттях і особливо під час самостійної роботи. Чим більше існує варіантів відповідей до запитання, тим меншою є ймовірність вгадування.

Особливу увагу треба звертати на якість тестових завдань, яка характеризується такими критеріями, як валідність та надійність. Під валідністю розуміють ступінь відповідності результатів тесту деякому об'єктивному показникові. Валідність показує, чи досягає тест мети при вимірюванні, діагностиці чи передба-



ченні того, що він має намір оцінити. Надійність характеризує ступінь точності вимірювання з допомогою тесту [2]. Чим вище валідність та надійність тестових завдань, тим об'єктивніше визначається рівень знань учня.

В процесі виконання функції підготовки до комп'ютерного контролю треба передбачити:

- застосування тестових завдань різних типів, що забезпечує широку варіативність тестових завдань;
- кожному запитанню призначити певний коефіцієнт, який характеризує його складність;
- можливість використання в завданнях графіки, звукових та відео елементів, створених за допомогою різних прикладних програм та відеоапаратури, що дозволяє зробити тестові завдання більш наочними і образними.

Процес створення тесту поділяється на два етапи. На першому етапі емпіричні дані обробляються за допомогою математично-статистичного апарату класичної теорії тестів. На другому, в процесі поглибленого аналізу якості завдань визначають коефіцієнти складності використовуючи спеціальну програму математичного моделювання на основі сучасної теорії тестів (IRT) [1, 4]. Застосування такого математичного апарату дозволяє створювати високоякісні тести, які дозволяють визначати дійсний рівень знань учнів.

Після пілотних досліджень тематичних тестів створюється база даних тестових завдань. Комп'ютерні технології уможливають формування наборів даних взятих з бази (за певними ознаками), обробку набору даних (пошук, сортування, аналіз і зміну інформації за заданими ознаками), використання модуля сервісної технології дозволяє застосування редактора образів, редактора тексту, та регламенту роботи. Підготовлені таким чином матеріали використовуються для тестування під управлінням комп'ютера.

Для аналізу і практичного використання результатів контролю, дані отримані під час тестування необхідно помістити в інформаційну систему – основу банку даних тестових завдань.

Побудована таким чином система створює можливість для підготовки тестових завдань і організації за допомогою них системного коригувального контролю процесу навчання, при якому

регулюється зміст навчального матеріалу і рівень проблемного навчання.

Варіанти організації і проведення контролю, з використанням певної системи завдань, можуть бути різні – від заданого викладачем порядку контролю до автоматичного комп'ютерного управління, і управління порядком контролю по вибору учня. В основі управління ходом контролю лежить комплекс спеціальних алгоритмів, які можуть задавати порядок та час відведений на виконання певного тестового завдання.

Нами розробляється система автоматизованої оцінки якості знань, яка призначена для автоматичного створення тестових завдань на основі сучасної теорії тестів, оцінки результатів тестування та обробки статистичних даних з метою постійного моніторингу якості навчання. Система створюється у вигляді програмної оболонки, що дозволяє використання в межах будь-якого навчального предмета.

Під час підготовки до тестування викладач може вибрати завдання певного рівня складності та їхню кількість надавши комп'ютеру можливість автоматичного формування варіантів контрольних завдань. Таким чином створюється велика кількість варіантів завдань необхідної складності, що забезпечує надання під час контролю індивідуальних завдань кожному учню.

Аналіз і перевірка результатів контролю також проводиться автоматизованим шляхом. За наслідками контролю автоматично формується підсумковий протокол, який може мати наперед задану структуру. В протокол тестування заносяться дані про відповіді на кожне запитання тесту, час, затрачений на проходження всього тесту, час відповіді на кожне запитання, сумарний бал, з урахуванням коефіцієнта складності кожного тестового завдання.

Безперечною перевагою комп'ютерного тестування є можливість обробки статистичних даних результатів контролю. В системі передбачається накопичення і аналіз статистичної інформації для забезпечення постійного зворотного зв'язку, з метою корекції навчального процесу. Для проведення досліджень застосовується апарат сучасної математичної статистики. Є можливість перегляду результатів, що необхідно для детального аналізу та ефективного управління навчально-пізнавальною діяльні-

стю учнів.

На відміну від багатьох тестових програм, проаналізованих нами, особливістю даної системи є те, що в ній передбачається застосування різних видів тестових запитань: альтернативні, з простим множинним вибором, на виявлення відповідності, на встановлення послідовності, введення відповіді у вигляді числових значень (в межах допустимої похибки). Це дозволяє створювати комплексні тести будь-якого рівня складності.

Таким чином система комп'ютерного тестування дозволяє активно впливати на навчальний процес, сприяти ефективному засвоєнню навчального матеріалу, і використовується вчителем для поточного та підсумкового контролю знань, а також учнем для самооцінки рівня його підготовки в процесі самостійної роботи.

#### Література.

1. Єфіменко В.І., Касярум О.П., Власенко В.М. Створення критеріально-орієнтованих тестів досягнень особистості та інтерпретація результатів тестування на основі положень Item Response Theory. //Вісник Черкаського університету. Випуск 26. Серія: педагогічні науки.: Збірник. – Черкаси: ЧДПУ, 2001. – С.40-46.
2. Коваленко А.М., Круцило І.К., Сергєєв О.В. Тестове навчання фізики й перевірка знань учнів // Проблеми методики викладання фізики на сучасному етапі: Зб. статей. – Кіровоград: РВЦ КДПУ ім. В.Винниченка, 2000. – С.54-58.
3. Талызина Н.Ф. Управление процессом усвоения знаний (психолого-педагогические аспекты). – М.: МГУ, 1984. – 344 с.
4. Чельшкова М.Б. Разработка педагогических тестов на основе современных математических моделей. – М. Исслед. центр проблем качества подготовки специалистов, 1995. – 32 с.

## СИСТЕМА ON-LINE ТЕСТІВ ДЛЯ КОНТРОЛЮ ЗНАТЬ З ПРОГРАМУВАННЯ ТА ІНФОРМАТИКИ

В.В. Войтенко, А.В. Морозов

м. Житомир, Житомирський інженерно-технологічний інститут  
voytenko@ziet.zhitomir.ua, mav@zt.ukrtel.net

В умовах запровадження нової шкали оцінювання знань студентів в сучасній освіті та тематичного оцінювання все гостріше постає проблема удосконалення форм і методів контролю знань. При цьому приділяється велика увага розробці україномовних електронних засобів навчання та контролю. Розв'язанню цієї проблеми сприяє як комп'ютеризація освітянської галузі взагалі, так і окремих навчальних закладів.

Представлена тестова система використовується в Житомирському інженерно-технологічному інституті на факультеті інформаційно-комп'ютерних технологій. В даний час за її допомогою регулярно проводиться тестування з предмету "Алгоритмічні мови та основи програмування". Орієнтованість системи на Internet-технології дала змогу застосування даної системи в локальній мережі інституту. Адміністратором системи виділяється власний обліковий запис та пароль для доступу. Можна визначити найбільш складні запитання, обрати час тестування, змінити кількість запропонованих тестових завдань і т.д. Комп'ютерна тестова система дозволяє за мінімальний проміжок часу провести одночасне опитування студентської групи. Універсальність запропонованих в ній тестів полягає у тому, що викладачі можуть запропоновувати довільні тестові завдання та їх відповіді, інтегровано визначати кількість балів за правильні та неправильні відповіді. При цьому викладач може володіти мінімальними навичками користувача ПК.

Відразу після тестування є можливість оцінити якість знань, середній бал по групі, яка тестувалася, що корисно для проведення поточного контролю знань або атестації. Студент до початку тестування за власним бажанням може обирати будь-який рівень складності тестів, на який він претендує у поточний момент. Адреса тестів в Інтернет: [www.ziet.zhitomir.ua:8118](http://www.ziet.zhitomir.ua:8118)

## РОЗВ'ЯЗУВАННЯ ПРИКЛАДНИХ ЗАДАЧ В ЕЛЕКТРОННИХ ТАБЛИЦЯХ

Т.В. Волкова

м. Бердянськ, Бердянський державний педагогічний університет

Одним з провідних напрямків якісної перебудови освітньої системи є перехід від концепції підтримуючого навчання до концепції випереджаючого навчання. Випереджаюче навчання орієнтоване на майбутнє, на ті умови життя та професійної діяльності, в яких випускник школи або вузу опиниться після їх закінчення. Такий підхід є надзвичайно актуальним, оскільки, як відомо, значна частина знань застаріває вже протягом 3-5 років. Тому головна увага у навчальному процесі за такою концепцією звернена на розвиток творчих якостей спеціаліста, його здібностей до самостійних дій в умовах невизначеності, а також розвиток навчальних здібностей до набуття нових знань та навичок, оволодіння сучасними методами отримання, накопичення, класифікації та передачі знань.

Необхідною умовою реалізації системи випереджаючої освіти є її тісний зв'язок з наукою. Освіта повинна бути "вбудованою" в систему наукових досліджень. Для науки такий зв'язок є гарантією більш якісної підготовки майбутніх вчених.

Важливим імперативом часу є залучення до освітнього процесу системи знань у галузі інформатики, її засобів та методів. Розвиток цивілізації відбувається у напрямі інформаційного суспільства, у якому об'єктами та результатами праці більшості населення будуть не матеріальні цінності, а, головним чином, інформація та наукові знання. Вже сьогодні діяльність значної частини населення у розвинутих країнах пов'язана з процесами обробки та передачі різного роду інформації, що примушує людей вивчати та використовувати відповідні інформаційні технології.

Шкільний курс інформатики базується на трьох фундаментальних поняттях: інформація, алгоритм, ЕОМ. Інформатика, як відомо, прикладна наука. Її предмет, основні цілі і задачі визначені практикою. Інформатика має необмежені можливості застосування в різних сферах життя людини. Це наука, техніка, економіка, виробництво, освіта. Таке широке застосування інформа-

тики в сучасному світі підкреслює важливість вивчення в школі саме прикладних аспектів цієї науки. Необхідність підсилення прикладної спрямованості шкільного курсу інформатики в наш інформаційний час перш за все обумовлена очевидною соціальною потребою серйозної підготовки людей різних професій в області застосування засобів інформатики, інформаційних і комунікаційних технологій, практичною необхідністю розвитку в учнів здатності самостійно і творчо застосовувати, постійно доповнювати і удосконалювати запас наявних в них знань і навичок з інформатики.

Досвід викладання базового курсу інформатики для учнів 10 класу в гімназії гуманітарного профілю дозволяє зробити висновок: проблема підсилення прикладної спрямованості базового курсу інформатики може бути успішно вирішена при системному підході до процесу вивчення базового курсу. Важливо при цьому враховувати такі моменти: удосконалення змісту (прикладна спрямованість його вивчення); виділення прикладного аспекту наявних задач шкільного курсу інформатики, розв'язування прикладних задач; використання нових комп'ютерних методів і форм навчання, засобів інформаційних і комунікаційних технологій.

Важливість використання прикладних задач визначається роллю цих задач в розвитку пізнавального інтересу учнів, їх творчих можливостей, самостійності, гнучкості розуму, умінь узагальнювати знання з різних предметів і наук, а також в розвитку інформаційної культури учнів.

Прикладна задача – теоретична задача практичного змісту [1, с. 72]. Прикладна задача – це задача, в якій описується практико-орієнтована ситуація і розв'язання якої вимагає визначених практичних навичок, в тому числі навичок використання засобів інформаційних технологій.

Очевидно, що навчання учнів використанню сучасних інформаційних технологій при розв'язуванні прикладних задач є найактуальнішою вимогою нашого часу. Темою одного з таких занять при вивченні електронних таблиць Microsoft Excel може стати розв'язування задач з різних галузей економіки і управління. Вивчаючи дану тему, можна показати учням, що існують задачі, які важко або неможливо вирішити без застосування

комп'ютерів. З появою математичних пакетів для розв'язування задач використовується принцип конструювання моделі. Ставиться задача, а методи та алгоритми розв'язування знаходить сама система [6]. Значення моделей в навчально-виховному процесі пов'язане з підвищенням наочності і науково-теоретичного рівня викладення матеріалу, з формуванням світогляду учнів, з розвитком їх мислення [7, с. 119]. Схема модельного вивчення об'єктів представлена на рис 1. Можна сказати, що модель – це такий матеріальний або мисленно уявний об'єкт, який заміщує об'єкт-оригінал з метою його дослідження, зберігаючи деякі важливі для даного дослідження типові властивості оригіналу, тобто його суттєві сторони. Гарно поставлена модель, як правило, доступніша для дослідження, ніж реальний об'єкт. Інше, не менш важливе призначення моделі полягає в тому, що з її допомогою виявляються найбільш суттєві фактори, які формують ті чи інші властивості об'єкта. Модель також дозволяє навчитися керувати об'єктом, що важливо в тих випадках, коли експериментувати з об'єктом буває незручно, важко або неможливо.

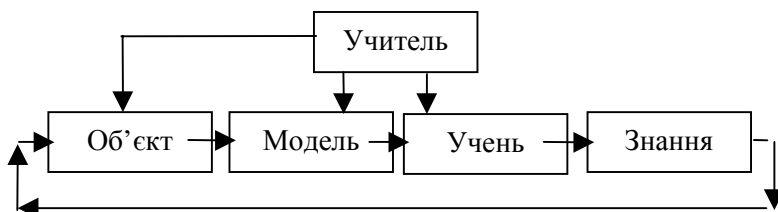


Рис. 1. Схема модельного вивчення об'єктів

Сучасна ЕОМ надає широкі можливості для моделювання різних явищ і процесів. В наш час одним з перспективних, але недостатньо розповсюджених способів розв'язування прикладних задач з економіки і управління є використання надбудови “Поиск решения” електронних таблиць MS Excel. Так, “Поиск решения” надає можливість використання планів великої розмірності (тобто з великою кількістю змінних); задання обмежень складного типу; пошуку оптимального з допустимих розв'язків; генерування множини різних рішень, які зберігаються в подальшому у вигляді сценаріїв; автоматичного створення звітів за результатами розв'язків.

Основні технологічні ланцюжки розв'язання прикладної за-

дачі з використанням команди електронної таблиці “Поиск решения”: побудова математичної моделі – задання обмежень – задання критерію оптимізації – вибір пошукових змінних – розв’язування задачі – аналіз результатів.

В процесі використання електронних таблиць для розв’язування прикладних задач необхідне розуміння учнями значення таких термінів, як «цільова комірка», «змінна комірка», «обмеження». «Цільова комірка» – це комірка, до якої програма введе шукане значення (мінімальне або максимальне) після розрахунків та перебору можливих варіантів розв’язків. «Змінні комірки» служать для розміщення в них значення змінних величин, при яких у «цільовій комірці» буде знайдено оптимальне значення. «Обмеження» – умови, яким повинні задовольняти значення у «змінних комірках».

Теоретичною основою надбудови “Поиск решения” є симплексний метод, який дозволяє знаходити оптимальний розв’язок задачі за допомогою ітераційного процесу переходу до найкращого розв’язку. Додаток електронної таблиці “Поиск решения” може не входити до стандартного варіанту установки. Для його додавання достатньо скористатися командою Сервіс, Надстройки, Поиск решения.

Послідовність розв’язування будь-якої прикладної задачі в електронних таблицях за допомогою команди “Поиск решения” завжди однакова і складається з таких етапів:

1. Складання розрахункової таблиці. На робочому аркуші створюється таблиця, в якій виділяються комірки для запису початкових даних, запису формул, виведення результатів розрахунків.

2. Занесення до комірок розрахункової таблиці формул та початкових даних. Цей етап є дуже важливим і найскладнішим тому, що вимагає від учня знань можливостей програми електронних таблиць, а також поглибленого розуміння суті задачі, що розв’язується.

3. Викликають команду “Поиск решения” і в її діалоговому вікні вказують адресу цільової комірки, адреси змінних комірок, адреси обмежень на значення даних у змінних комірках. Дають команду “Выполнить”.

4. Програма знайде розв’язання при заданих умовах і у від-



повідні комірки запише результати розрахунків.

5. Якщо розв'язання не може бути знайдене, необхідно змінити обмеження і запустити програму на виконання ще раз.

**Як приклад,** розглядається розв'язання транспортної задачі за критерієм вартості. Вперше вона виникла як задача визначення оптимальних схем перевезень [3, с. 22; 4, с. 23; 5, с. 201].

**Постановка задачі:** У деякого підприємця, який займається виробництвом та продажем підручників, є дві фабрики, які поставляють ці підручники до трьох книгарень. Кожна книгарня замовляє визначену кількість підручників, при цьому і кожна з фабрик може виділити лише визначену кількість підручників. Нехай першій книгарні, позначимо її  $B_1$ , потрібно 125 підручників "Інформатика", книгарні  $B_2$  – 86 і книгарні  $B_3$  – 94, таким чином 305 підручників. Нехай підприємець вирішив виготовити 163 підручники на першій фабриці  $F_1$  і 142 підручники на другій фабриці  $F_2$  (рис. 2).

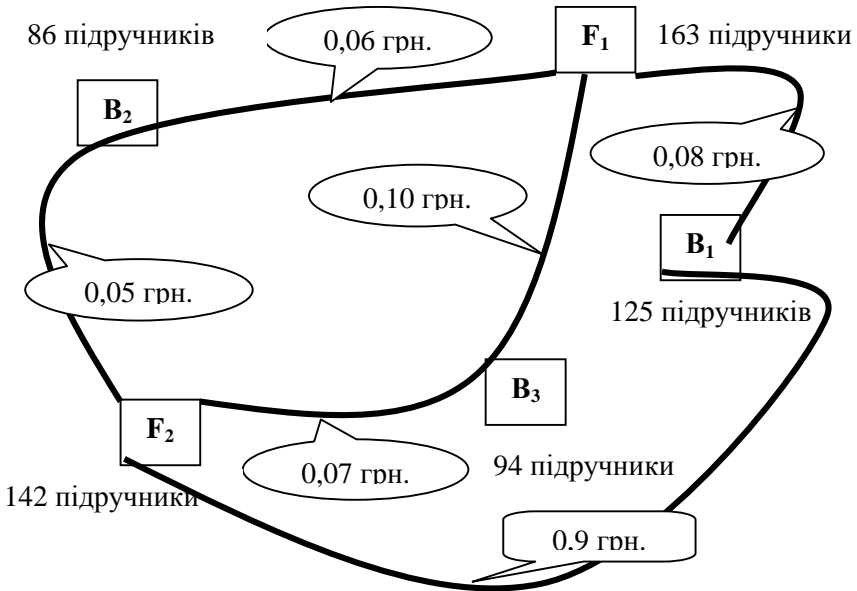


Рис.2. Схема розміщення фабрик і книгарень

Задача полягає в тому, щоб з'ясувати, скільки підручників потрібно відправити з кожної фабрики до кожної книгарні, щоб

загальна вартість всіх перевезень була мінімальною. Припускається, що можна відправити підручники деякими транспортними шляхами, які зв'язують кожну фабрику зі всіма книгарнями. Припускається, що нам відома вартість перевезення одного підручника з фабрики до книгарні і що вона лінійна, тобто якщо вартість перевезення одного підручника складає 0,02 грн., то вартість перевезення двох підручників складає 0,04 грн. Треба скласти такий план перевезень, за яким їхня загальна вартість буде найменшою.

**Побудова математичної моделі.** Приймаються такі позначення:  $m$  – кількість фабрик,  $n$  – кількість книгарень,  $a_i$  – загальна кількість підручників, яка виділяється для перевезення з  $i$ -ї фабрики,  $b_j$  – кількість підручників, яка необхідна  $j$ -й книгарні,  $x_{ij}$  – кількість підручників, яка перевозиться з  $i$ -ї фабрики до  $j$ -ї книгарні. Припускається, що загальна кількість підручників, яка виділяється для перевезення, дорівнює потрібній кількості, тобто

$$\sum_i a_i = \sum_j b_j .$$

Задача полягає у визначенні невідомих перевезень  $x_{ij}$ . Якщо скласти таблицю, то для випадку  $m=3$  і  $n=2$  вона буде мати такий вигляд (таблиця 1).

Таблиця 1.

Фабрики	Книгарні	1	2	3
1		$x_{11}$	$x_{12}$	$x_{13}$
2		$x_{21}$	$x_{22}$	$x_{23}$

З цієї таблиці видно, що сукупність перевезень з першої фабрики задовольняє лінійному рівнянню  $x_{11}+x_{12}+x_{13}=a_1$ . Аналогічно для фабрики 2:  $x_{21}+x_{22}+x_{23}=a_2$ . Записуються умови, що накладаються на перевезення  $x_{ij}$ , які направляються до кожної з трьох книгарень:

$$\left. \begin{aligned} x_{11} + x_{21} &= b_1 \\ x_{12} + x_{22} &= b_2 \\ x_{13} + x_{23} &= b_3 \end{aligned} \right\}$$

Підприємець знає вартість перевезення одного підручника від  $i$ -ї фабрики до  $j$ -ї книгарні. Так як залежність вартості перевезень від кількості товару лінійна, то вартість перевезень  $x_{ij}$  під-

ручників складає  $c_{ij}x_{ij}$ . Підприємцеві бажано визначити, скільки одиниць підручників  $x_{ij}$  потрібно відправити з кожної фабрики до кожної книгарні, щоб загальна вартість перевезень була мінімальною. Цільова функція є лінійною формою

$$c_{11}x_{11} + c_{12}x_{12} + c_{13}x_{13} + c_{21}x_{21} + c_{22}x_{22} + c_{23}x_{23},$$

яку необхідно перетворити на мінімум. Оскільки невід'ємні значення  $x_{ij}$  відповідають зворотнім перевезенням від  $j$ -ї книгарні до  $i$ -ї фабрики, то вимагається, щоб всі змінні  $x_{ij} \geq 0$ .

Розв'язування транспортної задачі в електронній таблиці.

Складання розрахункової таблиці. Використовується таблиця з вихідними даними (рис. 3).

	A	B	C	D	E	F
2		B1	B2	B3		
3	F1	0,08	0,06	0,1		
4	F2	0,09	0,05	0,07		
5						
6		кількість підручників				
7		B1	B2	B3	обмеження	формули
8	F1	0	0	0	163	
9	F2	0	0	0	142	
10	обмеження	125	66	94		
11						
12	формули					
13		функція мети			0	

Рис. 3. Електронна таблиця з вихідними даними для розв'язування транспортної задачі

Заносяться формули для обчислення суми підручників, які будуть вироблені фабриками до комірок розрахункової таблиці F8:F9. До комірок B12:D12 заносяться формули суми підручників, які будуть доставлені до книгарень. В комірку D13 заноситься формула для розрахунку загальної вартості перевезень (рис. 4).

	A	B	C	D	E	F
2		B1	B2	B3		
3	F1	0,08	0,06	0,1		
4	F2	0,09	0,05	0,07		
5						
6		кількість підручників				
7		B1	B2	B3	об'єм	формули
8	F1	0	0	0	163	=SUM(B8:D8)
9	F2	0	0	0	142	=SUM(B9:D9)
10	обмеж	125	86	94		
11						
12	формули	=SUM(B8:B9)	=SUM(C8:C9)	=SUM(D8:D9)		
13	функція мети	=SUMPRODUCT(B3:D4;B8:D9)				

Рис. 4. Електронна таблиця з розрахунковими формулами для розв’язування транспортної задачі

Викликається діалогове вікно команди “Поиск решения” (рис. 5). Вказується адреса “цільової комірки”; вибирається мінімальне значення для “цільової комірки”; вказуються адреси тих комірок, змінюючи значення яких, команда “Поиск решения” буде підбирати мінімальне значення в “цільовій комірці”; вводяться обмеження на розрахунки.

Рис. 5. Діалогове вікно команди “Поиск решения”

Натискається клавіша “Выполнить”. Комірки B8:D9 запов-

нюються числами. Це кількість підручників, яку необхідно перевезти від кожної фабрики до кожної книгарні, щоб забезпечити найменші транспортні витрати. В комірці D13 виводиться значення “цільової функції”, яке відповідає мінімальному значенню вартості перевезень підручників (рис. 6).

	A	B	C	D	E	F
1		вартість перевезень				
2		B1	B2	B3		
3	F1	0,08	0,06	0,1		
4	F2	0,09	0,05	0,07		
5						
6		кількість підручників				
7		B1	B2	B3	обмеж	формули
8	F1	125	38	0	163	163
9	F2	0	48	94	142	142
10	обмеження	125	86	94		
11						
12	формули	125	86	94		
13	функція мети			21,26		

Рис. 6. Розрахунковий лист електронної таблиці після виконання команди “Поиск решения”

**Аналіз результатів.** Таблиця показує, що при найменших транспортних витратах (21,26грн.) з першої фабрики до книгарень відправляється 163 підручники ( $x_{11}=125$ ,  $x_{12}=38$ ,  $x_{13}=0$ ), а з другої фабрики відправляється 142 підручники ( $x_{21}=0$ ,  $x_{22}=48$ ,  $x_{23}=94$ ). Загальна кількість підручників, які отримані першою книгарнею з обох фабрик, дорівнює 125 одиниць ( $x_{11}=125$ ,  $x_{21}=0$ ), другою книгарнею – 86 одиниць ( $x_{12}=38$ ,  $x_{22}=48$ ) і третьою книгарнею – 94 підручники ( $x_{13}=0$ ,  $x_{23}=94$ ). За розрахунковими даними будуватиметься діаграма (рис. 7).

Етап підведення підсумків має педагогічні переваги в залученні учнів до творчої діяльності. Так, заключний етап може слугувати полігоном для складання на базі даної задачної ситуа-

ції нових, споріднених задач [8, с. 23].

### Доставка підручників з фабрик до книгарень

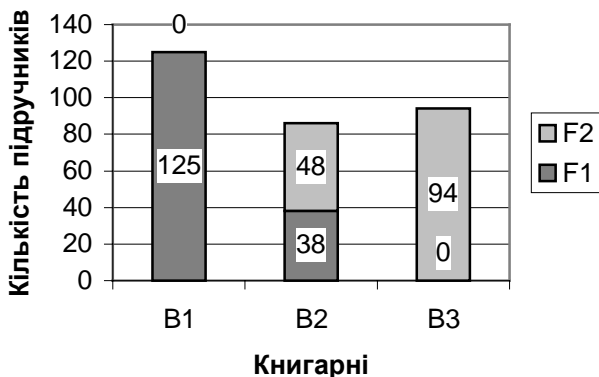


Рис. 7. Діаграма доставки підручників

Учням пропонується, використовуючи попередні вихідні дані, визначити максимум цільової функції. Як видно з листа електронної таблиці, при найбільших транспортних витратах (25,64 грн.) з першої фабрики до книгарень відправляється 163 підручники ( $x_{11}=0, x_{12}=69, x_{13}=94$ ), а з другої фабрики відправляється 142 підручники ( $x_{21}=125, x_{22}=17, x_{23}=0$ ). Загальна кількість підручників, які отримані першою книгарнею з обох фабрик, дорівнює 125 одиниць ( $x_{11}=0, x_{21}=125$ ), другою книгарнею – 86 одиниць ( $x_{12}=39, x_{22}=17$ ) і третьою книгарнею – 94 підручники ( $x_{13}=94, x_{23}=0$ ).

Основною позитивною рисою будь-якого пакета програмних засобів є те, що імітаційну модель можна запускати декілька разів, часто протягом короткого проміжку часу – доти, поки не буде потрібного або принаймні прийняттого результату. Під час цього процесу учень краще сприймає значення взаємодії різних факторів, а також глибше розуміє сутність подій, що відбуваються [2, с. 87].

## Література

1. Балл Г.А. Теория учебных задач: Психолого-педагогический аспект. – М.: Педагогика, 1990. – 184 с.
2. Вільямс Р., Маклін К. Комп'ютери в школі: Пер. з англ. / За ред. В.Б. Распопова. – К.: Радянська школа, 1988. – 295 с.
3. Гасс С. Линейное программирование. Методы и приложения: Пер. с англ./ Под ред. Д.Б. Юдина. – М., 1961. – 303 с.
4. Гасс С. Петешествие в страну линейного программирования: Пер. с англ./ Под ред. Ю.Н. Сударева. – М., 1973. – 175 с.
5. Гетманцев В.Д. Лінійна алгебра і лінійне програмування: Навч. посібник. –К.: Либідь, 2001. – 256 с.
6. Жалдак М.І. Комп'ютер на уроках математики: Посібник для вчителів. – К.: Техніка, 1997. – 303 с.
7. Изучение основ информатики и вычислительной техники в средней школе: опыт и перспективы / Сост. В.М. Монахов и др. –М.: Просвещение, 1987. – 192 с.
8. Саранцев Г.И., Миганова Е.Ю. Функции задач в процессе обучения. – Педагогика. – 2001. – №9.

## **ПЕРСПЕКТИВЫ ПРИМЕНЕНИЯ ОПЕРАЦИОННОЙ СИСТЕМЫ LINUX В УЧЕБНОМ ПРОЦЕССЕ**

А.Ю. Гайда

г. Николаев, Украинский государственный морской технический университет имени адмирала Макарова

Одним из главных направлений применения компьютерных информационных технологий в учебном процессе является повышение его эффективности за счет предоставления организованной среды распространения информации и взаимодействия между участниками учебного процесса [2]. Такую среду можно рассматривать как информационную систему (ИС) учебного заведения, в рамках которой необходимо решить задачи накопления, организации и предоставления информационных ресурсов [1].

На ранних этапах развития ИС учебных заведений происходит накопление информации, которую преподаватели размещают в ИС с целью повторного использования в учебном процессе. Организация информации производится непосредственно преподавателем во время реализации учебного процесса. Задача предоставления информации решается студентами путем выбора необходимых информационных ресурсов.

По мере развития ИС и накопления информационных ресурсов производится разработка средств организации и предоставления информации, которые предоставляют организованный доступ к информационным ресурсам в соответствии с учебными программами. В таких ИС акценты в работе преподавателя постепенно смещаются от организации информации к разработке и размещению информационных ресурсов.

Процесс обучения обычно предполагает наличие обратной связи между преподавателем и студентом, которая позволяет преподавателю учесть индивидуальные особенности студента. Развитая ИС учебного заведения может извлекать такую информацию в процессе взаимодействия студента с ИС (темп и количество обращений, результаты тестирований и т.п.) [3]. ИС, в которых информация о индивидуальных особенностях студента может быть использована процессами организации и предостав-



ления информации наряду с информацией об учебных программах можно рассматривать как системы дистанционного обучения (ДО).

Применение информационных технологий в учебном процессе имеет ряд особенностей:

- при работе студента в аудитории учебного заведения крайне затруднительно обеспечить каждого учащегося персональным компьютеризированным рабочим местом;
- при удаленном доступе студента к ИС учебного заведения затруднительно обеспечить рабочее место студента однотипным программным обеспечением с необходимыми настройками.

Еще одной проблемой является необходимость перемещения результатов работы преподавателя и студента с одного рабочего места на другое и обеспечение защиты этих результатов от несанкционированного использования. Одним из возможных решений этих проблем может быть моделирование программными средствами определенного информационного окружения, индивидуального для каждого пользователя – «виртуальных персональных компьютеров» и, на их основе, – «виртуальных учебных классов» и «виртуального образовательного пространства».

Для реализации процессов накопления, организации и предоставления информационных ресурсов и с целью моделирования «виртуального образовательного пространства» сервер в образовательной ИС должен предоставлять три вида информационных услуг:

1. Быть средой выполнения. При этом каждому клиенту должен предоставляться единообразный набор инструментальных средств – системных и прикладных программ, – хранимых в общей части сервера с возможностью иметь персональные настройки, хранимые в персональном каталоге пользователя. В этом режиме сервер должен работать в архитектуре клиент-сервер.
2. Быть средой хранения. В зависимости от назначения, доступ к файлам может быть осуществлен с использованием средств передачи файлов FTP (File Transport Protocol) между клиентским рабочим местом и сервером, или средств WWW (World Wide Web) для просмотра и сохранения на клиентском месте

текстовых и иных документов. Сервер должен обеспечивать защиту ресурсов пользователей от несанкционированного использования и управляемый доступ для владельца, рабочей группы и остальных зарегистрированных пользователей системы. В этом режиме сервер работает в архитектуре файл-сервер.

3. Быть средой взаимодействия. Средства FTP- и Web-доступа позволяют преподавателям в пределах своего персонального каталога размещать учебно-методические и нормативные материалы, а студентам – результаты выполнения своих работ. Общедоступные WWW и FTP серверы могут использоваться для размещения таких информационных ресурсов, как электронные учебники, пакеты прикладных программ и т.п. Еще одним важным сервисом обеспечения информационного обмена пользователей системы может быть электронная почта.

Два первых аспекта применения сервера могут использоваться в очном обучении. Для реализации системы ДО необходимы все три аспекта. Исходя из этого, при правильном выборе инструментальных средств могут быть достигнуты такие свойства ИС, которые позволят поэтапно развивать ее от вспомогательной образовательной системы очного обучения (среда выполнения, среда хранения) до полноценной системы ДО (среда выполнения, среда хранения, среда взаимодействия). Наличие среды выполнения позволяет размещать в ИС обучающие программы, что предоставляет возможность тиражировать и распространять электронные учебные материалы за пределами учебного заведения, сохраняя за учебным заведением его главную функцию – реализацию учебного процесса. Кроме этого, среда выполнения позволяет в пределах очного и дистанционного обучения предоставить всем участникам учебного процесса единые программные средства для обучения.

Задачи создания «виртуальных персональных компьютеров» и «виртуального образовательного пространства» могут быть решены с применением многопользовательских ОС, таких как UNIX-системы, Novell NetWare, Windows NT, но только UNIX-системы позволяют наиболее просто и общепринятыми средствами реализовать среду выполнения [4]. В последнее время среды UNIX-систем наиболее стремительно развивается ОС Linux.

Анализ предоставляемых ОС Linux сервисов и их потребительских качеств показывает, что в области серверных вычислений Linux предлагает целый ряд комплексных решений и по своим возможностям превосходит ближайшего конкурента от Microsoft – Windows NT (Windows 2000) по стоимости и надежности при полной лицензионной чистоте системного программного обеспечения. Как клиентское рабочее место, Linux имеет как сильные, так и слабые стороны: широкий выбор инструментов разработки и мощь отдельных прикладных программ в этой ОС соседствуют со слабой интеграцией различных пакетов программ. Система Novell NetWare может эффективно выполнять только функции среды хранения, чего явно недостаточно для реализации образовательной среды.

Еще одним важным аспектом применения ОС Linux является возможность использовать наиболее простое клиентское программное обеспечение («тонкий клиент») – через текстовые и графические терминалы на основе программ, аналогичных «Telnet» и «X Window». При этом, требования к программному и аппаратному обеспечению, настройка клиентского рабочего места могут быть наиболее простыми, а настройка сервера – централизованной.

В настоящее время на кафедре Информационных технологий УГМУ персональные компьютеры всех учебных классов объединены в единую локальную вычислительную сеть. В этой сети также размещен сервер с ОС Linux (Linux Red Hat 7.3, Pentium III/550, оперативная память 256 Мб, накопитель 40 Gb), с помощью которого решается следующий круг задач:

1. Предоставляется среда выполнения:
  - каждому участнику учебного процесса предоставлен персональный счет и единые инструментальные средства разработки ПО и документов;
  - каждому студенту, обучающемуся по специальностям кафедры, предоставлены персональная база данных и средства Web-сервера для выполнения лабораторных и практических работ.
2. Предоставляется среда хранения:
  - персональные (домашние) каталоги пользователей защищены паролем;

- на дисковое пространство под персональные каталоги пользователей установлены квоты.
- 3. Предоставляется среда взаимодействия:
  - пользователи организованы в рабочие группы, в пределах которых может производиться обмен информацией и сообщениями;
  - к серверу предоставлен доступ в терминальном режиме средствами программ «ssh» из ОС Linux и «putty» из ОС Windows (для текстовой консоли) и средствами системы X Window «XFree86» из ОС Linux и «X-Win32» из ОС Windows (для графической консоли).

Процессы накопления, организации и предоставления информационных ресурсов реализованы следующим образом:

1. Накопление информационных ресурсов производится путем размещения «информационных файлов» в выделенных каталогах персональных каталогов преподавателей кафедры.
2. Организация информационных ресурсов производится путем создания учебных документов из «информационных файлов» на основе учебных программ и планов и с помощью разработанного программного обеспечения.
3. Предоставление информационных ресурсов производится на основе списков пользователей (преподавателей и студентов) с помощью средств WWW и FTP.

Трехлетний опыт развития и эксплуатации ИС на основе ОС Linux показал:

- высокую надежность данной операционной системы;
- относительно высокие темпы развития при низких трудозатратах за счет использования возможностей ОС Linux (удаленный доступ, защита информации, средства разработки);
- наличие педагогического эффекта – многие студенты с интересом воспринимают тот факт, что в ИС для них выделены персональные ресурсы, а их использование способствует повышению учебной дисциплины.

На основе вышеизложенного можно сделать следующие выводы:

1. Системы ДО можно рассматривать как ИС, достигшие определенного уровня развития.
2. Предоставление среды выполнения является важной задачей

- образовательной ИС, а терминальный доступ позволяет обеспечить единообразие ПО клиентских мест и продлить жизнь морально устаревших вычислительных систем в качестве терминалов клиентских рабочих мест.
3. Применение бесплатного и лицензионно-чистого программного обеспечения на основе ОС Linux позволяет создать надежную информационную систему, наиболее полно отвечающую потребностям образовательной ИС.

#### Литература:

1. Гайда А.Ю. Управление проектами в дистанционном образовании. / Вестник Херсонского государственного технического университета. – 2002. – №1(14). – С. 485-488.
2. Рашкевич Ю.М., Пелешко Д.Д., Пасека М.С., Стецюк А.Б. Структурний аналіз систем управління навчанням. – Вестник Херсонского государственного технического университета. – 2002. – №1(14). – С. 464-470.
3. Романов А.Н., Торопцов В.С., Григорович Д.Б. Технология дистанционного обучения. – М.: ЮНИТИ, 2000. – 221 с.
4. Немет Э., Снайдер Г., Сибасс С., Хейн Т.Р. UNIX: руководство системного администратора. Для профессионалов / Пер. с англ. – СПб.: Питер; К.: Издательская группа BHV, 2002. – 928 с.: ил.

## **КОМПЛЕКСНАЯ АВТОМАТИЗАЦИЯ ПРОВЕРКИ ЗНАНИЙ СТУДЕНТА ДЛЯ СПЕЦИАЛЬНОСТИ «ЭКОНОМИЧЕСКАЯ КИБЕРНЕТИКА»**

Е.М. Галбай, Н.Н. Дудникова

г. Горловка, Автомобильно-дорожный институт Донецкого национального технического университета

Труд преподавателя остается одной из немногих областей человеческой деятельности, в которой до сих пор сохраняется «ручной труд», неизбежно малопроизводительный, изнурительный и дорогой.

Тестирование знаний студентов, проведение контрольных работ, защита лабораторных, довольно частая и рутинная задача, с которой сталкивается преподаватель. Появление и достаточно широкая доступность персональных компьютеров коренным образом изменяет сложившуюся систему обучения.

При современной наполняемости классов преподаватель физически не в состоянии осуществить принцип индивидуального обучения и общения со студентом, не затрачивая на это дополнительное время. Решением данной проблемы является создание тестирующей программы проверки знаний студента.

Созданная на кафедре АП и ПЭВМ Автомобильно-дорожного института Донецкого национального технического университета программа проверки знаний студента представляет собой форму, на которой отображаются вопросы и варианты ответов. В ходе тестирования нажатием левой кнопкой мыши студент выбирает вариант ответа, который он считает верным. О правильности своего ответа он может судить по изменению цвета букв в выбранном варианте: с синего на зеленый – ответ правильный, с синего на красный – ответ неправильный. Для наглядности на форме расположена гистограмма, которая фиксирует уровень знаний студента на данном этапе тестирования (в одном столбце накапливаются правильные ответы, в другом – неправильные ответы). Тестирование ограничено по времени.

Вопросы с вариантами ответов находятся в текстовом файле, который логически связан с другим текстовым файлом, хранящим номера правильных ответов. Количество вопросов, храня-

щихся в файле, не ограничено. Гибкость данной программы заключается в том, что в любой момент времени вопросы можно дополнять и изменять, редактируя только текстовые файлы.

Главная особенность программы состоит в том, что при очередном запуске программы последовательность появления вопросов и порядковый номер правильного ответа носят случайный характер, т.е. исключена возможность подбора списка номеров правильных ответов.

Количество задаваемых вопросов может быть ограничено.

По окончании тестирования на экран выводится форма, на которой отображены фамилия и группа тестируемого, дата и время проведения теста, название дисциплины, количество заданных вопросов и количество правильных ответов, оценка и электронная подпись преподавателя, которая является своего рода подтверждением сдачи теста и защитой от подделывания результатов.

Программа занимает всего 30 КБ дискового пространства. и работает с текстовыми файлами с расширением txt. Такой файл, содержащий 100 вопросов, занимает не более 20 КБ.

Разработанная программа может применяться для проверки уровня знаний студента во всех областях знаний.

Для создания программы проверки уровня знаний нам необходимо подготовить форму. Прежде всего, вынесем на форму метку Label1, размер которой должен быть достаточным для занесения в нее вопроса, состоящего из нескольких строк (для специальности «Экономическая кибернетика» вопрос может состоять из фрагмента кода программы и занимать 5-7 строк – и это нужно учесть при задании размера метки).

Ниже на форму поместим массив меток Label2(), в которых в ходе выполнения программы будут отображены варианты ответов. Эти метки, имеющие одинаковые имя, размер, будут различаться только одним свойством – Index.

Для контроля времени проведения теста на форму поместим таймер Timer1, свойству Interval которого зададим значение 1000, а свойству Enabled значение False.

Предположим, у вас уже есть два текстовых файла, один из которых содержит вопросы и варианты ответов, другой – номера правильных ответов.

При загрузке программы данные из текстового файла считываются в массив  $a(i)$ , каждый элемент которого будет содержать строку из текстового файла и где  $i$  – номер строки. Данные будут считываться до тех пор, пока не будет достигнут конец файла, т.е. используется цикл Do While Not EOF ...Loop. Поэтому любые изменения, сделанные в текстовом файле до загрузки программы будут замечены. Вопросы друг от друга должны быть отделены пустой строкой.

Далее отсортируем данные из массива  $a(i)$  и положим их в два других – массив  $c(k)$ , который будет содержать вопросы, где переменной  $k$  будет присвоен номер вопроса, и массив  $b(k,j)$ , элементы которого будут содержать варианты ответов, где переменная  $k$  – номер вопроса, переменная  $j$  – номер варианта ответа.

Из другого текстового файла данные считываются в массив  $d(v)$ , где  $v$  – номер строки и номер вопроса соответственно. Элементы массива будут содержать цифру правильного ответа на  $v$ -й вопрос.

При тестировании программы в метку Label1 заносится один из элементов массива  $c(k)$ , где переменная  $k$  – номер вопроса – выбирается случайным образом: Для генерации случайных чисел используется оператор Randomize и функция Rnd. Если не применять оператор Randomize, то функция Rnd всегда будет использовать одно и то же число в качестве опорного при первом вызове, т.е. полученное таким образом число не будет таким уж «случайным». После того как функция Rnd возвратит нам какое-то число от 0 до 1, оно умножается на общее количество вопросов в текстовом файле и округляется до целого. Такой же принцип действует и при занесении вариантов ответов в массив меток Label2().

Что касается вариантов ответов, помещаемых в ходе выполнения программы в массив меток, то их порядковый номер запоминается в массиве  $g(i)$ . К примеру, вопрос № 6 имеет 4 варианта ответа. В текстовом файле правильный ответ стоит на 3 месте. Предположим, что в ходе тестирования программы он был помещен в первую метку. Значит, элемент массива  $g(1)$  будет содержать цифру 3.

При тестировании студент выбирает правильный ответ нажатием левой кнопки мыши по одной из меток, содержащих



один из вариантов ответа, при этом генерируется событие `MouseDown`. После отпускания кнопки мыши обрабатывается событие `MouseUp` для массива меток `Label2()`, в котором проверяется правильность выбранного ответа. А именно сравнивается цифра из текстового файла с цифрой, которую хранит элемент массива `g(Index)`, где `Index` – свойство метки, определяющее в данной программе ее порядковый номер, т.е. метка, стоящая на первой месте, имеет свойство `Index` равное 1, стоящая на втором месте – 2 и т.д.

Вернемся к нашему примеру. Предположим, что студент при выборе правильного ответа на вопрос №6 щелкнул левой кнопкой мыши по первой метке. Генерируется событие `MouseDown` для массива меток `Label2()`, в котором происходит сравнении элемента массива `d(6)` с элементом массива `g(1)`. В данном случае оба элемента массива будут содержать цифру 3, и так как выбранный ответ совпадает с правильным, мы зададим свойству `ForeColor` данной метки такое значение, которое изменит цвет букв с синего на зеленый. В противном случае свойство `ForeColor` будет иметь значение, определяющее красный цвет букв.

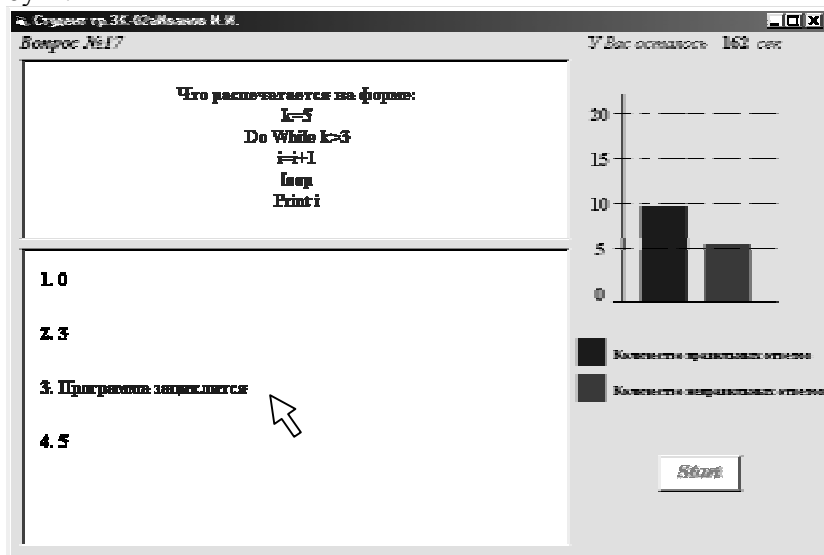


Рис 1. Программа тестирования уровня знаний

Таким образом, о правильности своего ответа студент может судить по изменению цвета букв в выбранном варианте: с синего на зеленый – ответ правильный, с синего на красный – ответ неправильный.

После отпускания кнопки мыши обрабатывается событие `MouseDown`, о котором было сказано выше. Сравнение элементов массива происходит так же, только в этом случае, если значения сравниваемых элементов совпали, переменная, которая накапливает количество правильных ответов увеличится на 1, в противном случае на 1 увеличится переменная, которая хранит количество неправильных ответов.

При перемещении мыши над элементом генерируется событие `MouseMove`. В его процедуре можно изменить состояние элемента, находящегося под указателем мыши. К примеру для того, чтобы изменить цвет букв при перемещении мыши над меткой необходимо в процедуре `Sub Label2()_MouseMove` изменить свойство `ForeColor` метки на нужный цвет. Для всех остальных элементов данного массива цвет букв должен оставаться черным (цвет, который установлен по умолчанию).

В рабочем виде программа представлена на рис 1.

Для запуска программы на форму была помещена кнопка `Start`, при нажатии на которой появляется диалоговое окно, в котором студент должен ввести свое имя, фамилию и группу. После этого свойство `Enabled` таймера `Timer1` установим в `True`. Именно после этого время начинает свой отсчет.

По окончании тестирования на экран выводится форма, на которой отображены фамилия и группа тестируемого, дата и время проведения теста, название дисциплины, количество заданных вопросов и количество правильных ответов, оценка и электронная подпись преподавателя, которая является своего рода подтверждением сдачи теста и защитой от подделывания результатов.

# ВИКОРИСТАННЯ МЕТОДУ ПРОЕКТІВ У ПРОЦЕСІ ВИВЧЕННЯ ІНФОРМАТИКИ ТА КОМП'ЮТЕРНОЇ ТЕХНІКИ В ЕКОНОМІЧНОМУ ВУЗІ

М.С. Головань

м. Суми, Українська академія банківської справи  
golovan@academy.sumy.ua

Ідея навчання за методом проектів не є принципово новою в педагогіці. Вона була запропонована практично паралельно на початку ХХ століття російським педагогом С.Т. Швацьким та американським – О. Паркхерст з метою індивідуалізації процесу навчання. Ця форма навчання використовувалася в Радянському Союзі в 20-30-ті роки ХХ століття під назвою “дальтон-план.” Бурхливий розвиток нових інформаційних технологій сприяв відродженню цієї форми навчання в теперішній час.

Основою методу проектів є розвиток пізнавальних умінь та навичок, умінь самостійного конструювання знань, умінь орієнтуватися в інформаційному просторі, розвиток критичного мислення.

Метод проектів передбачає самостійну пізнавальну діяльність студентів – індивідуальну, парну, групову, яку вони виконують протягом певного відрізка часу. Метод проектів завжди припускає розв'язання деякої проблеми, яка передбачає, з одного боку, використання різноманітних методів та засобів навчання, а з іншого – інтегрування знань, умінь і навичок з різних галузей науки, техніки, технології.

Метод проектів надає можливості для:

- 1) формування дослідницьких умінь збирати необхідну інформацію, висувати гіпотезу, робити висновки і умовиводи, використовувати для роботи нові інформаційні технології;
- 2) розвитку самостійного мислення і діяльності, системного підходу у самоорганізації, груповій взаємодії;
- 3) підвищення мотивації навчання та самонавчання: актуалізація незатребуваних знань і стимуляція надбання нових знань;
- 4) формування особистісних якостей студентів: взаєморозуміння, взаємоповаги, відповідальності;

- 5) організації творчої діяльності студентів;
- 6) структурування знань шляхом встановлення міжпредметних зв'язків;
- 7) вироблення навичок ефективного використання комп'ютера в своїй подальшій професійній діяльності;
- 8) формування проектної культури (знань і умінь визначення потреб і можливостей діяльності при виконанні проекту, вибір оптимальної ідеї та її дослідження, планування, організація і виконання роботи з реалізації проекту, оцінка і його презентація);
- 9) формування інформаційної культури (знання, уміння і навички використання інформації з різних джерел).

У навчанні інформатики метод проектів використовується в двох напрямках: телекомунікаційні проекти та проекти створення інформаційних моделей реальних систем..

Під проектом розуміють чітко поставлену задачу, яка має кінцевий результат, практичну значущість і структурний зміст.

Виконання проекту в курсі інформатики та комп'ютерної техніки переслідує наступні цілі:

- розвиток навичок самостійної роботи;
- розвиток умінь і навичок комплексного використання комп'ютера для розв'язання прикладної задачі;
- розвиток дослідницьких навичок і навичок аналізу емпіричних даних;
- закріплення навичок роботи в MS Word, Excel, Access, PowerPoint і обміну інформацією між ними;
- можливість об'єктивної оцінки знань студентів.

Для студентів економічних вузів пропонуються завдання розробити проект з певної предметної галузі економіки з використанням програм пакету Microsoft Office. Виконання такого завдання дозволить студентам на практиці набути навичок використання сучасних технологій створення комбінованих документів. Завдання у проекті сформульовані у термінах предметної галузі, тому спочатку студенти повинні проаналізувати предметну галузь, створити інформаційну модель, розробити алгоритм реалізації моделі за допомогою відповідного програмного засобу.

Структурність змістової частини проекту передбачає можливість виділення його етапів та підзадач, які у більшості випад-

ків розв'язуються за допомогою різноманітних формальних мов описування об'єктів і процесів та різного програмного забезпечення.

Прикладом навчального проекту може бути наступний: створити програмний засіб для проведення аналізу надійності банку в 1999, 2000 та 2001 році за допомогою відповідних коефіцієнтів. Відобразити на діаграмі динаміку надійності банку. Процес створення програмного засобу описати засобами MS Word у вигляді інструкції. Доповідь оформити у вигляді презентації PowerPoint.

Оскільки студенти ще не вивчали фахових дисциплін, тому постановка задачі і її математична модель формулюється викладачем. Наводиться загальна формула надійності банку, формули для обчислення коефіцієнтів, що входять у загальну формулу та тлумачення параметрів, що зустрічаються у формулах.

Робота над проектом та його описова частина здійснюється згідно наступного плану:

1. Аналіз поставленої задачі.
  - 1.1 Постановка задачі.
  - 1.2 Математична модель задачі.
2. Проектування програмного засобу.
  - 2.1 Проектування книги MS Excel.
  - 2.2 Проектування розміщення інформації на аркушах книги MS Excel.
3. Реалізація програмного засобу на комп'ютері.
  - 3.1 Створення книги MS Excel.
  - 3.2 Ведення інформації.
  - 3.3 Форматування аркуша.
  - 3.4 Виконання чисельних розрахунків.
  - 3.5 Побудова діаграм.
4. Аналіз результатів.
5. Створення презентації засобами PowerPoint.
  - 5.1 Планування презентації.
  - 5.2 Розробка елементів презентації.
  - 5.3 Програмна реалізація.
  - 5.4 Тестування презентації.
  - 5.5 Демонстрація.
6. Звіт про виконану роботу.

Опишемо коротко зміст дій студентів згідно наведеного плану.

У п.1 наводиться формулювання задачі, та її математична модель, назви величин, їх позначення та розрахункові формули. У п.2 вказується назва файлу, що містить книгу, кількість аркушів у книзі, їх назви та призначення. На робочих аркушах створюються макети необхідних таблиць з назвами заголовків рядків та стовпців, тут же записуються формули з використанням посилань на клітини.

У п.3 послідовно виконуються дії щодо створення нової книги, присвоєнню імен книзі та її аркушам, вводиться в таблиці необхідні числові та текстові дані, а також формули, використовуючи прийоми введення інформації (копіювання, авто заповнення, авто сумування, майстер функцій та ін). Після введення інформації виконується відповідне форматування електронної таблиці. Після цього здійснюється побудова необхідних діаграм.

У п.4 проводиться аналіз отриманих результатів розв'язання поставленої задачі.

У п.5 здійснюється розробка презентації звіту про виконану роботу. У процесі роботи над презентацією студент повинен продумати яка інформація буде розміщена в презентації, розробити навігаційну структуру, дизайн слайдів, підготувати текстовий та ілюстративний матеріал для наповнення слайдів, власне сформулювати слайди, перевірити коректність роботи презентації та продемонструвати створену презентацію товаришам.

Іншими прикладами проектів можуть бути наступні: “Розподіл прибутку підприємства або банку”, “Оцінка кредитоздатності позичальника”, “Визначення ліквідності банку”, “Розрахунок лізингових платежів”, “Прогнозування динаміки курсу долара”, “Економічна оцінка ефективності фінансування інвестиційних проектів” тощо.

Основою навчального проекту може бути також розробка бази даних засобами Microsoft Access. Тут студенти на основі аналізу предметної галузі повинні створити необхідні таблиці бази даних (3-4 таблиці), правильно розподілити дані між ними, визначити ключові поля, встановити зв'язки між таблицями, розробити необхідні форми для введення і модифікації даних, розробити запити на вибірку, на обчислення над даними у полях,

на отримання підсумкових значень, на модифікацію бази даних (оновлення, додання та вилучення записів), розробити звіти на основі запитів з параметрами, розробити головне меню у вигляді кнопкової форми. У текстовому процесорі створити опис проекту, а також розробити презентацію до проекту.

Приступаючи до виконання проекту студенти повинні бути ознайомлені з основними прийомами роботи з відповідними додатками Microsoft Office. Кожний із студентів отримує індивідуальне завдання з постановкою задачі та її математичною моделлю, з планом виконання проекту і списком необхідної літератури. Студенти також знайомляться з прикладом виконаної роботи. У всьому іншому робота студента над проектом здійснюється самостійно.

Завдання на створення проекту видається студентам на момент вивчення MS Excel або MS Access. Робота над проектом триває протягом 3-4 місяців і окремі його етапи виконуються на лабораторних заняттях, після того як вироблені достатні навички, необхідні для створення відповідного етапу проекту. Таким чином, робота студентів над проектом чергується з традиційним практичним вивченням відповідного матеріалу курсу інформатики та комп'ютерної техніки.

Зауважимо, що навчальні припускають відхід від авторитарних методів навчання, проте, передбачають ретельно продумане і обгрунтоване поєднання різноманітних методів, форм і засобів навчання. У процесі роботи над проектом студенти виступають як суб'єкти навчання, викладачі виступають консультантами проектів. При цьому студенти набувають навичок здобувати знання самостійно, а також уміння працювати в різних групах, виконуючи різні соціальні ролі (лідера, виконавця, посередника тощо).

## ФОРМУВАННЯ ЯКІСНИХ ЗНАТЬ УЧНІВ З ВИКОРИСТАННЯМ ЕКСПЕРТНИХ СИСТЕМ

М.С. Головань<sup>1</sup>, М.О. Антонченко<sup>2</sup>

<sup>1</sup> м. Суми, Українська академія банківської справи

<sup>2</sup> м. Суми, Сумський державний педагогічний університет ім.  
А.С. Макаренка

Сучасна школа України переживає процес перетворення, розробки нових концепцій навчання. Ці процеси обумовлені переходом нашої держави від індустріального до інформаційного суспільства, в якому вирішальним чинником, що визначає розвиток суспільства в цілому, є інформація та її вища форма – знання. В умовах постійно зростаючого обсягу інформації школа покликана забезпечити учнів цілісними знаннями, що відбивають сучасний рівень науки і культури, навчити використовувати інформаційні технології для вдосконалення своєї практичної діяльності та орієнтації в інформаційному просторі. Так, у Державній національній програмі “Освіта” (Україна ХХІ століття) підкреслюється, що одним із шляхів реформування освіти є широке використання сучасних інформаційних технологій з метою виховання творчої, самостійної особистості, всебічного розвитку здібностей учнів. У цих умовах доцільною є розробка питань формування якісних знань, що складають основу цілісного наукового світогляду, творчості суб’єктів навчання.

На даний час у педагогіці та психології накопичений досить вагомий досвід з проблеми формування якості знань учнів (Ю.К. Бабанський, П.Я. Гальперін, Л.В. Занков, Л.Я. Зоріна, В.В. Краєвський, І.Я. Лернер, С.Л. Рубінштейн, М.М. Скаткін та ін.). Проте, ця проблема розглядалась недостатньо в умовах широкого використання комп’ютера як засобу навчання.

У сучасній психолого-педагогічній літературі можна виділити декілька напрямків, за якими здійснювалося вивчення проблеми формування якостей знань учнів:

- філософський аналіз категорії “знання” (В.І. Генєцинський, В.В. Орлов, В.А. Чечнев та ін.);
- вивчення ролі діяльності і реалізація діяльнісного підходу в процесі формування якісних знань учнів (Є.П. Бруновт,



В.К. Буряк, В.В. Давидов, П.Я. Гальперін, Є.М. Кабанова-Меллер, Н.Ф. Тализіна, І.Ф. Харламов, Г.І. Щукіна та ін.);

- визначення особливостей формування знань при використанні різноманітних педагогічних технологій (О.С. Андрієнко, В.П. Безпалько, В.І. Євдокимов, М.В. Кларін, І.Ф. Прокопенко та ін.);
- характеристика окремих якостей знань, умов та засобів їх формування (І.Я. Бесонова, А.Н. Зв'ягін, Л.Я. Зоріна, А.В. Усова, Л.А. Федченко та ін.);
- розгляд основних аспектів формування цілісної системи якостей знань (Т.Л. Коган, Є.А. Красновський, І.І. Кулібаба, І.Я. Лернер, М.М. Скаткін, Т.І. Шамова та ін.);
- застосування нових інформаційних технологій навчання для формування якісних знань учнів (Н.В. Апатова, Л.І. Білоусова, Б.С. Гершунський, Ю.В. Горошко, М.І. Жалдак, Ю.І. Машбиць, Н.Ф. Тализіна, С.А. Раков та ін.).

У той же час, у педагогічній науці залишаються недостатньо розглянутими проблеми теоретичного обґрунтування і впровадження технології організації навчально-пізнавальної діяльності учнів в умовах інформатизації освіти, зокрема, з використанням засобів нових інформаційних технологій навчання. Ці засоби розвиваються за двома основними напрямками:

- удосконалення якості навчальних та тренувальних комп'ютерних програм, призначених для передачі певного обсягу знань і для набуття визначеного набору вмій та навичок;
- створення принципово нових інтелектуальних навчальних систем, що мають різні способи подання знань і відповідають сучасним вимогам інформаційного суспільства – практично необмеженого поповнення знань.

У зв'язку з реалізацією другого напрямку актуальним стає оволодіння інтелектуальними засобами інформаційних технологій, зокрема, експертними системами як для засвоєння знань, так і для напрацювання вмій і навичок дослідження інформації, набуття нових знань, використання нових засобів і методів пізнавальної діяльності та формування світогляду.

У науковій літературі виділяються такі аспекти розгляду

експертних систем:

- експертні системи як один із напрямків досліджень у галузі штучного інтелекту (О.О. Молокова, К. Нейлор, С. Осуга, Е.В. Попов, Д. Уотерман, Р. Форсайт та ін.);
- дослідження психолого-педагогічних аспектів використання експертних систем у процесі навчання (Н.Р. Балик, В. Буцик, С.А. Раков, Ю.С. Рамський, Ю.А. Шрейдер та ін.).

Аналіз досвіду навчального використання експертних систем в Україні та за кордоном дозволяє зробити висновок, що вони є перспективним сучасним дидактичним засобом, проте в практиці навчання (зокрема, природничих дисциплін) ці системи використовуються мало, що в деякій мірі обумовлено недостатньою розробленістю педагогічних і методичних аспектів їх упровадження у навчальний процес.

Процес систематичного наукового пізнання навколишнього світу сприяє накопиченню інформації у вигляді різних форм знання (фактів, наукових теорій тощо). Змістовна навчальна інформація стає результатом відбору матеріалу, що повинен завоювати учень. Вона включає основні наукові положення й актуальну поточну інформацію, яка в сучасних умовах наближує зміст навчання до потреб суспільства й особистості.

Проте обсяг засвоєної змістовно-навчальної інформації не є головним критерієм при визначенні рівня досягнень учнів у навчанні і такі знання ще не можна вважати якісними. Повноцінні знання учнів можуть сформуватися лише в результаті усвідомлення зв'язків між елементами знань на основі оволодіння таким змістом загальної середньої освіти, який включає досвід творчої діяльності і закладає фундамент цілісного світосприйняття.

Сучасні уявлення про якісні знання передбачають також сформованість певних компетенцій (соціальних, полікультурних, комунікативних, інформаційних, саморозвитку та самоосвіти) і таких якостей особистості, як ініціативність та творчість. Прагнення до саморозвитку та самоосвіти, творча ініціатива, наполегливість притаманні тільки тим учням, у яких достатньо розвинені пізнавальні інтереси. Соціальні, полікультурні, комунікативні компетенції, психоемоційний досвід і ставлення до себе, оточуючого світу можна віднести до світоглядних позицій учнів. Вимога наявності інформаційних компетенцій обумовлена

зростанням ролі інформації в сучасному суспільстві. Ці компетенції включають уміння застосовувати інформаційні технології в практичній діяльності, відшукувати, критично осмислювати, опрацьовувати і використовувати різноманітну інформацію. Вони виявляють себе через зміну характеру та способів практичної діяльності учнів.

З огляду на вище зазначене можна стверджувати, що учні оволодівають якісними знаннями лише тоді, коли вони не тільки засвоїли певний обсяг змістовної навчальної інформації, але й у них сформовані цілісна сукупність характеристик, які описують результат засвоєння знань (система якостей знань), пізнавальні інтереси та світоглядні позиції.

Отже, якісні знання учнів – це інтегративний результат засвоєння учнями змістовної навчальної інформації, відмінними рисами якого є: усвідомлення учнями зв'язків між елементами знань; уміння відтворювати елементи знань залежно від необхідності їх використання в практичній діяльності; сформованість в учнів пізнавальних інтересів та основ наукового світогляду.

Для формування якісних знань учнів експертні системи мають значні дидактичні можливості [1]. Під час навчання експертні системи утворюють з учнем єдине функціональне предметно-орієнтоване середовище. За допомогою експертних систем учень здійснює аналіз окремих аспектів змісту знань, структурує їх, порівнює з тими знаннями, що уже містяться у базі знань, використовує їх для організації практичної діяльності, формуючи таким чином системність, дієвість і міцність знань на більш високому рівні.

Особливістю експертної системи є те, що в її основі лежить база знань, яка в структурованому вигляді містить знання з певної галузі знань. За допомогою експертної системи можна організувати спеціальну навчально-пізнавальну діяльність учнів з предметів природничого циклу, зокрема з біології та хімії. До такого виду діяльності віднесено:

- інформаційно-пошукову діяльність, що розширює обсяг знань про об'єкти, які вивчаються;
- діяльність з вивчення логічних кроків роботи експертної системи з базою знань, що сприяє розвитку вмінь послідовно застосовувати потрібні знання для прийняття рішень;

- діяльність з розробки інформаційних моделей навчального матеріалу, що формує вміння виявляти внутрішні зв'язки, ієрархічні рівні елементів знань, класифікувати, структурувати та систематизувати навчальну інформацію;
- діяльність по заповненню експертної оболонки, в результаті чого учень здійснює ряд розумових дій (аналіз, синтез, порівняння, класифікація та ін.), розвиток яких сприяє формуванню якісних знань.

Оскільки знання, які опановує учень, набувають необхідних якостей у процесі діяльності, тому нами визначені показники, за якими можна судити не лише про зміст засвоєних учнями знань, але й про процес їх набуття. Показники, які характеризують змістовний аспект знань з предметів природничого циклу, відбивають ступінь відтворення учнями фактів, понять, подій, різноманітних зв'язків між ними.

Процесуальний аспект якості знань відбивається у показниках, які характеризують рівень розвитку інтелектуальної, мотиваційної та емоційно-вольової сфер особистості.

Аналіз різних характеристик знань, які розглядаються в педагогічній літературі, на основі інтегративної ознаки дозволив виділити ті з них, які є найбільш суттєвими для визначення якісних знань учнів: системність, дієвість і міцність [2].

Оцінка сформованості системи якостей знань здійснюється через оцінку показників, які характеризують інтегративні якості системності, дієвості та міцності знань. Для кожного з інтегративних компонентів системи якостей знань в [3] запропоновано набір показників, які характеризують їх рівень сформованості.

Так, показниками аналізу системності знань є вміння виділяти головне і другорядне, вміння підводити об'єкт під поняття на основі ознак, вміння встановлювати причинно-наслідкові і родовидові зв'язки, вміння структурувати навчальний матеріал за різними основами.

Показниками аналізу дієвості знань є вміння вибирати засоби (правила, закони, необхідне програмне забезпечення, довідкову літературу тощо) для розв'язання навчального завдання, вміння будувати модель навчальної теми, вміння аналізувати отримані результати і відповіді інших учнів.

Одиницею аналізу міцності знань учнів є вміння відтворю-

вати навчальний матеріал через деякий час.

Звичайно, ці показники відкриті для подальшого доповнення й удосконалення, але, на нашу думку, вони містять у собі визначальні ключові моменти, за якими опосередковано можна оцінювати сформованість системи якостей знань учнів.

На основі виділених показників нами визначені та охарактеризовані такі рівні сформованості якості знань учнів з предметів природничо-математичного циклу: низький, середній, та високий.

Низький рівень передбачає репродукцію знань і способів діяльності. На цьому рівні учень користується експертною системою як інформаційно-пошуковою, використовуючи знання, що містяться в базі знань для розв'язання поставлених навчальних задач. Поставлені учителем завдання розв'язуються учнем на відтворюючому рівні за зразком або з постійною допомогою учителя.

Середній рівень характеризується вміннями учня перетворювати знання. Перед заповненням бази знань експертної оболонки учень повинен здійснити аналітико-синтетичну діяльність по структуруванню навчального матеріалу. Ця діяльність на даному рівні виконується під керівництвом і незначною допомогою вчителя.

На високому рівні учень опановує нові способи і прийоми дій. На цьому рівні робота з експертною системою набуває більш самостійного характеру, здійснюється така складна діяльність, як побудова інформаційної моделі навчальної теми, учні здійснюють прогнозування як результатів, так і способів діяльності.

На основі аналізу досвіду використання експертних систем у процесі навчання предметів природничого циклу було виявлено дидактичні умови застосування експертних систем для формування системи якостей знань учнів. Ці умови можна розподілити на чотири групи.

Перша група умов пов'язана з вибором експертних систем:

- вибір експертної системи або експертної оболонки слід здійснювати відповідно до структури навчального предмету і дидактичної мети уроку;
- експертні системи повинні задовольняти наявним технічним можливостям комп'ютера, вимогам концептуальної

зрозумілості, доступності, зручності в роботі, надійності.

Друга група умов, стосується підготовки вчителів:

- зацікавленість учителів у пошуку нових підходів до вивчення навчального матеріалу;
- зацікавленість учителів у використанні нових інформаційних технологій;
- достатній рівень володіння засобами НІТ.

Третя група умов, пов'язана з підготовкою учнів:

- наявність в учнів позитивної мотивації використання ЕС у процесі вивчення предмету;
- проведення пропедевтичної роботи з учнями щодо ознайомлення з ЕС (знайомство з інтерфейсом, механізмом логічного виводу, структурою бази знань ЕС);
- доступність знань про експертні системи;
- послідовності і поступовості роботи з ЕС (спочатку ознайомлення з експертними системами, потім заповнення бази знань ЕС під керівництвом учителя, після цього самостійна робота учнів з експертною системою).

Четверту групу складають умови, які стосуються методів та організаційних форм навчання:

- переважне застосування активних методів навчання;
- вибір і раціональне поєднання методів і організаційних форм навчання залежить від складності матеріалу, що вивчається, етапу дидактичного циклу, рівня підготовленості учнів і спирається на диференціацію й індивідуалізацію навчання.

У ході експериментальної оцінки сформованості якісних знань учнів 7-8 класів у процесі навчання хімії та біології встановлено, що високого рівня сформованості якісних знань на основі вивчення матеріалу з використанням експертних систем досягли 15% учнів в експериментальних класах ( в контрольних – 9% учнів), середнього рівня досягли 55% і відповідно 46% учнів і низького рівня – 30% і 45% відповідно.

Таким чином, експериментально підтверджено вплив використання експертних систем на підвищення рівня якості знань учнів.

## Література

1. Антонченко М.А., Головань Н.С. Экспертные оболочки как средство повышения уровня сформированности системы качеств знаний учащихся // Педагогічні науки. Збірник наукових праць. – Суми: СДП, 1998. – С. 106-110.
2. Антонченко М.О. Означення поняття “система якостей знань учнів” // Педагогіка та психологія: Збірник наукових праць ХДПУ. – Харків: Медіа Група, 2000. – Вип. 14. – С. 22-26.
3. Антонченко М.О. Кваліметрія поняття “система якостей знань учнів” // Педагогічні науки. Збірник наукових праць. – Суми: СДПУ ім. А.С. Макаренка, 2000. – С. 345-350.

## МЕТОДИКА НАВЧАННЯ ОСНОВАМ ІНФОРМАТИКИ І ПРОГРАМУВАННЯ У VBA

Л.П. Голубєв, Ю.М. Пилипенко, Ю.П. Юрачківський  
м. Київ, Київський національний університет технологій і дизайну

### *Лабораторна робота №1.*

*Тема: “Програмування мовою VBA. Макроси в Excel. Редактор VBA. Алгоритми з розгалуженням”.*

**Ціль:** Навчити студентів записувати макроси за допомогою макрорекордера. Навчити працювати з редактором **VBA**. Функція **InputBox**, оператор **MsgBox**. Робота з кодами, що містять умовні оператори. Редагування коду і запуск його на виконання.

#### **Завдання 1.**

1. Створіть свій перший макрос у “Цій книзі” з ім'ям “**Проба**”, який би в активному осередку **Excel** (у шрифті **Arial**, кегль **14**, курсив) друкував текст: “**Студент – прізвище, ім'я**” (прізвище, ім'я – виконавця).

2. Призначте макрос

- а) комбінацією гарячих клавіш **Ctrl+F**;
- б) створеної на робочому листі кнопці; і переконаєтеся, що макрос правильно працює.

#### **Завдання 2.**

1. Створіть блок-схему алгоритму запропонованої Вам задачі.

2. Прорахуйте контрольний приклад.

3. Складіть код (програму) на **VBA**, який би реалізовував створений Вами алгоритм.

4. Використовуючи команди: **Сервіс** ⇒ **Макрос** ⇒ **Почати запис**, створіть новий макрос у “Цій книзі” з назвою “**Розгалуження**”, що **крім назви** нічого не містить.

5. Використовуючи команди: **Сервіс** ⇒ **Макрос** ⇒ **Макроси** (чи комбінацію клавіш **ALT+F8**) знайдіть потрібний макрос “**Розгалуження**” і, виділивши його, виберіть команду “**Змінити**”. При цьому відкриється вікно редактора **VBA**, куди Ви будете вводити **підготовлений Вами заздалегідь** код. Відзначимо, що доступ до редактора **VBA** можна здійснити і так:



а) **ALT+F11**;

б) **Сервіс** ⇒ **Макрос** ⇒ **Редактор VBA**.

Введення інформації здійснити, використовуючи функцію **InputBox**, а виведення інформації, використовуючи вікно повідомлень **MsgBox** по зразку: “Значення функції в точці  $x = \dots$  дорівнює  $y = \dots$ ”

6. Навчитися запускати підготовлений Вами макрос:

а) знаходячись у редакторі **VBA**;

б) з робочого листа **Excel**, використовуючи: **Сервіс** ⇒ **Макрос** ⇒ **Макроси** ⇒ **Виконати**.

7. Скопіюйте текст Вашого коду в документ **MSWord** і роздрукуйте його.

8. Збережете робочу книгу і текст коду, з яким Ви працюєте на Вашій дискеті в папці **VBA**, під ім'ям **VBA1.xls**, **VBA1.doc** відповідно, попередньо зберігши файли на диску **C:** у відведеному Вам місці.

**Звіт по роботі повинний містити:**

1. Постановку задачі .

2. Блок-схему алгоритму і контрольний приклад.

3. Роздруківку коду макросу “Розгалуження”.

4. Дискету з підготовленими файлами.

## **Лабораторна робота №2**

**Тема: “Програмування мовою VBA. Циклічні алгоритми”**

**Ціль:** Навчити студентів працювати з циклами типу **For... Next, Do... Loop**.

**Завдання .:**

1. Створіть блок-схеми алгоритмів рішення двох задач:

а) підрахунку суми (добутку) із заданим числом доданків (множників) типу

$$F = \sum_{i=l}^k \frac{i-3}{i^2 - 2i - 3}.$$

У таких задачах доцільне застосування циклу **For... Next**, де параметри  $l$  і  $k$  вводяться вручну в ході виконання програми. При цьому невизначені доданки (множники) повинні бути відкинуті.

б) знаходження суми членів ряду, закінчуючи підсумовування, коли ***n-ий*** член ряду стане менше (по модулю) наперед зада-

ній точності обчислень (число  $\epsilon$ ).

Відзначимо, що при підрахунках для збіжності ряду необхідно брати аргумент  $x$ , виходячи з того, що  $|x| < 1$ . Хоча дане обмеження і не стосується абсолютно усіх варіантів, для контрольного приклада візьміть  $x$  з умови  $0 < |x| < 1$ .

У задачах другого типу доцільне використання циклу типу **Do... Loop**.

Числа  $x$  і  $\epsilon$  ( $\epsilon > 0$ ) вводяться вручну, у ході виконання програми.

2. Підрахуйте контрольні приклади, обмеживши 4–5 доданками (при підрахунку добутку **не** повинне бути нульового множника серед співмножників, розглянутих у конкретній задачі).

3. Створіть коди на **VBA**, що реалізують Ваші алгоритми.

Перший макрос назвати “**Арифм – цикл**”,

другий – “**Ітерац – цикл**”.

4. При написанні коду в коментарях **обов’язково** вказати:

а) прізвище і групу виконавця;

б) ділянки, де йде введення-виведення інформації.

5. Ділянки коду між ключовими словами **For... Next**, **Do... Loop** зобов’язані бути зрушені вправо стосовно початку і кінця циклу !

6. Введення-виведення інформації здійснити, використовуючи функцію **InputBox** і оператор **MsgBox** відповідно.

7. **Обов’язково** протестуйте Вашу програму, звіривши дані з результатами контрольного прикладу.

8. Скопіюйте тексти кодів у документ **MSWord** і роздрукуйте його.

9. Зберегти отримані файли на Вашій дискеті в папці **VBA** під ім’ям “**VBA-2**” з розширенням **xls** і **.doc** (відповідно робочу книгу і документ **MSWord**).

**Звіт по роботі повинний містити:**

1. Постановку задачі (указати номер задач).

2. Блок-схеми алгоритмів і контрольні приклади.

3. Роздруківку коду.

4. Дискету з підготовленими Вами файлами.

*Лабораторна робота № 3*

*Тема: “Програмування мовою VBA. Підпрограми і функції”*

*користувача”*

**Ціль:** навчити студентів створювати і використовувати власні підпрограми і функції; навчити оформляти підпрограми і функції як окремі модулі, для створення власної бібліотеки підпрограм.

**Завдання:**

Метод Ньютона призначений для знаходження кореня рівнянь виду  $f(x) = 0$ , де  $f(x)$  – диференційована функція і  $d(x)$  – її похідна. Метод полягає в наступному. Вибирається початкове наближення кореня  $x_0$ . Для полегшення знаходження наближеного значення кореня  $x_0$  рівняння  $f(x) = 0$ , рівняння часто доцільно представити у виді  $g(x)=t(x)$ , побудувати графіки лівої і правої частин і знайти абсцису точки, у якій графіки перетинаються. Потім по формулі  $x_{k+1} = x_k - f(x_k) / d(x_k)$  для  $k = 0, 1, 2, \dots$ , по попередньому ( $k$ -ому) наближенню обчислюється наступне ( $(k+1)$ -е). Процедура виконується доти, поки  $|x_{k+1} - x_k| > \epsilon$ , де  $\epsilon$  – деяке досить мале дійсне число, наприклад, 0,0001.

Завдання полягає в тому, щоб:

1. По заданій функції  $f(x)$  обчислити функцію  $d(x)$  (без допомоги комп'ютера).

2. Написати мовою **VBA** програму визначення кореня рівняння  $f(x) = 0$  методом Ньютона. Програм повинна складатися з підпрограми (**Sub**), що реалізує власне метод Ньютона і двох функцій (**Function**), що повертають значення функцій  $f(x)$  і  $d(x)$  відповідно. Підпрограму (**Sub**) і функції (**Function**) оформити як окремі **модулі**, зробивши їх видимими у всьому проекті. У модулях, у розділі GENERAL DECLARATIONS, **обов'язково** введіть оператор OPTION EXPLICIT.

3. Вивести на екран у вікно повідомлень (**MsgBox**) повідомлення: Корінь рівняння дорівнює = [значення знайденого кореня].

**Звіт по роботі повинний містити:**

1. Постановку задачі.
2. Представлення функції  $f(x)$  у виді  $g(x)=t(x)$  і побудова графіків функцій лівої і правої частин. Запишіть у зошиті, яке початкове наближення кореня Ви знайшли.
3. Блок-схему програми і підпрограм.
4. Результат, виданий комп'ютером.

5. Підклеєну в зошит роздруківку програмного коду.
6. Намальоване в зошиті (від руки) дерево вікна Вашого проекту.

#### **Лабораторна робота №4.**

**Тема: “Програмування мовою VBA. Знайомство з властивостями і методами основних об’єктів VBA(Excel): Application, WorkBook, WorkSheet, Range.”**

#### **Завдання.**

Протабулювати функцію, значення якої обчислювалося в лабораторній роботі №1.

Для цього необхідно:

1. Скопіювати модуль, у якому обчислювалася функція в споруджуваній проект (використовуйте **Project Explorer**).

2. Оформити функцію як **Public Function**, зробивши відповідні присвоєння типу:

<ім’я функції> = <значення>

і відкинувши, тепер уже непотрібні, повідомлення (**MsgBox**) про значення функції. Зверніть увагу, що необхідно в заголовку функції ввести формальний аргумент, а також на те, що тип функції, що **обчислюється**, Variant, тому що вона може приймати як речовинні значення, так значення виду: “функція невизначена”, (тип **String**).

Відзначимо, що при необхідності (тобто коли відповідної функції, що бере участь в обчисленнях, немає серед стандартних функцій **VBA**) **обов’язково** використовуйте функції робочого листа.

**Табуляцію провести в підпрограмі “Табуляція”, використовуючи побудовану функцію, а результати табуляції розмістити на робочому листі.**

Адреса клітини, з якої починається табуляція, ввести за допомогою функції **InputBox**. У цю клітину ввести заголовок споруджуваної таблиці – “табуляція” (шрифт “**Times New Roman**”, розмір **14**, напівжирний).

Під заголовком йдуть клітини з назвами:

початок	кінець	число інтервалів
---------	--------	------------------

Під ними, у наступній рядку, розмістити інформацію про початок інтервалу (**a**), кінець інтервалу (**b**), кількості інтервалів

(k), на які буде розбитий інтервал табулювання.

Числа **a**, **b**, **k** увести за допомогою функції **InputBox**, організувавши перевірку того, що вводиться не порожній рядок, оформивши введення-перевірку перемінної і виді окремої підпрограми.

Після введеної інформації в наступному рядку йдуть назви:

аргумент	функція
----------	---------

під якими власне і будуть розміщені значення аргументу і відповідні їм значення функції.

Лист, на якому розміщається інформація, назвати “табуляція”.

У споруджуваному модулі обов’язкова наявність оператора **OPTION EXPLICIT**.

Після всієї підготовленої інформації уведіть своє прізвище і групу. Збережете файл у робочій книзі з ім’ям **VBA4.xls**.

**Звіт по роботі повинний містити:**

1. Назва роботи.
2. Постановку задачі.
3. Підклеєні роздруківки коду програми і робочого листа з табуляцією в конкретному інтервалі. Інтервал табуляції і крок вибрати так, щоб пропраховувалися значення по кожній гілці алгоритму.

### **Лабораторна робота №5.**

**Тема: “Програмування мовою VBA. Робота з масивами.”**

**Ціль:** Навчити студентів, використовуючи **VBA (Excel)**, працювати з одномірними і двовимірними масивами. Повторити матеріал по перетворенню робочих аркушів (**Excel**).

**Завдання.**

1. Використовуючи можливості прикладної програми “**Excel**”, здійснити наступні дії:

а) перейменувати “**Лист1**” у “**w1**”, “**Лист2**” у “**w2**”, “**Лист3**” у “**rez**”;

б) Лист “**w1**” заповнити (заповнення, прогресія):

перший стовпець числами від 1 до 10

перші десять стовпців першого рядка – прогресією з кроком

2.

в) Нехай **i** – номер рядка, **j** – номер стовпця.

Помістити (копіюванням) в клітини діапазону **B2 : J10** формулу **7cos(i) ln(j)**. Діапазон **A1 : J10** представити в числовому форматі з двома знаками після коми.

Вказівка. Використовуйте функції **СТОВПЕЦЬ**, **РЯДОК** категорії “посилання і масиви”.

г) ввести аналогічну інформацію в лист “**w2**” з тією різницею, що замість формули пункту в) в клітинах повинна бути наступна формула: “якщо **i + j** ділиться на 3, то вносимо **j – i**, у протилежному випадку **i<sup>2</sup> – 2j**.”. Діапазон **A1:J10** представити в числовому форматі без знаків після десяткової крапки.

Вказівка. Використовуйте математичну функцію **ОСТАТ**.

2 Написати програму на **VBA**, що виконує наступні дії:

а) об’являє всі змінні (**OPTION EXPLICIT**);

б) об’являє динамічні: одномірний цілочисленний масив **C**, і дійсні масиви **B** і **D**;

в) запитує, з якого листа зчитувати інформацію;

г) по черзі зчитує з діалогових вікон з повідомленням “Уведіть число рядків **N** (стовпців **M**) масиву **A**” числа **N** і **M** доти, поки вони менше 2 чи більше 6;

д) об’являє, що масив **B** розміру **M**, масив **C** розміру **N**, масив **D** розміру **N**;

е) присвоює елементам **A(1,1), ... A(N,M)** вміст клітин **(2,2), ..., (N+1, M+1)** відповідного робочого листа;

ж) присвоює елементам **B(1), ..., B(M)** вміст *клітин* **(2, M+2), ..., (M+1, M+2)** відповідного робочого листа.

Далі програма виконує наступні дії:

а) обчислює масив **C** як вектор, що є добутком матриці **A** на вектор **B**. Нагадуємо, що **i**-тий елемент вектора **C** (**i=1,2,...,N**) обчислюється по формулі:

$$C_i = \sum_{j=1}^M A_{ij} B_{ij}.$$

б) перетворить масив **C** згідно зі своїм варіантом. Підпрограму перетворення масиву оформити у виді окремої підпрограми.

в) виводить результати (масиви **A**, **B**, **C**, **D**) на лист “**rez**”. При цьому висновок одномірного масиву оформити у виді окремої підпрограми, що викликається в потрібному місці головної програми. Зверніть увагу на те, що в підпрограмі необхідні па-

раметри, що вказують, з якого осередку робочого листа буде йти заповнення потрібного діапазону і який заголовок повинний стояти над відповідним масивом (над кожним з виведених масивів повинне бути назва “масив А”, “масив В”, “масив С”, “масив D” відповідно).

При налагодженні програм варто навчитися встановлювати крапки зупинки, спостерігати за перемінними, здійснювати покрокове виконання програм!

***Звіт по роботі повинний містити:***

1. Блок-схему перебування масиву  $C_N$  по наявних масивах  $A_{M \times N}$ ,  $B_M$ .
2. Постановку задачі по перетворенню масиву  $C$  и блок-схему її реалізації.
3. Роздруківку всіх програм і листа “rez”.
4. Збережений на диску **A**: файл **VBA.5.xls** з Вашою роботою.

### ***Лабораторна робота № 6***

***Тема: “Програмування мовою VBA Середовище розробки VBA – програм”***

***Ціль:*** Вивчити основні компоненти середовища розробки VBA–програм.

- 1) Освоїти головне меню системи, ніспадаюче підміню, стандартну панель інструментів.
- 2) Навчитися працювати з вікном інспектора проекту, вікном властивостей, вікном інструментів.
- 3) Навчитися працювати з формою, змінювати її властивості.
- 4) Навчитися розміщати на формі елементи керування і змінювати їхньої властивості.
- 5) Навчитися працювати з програмним кодом.
- 6) Навчитися запускати програму на рахунок.
- 7) Навчитися налагоджувати програму.
- 8) Навчитися зберігати проект.
- 9) Навчитися викликати збережений проект.

***Завдання:***

Створити і зберегти VBA–проект, що виконує операцію конвертації гривні в USD. (Приклад розглядається на лекції).

## Лабораторна робота № 7

### Тема : “Програмування мовою VBA. Ітераційні цикли”

**Ціль:** познайомити з новими елементами керування **ScrollBar, ComboBox**, новими подіями **Change, Scroll**.

#### Завдання.

Створити **VBA-проект** обчислення функції **Y=f(X)** як суми членів із заданою точністю. Кожен доданок суми (член ряду) обчислюється по рекурентному співвідношенню  $U_{n+1}=F(U_n, N, X)$  (виразу, що зв'язує наступний член ряду з попереднім), використовуючи наступні елементи керування :

**Label** – інформація про студента;

**Label** – інструкція користувачеві;

**ScrollBar** – для введення аргументу;

**ComboBox** – для введення точності обчислень;

#### Указівка:

На робочому листі в обраному Вами діапазоні розмістити значення, використовувати для різного ступеня точності обчислень ( наприклад, 0.1, 0.01, 0.0001).

Використовуючи програмний код, занесіть підготовлені Вами числа в поле зі списком (**ComboBox**).

**Command Button** – для обчислення функції;

**TextBox** – результати обчислень;

Обов'язково при висновку результатів обчислень здійснювати висновок функції, отриманої як суму членів ряду і по стандартній підпрограмі. (Результати повинні бути приблизно рівні).

Результати обчислень розмістити також на робочому листі.

### Варіанти завдань до лабораторних робіт.

#### Лабораторна робота №1

1	$W = \begin{cases} 0,3^v - v^2 + \cos v & -3 < v < 1; \\ \operatorname{ctg} (0,34v - 0,2) & 1 < v \leq 7; \end{cases}$
2	$U = \begin{cases} 0,2t + \operatorname{arctg} t & t < -2; \\ \operatorname{arcsin}(0,25t) & -2 < t \leq 4; t \neq 10; \end{cases}$
3	$Y = \begin{cases} \operatorname{arctg} x + e^x & x < 0,5; \\ \ln(x + \sin x) & 0,5 < x \leq 8; x \neq 3,0; \end{cases}$



4	$Z = \begin{cases} x^3 + \sin x; \\ \operatorname{arctg}(x + \lg x); \end{cases}$	$x < 0,3;$ $0,3 < x \leq 2; x \neq 0,5;$
5	$V = \begin{cases} z + \cos z; \\ \operatorname{arctg}(z + \ln z); \end{cases}$	$z < 0;$ $0 < z \leq 3; z \neq 10;$
6	$W = \begin{cases} 0,6v - 0,3^v; \\ \ln(v + \sqrt{v + \cos v}); \end{cases}$	$-2 < v < 0,3;$ $0,3 < v \leq 5; v \neq 2,0;$
7	$U = \begin{cases} 1,3t - \sin t; \\ \lg(t + \sqrt{t}); \end{cases}$	$t < 0;$ $0 < t \leq 5; t \neq 3,5;$
8	$T = \begin{cases} 1,3y - \sin y; \\ \operatorname{arctg}(y + \sqrt{y}); \end{cases}$	$y < 0,3;$ $0,3 < y \leq 2; y \neq 10;$
9	$F = \begin{cases} v^2 - \sqrt[3]{v}; \\ \operatorname{arctg}(v + \ln v); \end{cases}$	$v < 0;$ $0 < v \leq 3; v \neq 2,5;$
10	$Y = \begin{cases} x - e^x; \\ \operatorname{arctg}(x + \sqrt{x - 1,4}); \end{cases}$	$x < 0;$ $2 \leq x < 5; x \neq 3;$
11	$W = \begin{cases} y + \sin y; \\ \ln(y + \sqrt[3]{y}); \end{cases}$	$y < 0,5;$ $0,5 \leq y < 8; y \neq 1;$
12	$U = \begin{cases} \operatorname{arctg} v - e^v; \\ \lg(v + \cos v); \end{cases}$	$v < 1;$ $1 < v \leq 3; v \neq 2;$
13	$V = \begin{cases} t + \cos t; \\ \operatorname{arctg}(t + \ln t); \end{cases}$	$t < 0,5;$ $0,5 < t \leq 2; t \neq 1;$
14	$W = \begin{cases} z + \sin z; \\ \operatorname{arctg}[z + \lg(z + 0,2)]; \end{cases}$	$z < 0,5;$ $0,7 < z \leq 3; z \neq 2;$
15	$T = \begin{cases} x^2 - e^x; \\ \ln(\operatorname{arctg} x + x); \end{cases}$	$x < 0,1;$ $0,1 < x \leq 2; x \neq 1,5;$
16	$F = \begin{cases} t + e^x; \\ \sqrt{t} - \ln(t + \operatorname{arctg} t); \end{cases}$	$t < 0,5;$ $0,7 < t \leq 4,5; t \neq 2;$

17	$Y = \begin{cases} \sin x + e^x; \\ \operatorname{arctg}(x-0,3); \end{cases}$	$x < 0;$ $0 < x \leq 3; x \neq 2,5;$
18	$V = \begin{cases} \cos z - z; \\ \ln(z + \sqrt{z}); \end{cases}$	$z < 0,5;$ $0,5 < z \leq 7; z \neq 4;$
19	$W = \begin{cases} v^2 + \sqrt[3]{v}; \\ \ln(v + \sin v); \end{cases}$	$v < 0,5;$ $0,5 < v \leq 8; v \neq 4;$
20	$U = \begin{cases} x - 0,8 \sin x; \\ \operatorname{arctg}(\lg x + 0,3); \end{cases}$	$x < 0,2;$ $0,2 < x \leq 3; x \neq 1;$
21	$F = \begin{cases} y + 0,1 \cos y; \\ \lg(y + \sqrt{y} + 0,6); \end{cases}$	$-2 < y < 0,5;$ $0,5 < y; y \neq 3;$
22	$X = \begin{cases} \sin z - z^2; \\ \operatorname{arctg}(\lg z + \sqrt{z}); \end{cases}$	$z < 0,2;$ $0,5 < z \leq 3; z \neq 1;$
23	$X = \begin{cases} 1,26^v + v; \\ \operatorname{arctg}(v + 0,4); \end{cases}$	$v < 0,1;$ $0,1 \leq v < 4; v \neq 3;$
24	$Y = \begin{cases} w + \cos w; \\ \operatorname{arctg} w - \lg(w + \sqrt{w}); \end{cases}$	$w < 0,5;$ $0,5 < w \leq 2; w \neq 1,5;$
25	$Y = \begin{cases} \operatorname{arctg} z + z; \\ \lg z + \sqrt{z}; \end{cases}$	$z < 0;$ $0 < z \leq 5; z \neq 3;$
26	$Y = \begin{cases} x - e^x; \\ \operatorname{arctg}(x + \sqrt{x} - 1,4); \end{cases}$	$x < 0;$ $2 \leq x < 5; x \neq 3;$
27	$W = \begin{cases} y + \sin y; \\ \ln(y + \sqrt{y}); \end{cases}$	$y < 0,5;$ $0,5 < y \leq 8; y \neq 4;$
28	$U = \begin{cases} \operatorname{arctg} -e^v; \\ \lg(v + \cos v); \end{cases}$	$v < 1;$ $1 < v \leq 3; v \neq 2;$
29	$V = \begin{cases} t + \cos t; \\ \operatorname{arctg}(t + \ln t); \end{cases}$	$t < 0,5;$ $0,5 < t \leq 2; t \neq 1;$

30	$W = \begin{cases} z - \sin z; \\ \operatorname{arccctg}[z + \lg(z + 0,2)]; \end{cases}$	$z < 0,5;$ $0,75 < z \leq 3, z \neq 2;$
----	--	---

Лабораторна робота №2

1	$Y = \sum_{n=i}^k \frac{n}{n^2 - 5n + 6}$	16	$F = \prod_{n=l}^m \frac{n}{(n-2)(n-3)}$
2	$Y = \sum_{n=i}^k \frac{n-2}{n^2 - 16}$	17	$Y = \sum_{j=n}^i \frac{(j+2)^3}{(j-1)(j+6)^2}$
3	$Y = \sum_{n=i}^k \frac{n+1}{n^2 - 3}$	18	$Y = \prod_{l=i}^k \frac{l^2 + 4l + 8}{l-5}$
4	$F = \prod_{k=m}^n \frac{k+3}{(k-5)(k-6)}$	19	$S = \sum_{i=k}^m \frac{(i+1)^3}{(i^2 - i - 6)}$
5	$Y = \sum_{n=l}^k \frac{n^2 - n}{n^2 + n + 6}$	20	$Y = \prod_{l=i}^n \frac{l^2 + 2l}{(l-5)(l+2)}$
6	$W = \prod_{k=n}^i \frac{k+1}{(k-5)(k-7)}$	21	$Y = \sum_{n=k}^i \frac{(n+3)^2}{(n-5)(n-7)}$
7	$S = \sum_{n=i}^k \frac{n^2 - 5n + 7}{n^2 - 8n + 15}$	22	$V = \prod_{k=m}^n \frac{k+3}{k^2 - 10k + 24}$
8	$Y = \prod_{n=i}^l \frac{n}{(n-3)(n-8)}$	23	$Y = \sum_{i=k}^n \frac{i+2}{(i-3)(i-6)}$
9	$S = \sum_{i=k}^n \frac{i^2 - i + 3}{i-5}$	24	$Y = \prod_{k=n}^m \frac{k^2}{(k-3)(k-4)}$
10	$Y = \prod_{n=l}^m \frac{(n+3)^2}{(n-5)^3}$	25	$Y = \sum_{n=k}^j \frac{n+3}{(n-5)(n-6)}$
11	$S = \sum_{i=k}^j \frac{i(i+1)}{(i+7)(i+3)}$	26	$F = \sum_{i=k}^n \frac{i+4}{(i-2)(i-1)}$
12	$Z = \prod_{k=i}^j \frac{(n+3)^2}{(n-7)(k-9)}$	27	$Z = \sum_{n=k}^i \frac{(n-3)}{(n-1)(n+3)}$

13	$Z = \sum_{i=k}^n \frac{(i+1)^2}{2i^3 - 3i - 10}$	28	$V = \sum_{k=i}^m \frac{k-5}{k+1}$
14	$Z = \prod_{k=m}^l \frac{k+1}{(k-8)(k-6)}$	29	$V = \sum_{m=n}^j \frac{m+3}{(m-1)(m+2)}$
15	$S = \sum_{j=n}^k \frac{(j+5)^3}{j^3 - j^2 - j - 2}$	30	$Z = \sum_{j=i}^n \frac{j-1}{j+3}$

### Лабораторна работа №3

1	$x^2 - 5 * e^x = 0;$
2	$1/x + \cos(x) - 4 = 0;$
3	$e^x + x - 10 = 0;$
4	$\sin(x) + x - 10 = 0;$
5	$e^x + x^2 - 20 = 0;$
6	$\cos(x) - x + 2 = 0;$
7	$x^5 + \cos(x) - 20 = 0;$
8	$x^6 - \sin(x) - 10 = 0;$
9	$e^x + \cos(x) - 5 = 0;$
10	$e^x - x^3 + 2 = 0;$
11	$x^5 + \cos(x^2) - 10 = 0;$
12	$1/x + \sin(x) = 0;$
13	$x^{-3} + \cos(x) = 0;$
14	$x^3 - 5 * e^x + 3 = 0;$
15	$x^5 + \sin(x) - 2 = 0;$
16	$1/x + \cos(x) = 0;$
17	$e^x + x^3 - 12 = 0;$
18	$x^7 - \sin(x) - 10 = 0;$
19	$e^x - x - 3 = 0;$
20	$1/x - \sin(x) + 3 = 0;$
21	$x^{-2} + \sin(x) - 1 = 0;$
22	$1/x - \cos(x) - 0.5 = 0;$
23	$x^5 + x^3 - 1 = 0;$
24	$1/x + \sin^2(x) - 2 = 0;$
25	$2 * e^x + \operatorname{tg}(x) - 10 = 0.$
26	$e^x + x^3 - 5 = 0$
27	$e^x + x - 8 = 0$

28	$x^3 + 5 * e^x + 3 = 0$
29	$e^x + x^2 - 10 = 0;$
30	$-e^x + x^2 - 2x + 1 = 0;$

Лабораторна робота №5

У кожнім із завдань перетворити масив  $\{C_i\}$ ,  $i=1, \dots, n$  у масив  $\{D_i\}$ .

1.  $D_i = C_i / \max_{1 \leq i \leq n} \{C_i\}$ . Якщо  $\max_{1 \leq i \leq n} C_i = 0$ , масив  $\{C_i\} = \{D_i\}$

2.  $D_i = C_i / \min_{1 \leq i \leq n} \{C_i\}$ . Якщо  $\min_{1 \leq i \leq n} C_i = 0$ , то  $\{C_i\} = \{D_i\}$ .

3. Від'ємні елементи перетворити в нульові, нульові – зробити рівними числу 3, додатні – рівні числу 1.

4. Масив  $\{D_i\}$  отриманий з  $\{C_i\}$  шляхом сортування масиву  $\{C_i\}$  по убутанню абсолютних величин його елементів.

5. У масиві  $\{D_i\}$  міститися тільки ті елементи масиву  $\{C_i\}$ , що більше заданого числа  $K$ . Якщо таких елементів ні, те вивести відповідне повідомлення.

6.  $D_i = (C_i + C_{i+1} + C_{i+2}) / K$ ,  $i = 1, 2, \dots, n-2$ . Число  $K$  – параметр, що задається.

7.  $D_i = \frac{|C_i| + |C_{i+1}|}{2}$ ,  $i = 1, 2, \dots, n-1$ .

8. У масиві  $\{D_i\}$  містяться тільки ті елементи масиву  $\{C_i\}$ , що менше суми всіх його елементів, що стоять на непарних місцях.

9. Масив  $\{D_j\}$  складається з номерів від'ємних елементів масиву  $\{C_i\}$ .

10. Масив  $\{D_j\}$  складається з номерів парних елементів масиву  $\{C_i\}$ .

11. Непарні елементи  $\{C_i\}$  звести в квадрат, парні - в куб.

12. Елементи, що стоять на непарних місцях звести в квадрат, на парних місцях звести в куб.

13.  $D_k = \begin{cases} K \cdot C_k, & \text{якщо } K - \text{парне} \\ \frac{C_k}{K}, & \text{якщо } K - \text{непарне} \end{cases}$

14. В елементів з індексами, що кратні трьом, перемінити знак, інші не торкати.

15. Знайти добуток елементів, що стоять на парних місцях, і кожен елемент масиву  $\{C_i\}$  розділити на отримане ненульове число. Якщо число дорівнює 0, вивести відповідне повідомлення, а елементи масиву  $\{C_i\}$  розділити на число 3.

16. Задано числа  $x$  і  $z$ . Всі елементи  $\{C_i\}$ , що менше ніж  $x$ , розділити на  $n$ , а ті що чи більше рівно  $x$ , на число  $z$ .

17. Знайти мінімальне по модулі число і всі від'ємні елементи розділити на нього. Якщо число дорівнює 0, то кожен елемент помножити на 2.

18. Масив  $\{D_j\}$  складається з перших  $n$  елементів масиву  $\{C_i\}$ , добуток яких менше заданого числа  $P$ . Якщо перший елемент більше числа  $P$ , вивести повідомлення про значення першого елемента масиву  $\{C_i\}$  і попросити ввести інше число  $P$ .

19. Кожен другий елемент звести в квадрат, інші зменшити на порядковий номер елемента в масиві.

20. Кожен третій елемент масиву збільшити на суму додатних елементів масиву.

$$21. D_j = \begin{cases} X_{j+1} - |X_j|, & \text{якщо } j - \text{парне} \\ X_{j+1} + |X_j|, & \text{якщо } j - \text{непарне}; j = 1, 2, \dots, n-1 \end{cases}$$

$$22. D_j = \begin{cases} -x_j, & \text{якщо } x_j < 0 \\ 0, & \text{якщо } x_j > 0 \\ 33, & \text{якщо } x_j = 0 \end{cases}$$

23. Кожен елемент, що поділяється на три, замінити його номером.

24. Кожен третій елемент замінити різницею між елементом і його індексом.

25. Кожен третій елемент, що не дорівнює нулю, замінити зворотною величиною. Інакше елемент не змінювати.

26. Кожен елемент кратний числу 5 замінити протилежним числом.

27. Кожен від'ємний елемент перетворити в додатний, позитивний зменшити вдвічі.

28. Кожен елемент скласти з добутком усіх чисел. Якщо добуток дорівнює 0, то із сумою всіх чисел.

29. Кожен елемент скласти з добутком негативних чисел масиву  $C_i$ .

30. Кожен елемент скласти із сумою непарних елементів.

Лабораторна робота № 2, 7

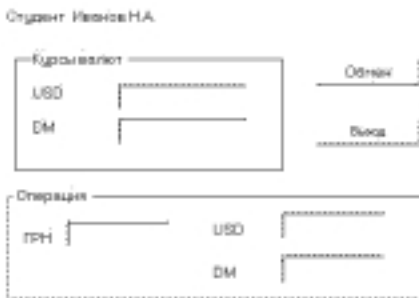
1	$\ln(x+1) = x - \frac{x^2}{2} + \frac{x^3}{3} - \dots$ $U_{n+1} = -\frac{U_n XN}{N+1}$
2	$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots$ $U_{n+1} = \frac{U_n X}{N}$
3	$\arctg x = x - \frac{x^3}{3} + \frac{x^5}{5} - \dots$ $U_{n+1} = -\frac{U_n (2N-1)X^2}{2N+1}$
4	$ch x = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \dots$ $U_{n+1} = \frac{U_n X^2}{2N(2N-1)}$
5	$sh x = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \dots$ $U_{n+1} = \frac{U_n X^2}{2N(2N+1)}$
6	$\ln(1-x) = -x - \frac{x^2}{2} - \frac{x^3}{3} - \dots$ $U_{n+1} = \frac{U_n XN}{N+1}$
7	$\frac{1}{1+x} = 1 - x + x^2 - x^3 - \dots$ $U_{n+1} = -U_n X$
8	$\frac{1}{(1+x)^2} = 1 - 2x + 3x^2 - \dots$ $U_{n+1} = -\frac{U_n X(N+1)}{N}$
9	$e^{-x} = 1 - \frac{x}{1!} + \frac{x^2}{2!} - \dots$ $U_{n+1} = -\frac{U_n X}{N}$
10	$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots$ $U_{n+1} = -\frac{U_n X^2}{2N(2N-1)}$
11	$e^{-x^2} = 1 - \frac{x^2}{1!} + \frac{x^4}{2!} - \dots$ $U_{n+1} = -\frac{U_n X^2}{N}$
12	$\frac{\sin x}{x} = 1 - \frac{x^2}{3!} + \frac{x^4}{5!} - \dots$ $U_{n+1} = -\frac{U_n X^2}{2N(2N+1)}$

13	$U_{n+1} = -\frac{U_n X(N+2)}{N}$
14	$\sin x = \frac{x}{1!} - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots \quad U_{n+1} = -\frac{U_n X^2}{2N(2N+1)}$
15	$3^x = 1 + \frac{\ln 3}{1!} + \frac{\ln^2 3}{2!} x^2 + \dots \quad U_{n+1} = \frac{U_n X \ln 3}{N}$
16	$\frac{1}{\sqrt{1-x^2}} = 1 + \frac{1}{2}x^2 + \frac{1 \cdot 3}{2 \cdot 4}x^4 + \dots \quad U_{n+1} = \frac{U_n (2N-1)X^2}{2N}$
17	$\sin x^2 = x^2 - \frac{x^6}{3!} + \frac{x^{10}}{6!} - \dots \quad U_{n+1} = -\frac{U_n X^4}{2N(2N+1)}$
18	$\frac{x(3-x)}{(1-x)^3} = 3x + 8x^2 + 15x^3 + \dots \quad U_{n+1} = \frac{U_n (N+1)(N+3)X}{N(N+2)}$
19	$\frac{1}{2} \ln \frac{1+x}{1-x} = x + \frac{x^3}{3} + \frac{x^5}{5} + \dots \quad U_{n+1} = \frac{U_n (2N-1)X^2}{2N+1}$
20	$\frac{1}{2} \ln x = \frac{x-1}{x+1} + \frac{1}{3} \left( \frac{x-1}{x+1} \right)^3 + \dots \quad U_{n+1} = U_n \left( \frac{X-1}{X+1} \right)^2$
21	$\ln \frac{1}{x^2 + 2x + 2} = (1+x)^2 + \frac{(1+x)^4}{2} - \dots \quad U_{n+1} = -U_n \frac{(X+1)^2 N}{N+1}$
22	$\frac{\cos x}{x} = \frac{1}{x} - \frac{x}{2!} + \frac{x^3}{4!} - \dots \quad U_{n+1} = -\frac{U_n X^2}{2N(2N-1)}$
23	$e^{2x} = 1 + \frac{2x}{1!} + \frac{4x^2}{2!} + \frac{8x^3}{3!} + \dots \quad U_{n+1} = U_n \frac{2X}{N}$
24	$\frac{1}{\sqrt{1+x}} = 1 - \frac{1}{2}x + \frac{1 \cdot 3}{2 \cdot 4}x^2 - \dots \quad U_{n+1} = U_n \frac{(2N-1)X}{2N}$
25	$\frac{1}{\sqrt[3]{x+1}} = 1 - \frac{1}{3}x + \frac{1 \cdot 4}{3 \cdot 6}x^2 - \dots \quad U_{n+1} = -U_n \frac{3N-2}{3N} X$



26	$\arctg x = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots$ $U_{n+1} = U_n \left( - \left( \frac{(2N-1)X^2}{(2N+1)} \right) \right)$
27	$\arcsin x = x + \frac{x^3}{2 \cdot 3} + \frac{1 \cdot 3 \cdot x^5}{2 \cdot 4 \cdot 5} - \dots$ $U_{n+1} = U_n X^2 \frac{(2N-1)}{2N(2N+1)}$
28	$\ln(x) = 2 \left( z + \frac{z^3}{3} + \frac{z^5}{5} + \dots \right); z = \frac{(x-1)}{(x+1)}$ $U_{n+1} = U_n z^2 \frac{(2N-1)}{(2N+1)}$
29	$\ln \frac{1+x}{1-x} = 2 \left( x + \frac{x^3}{3} + \frac{x^5}{5} + \dots \right)$ $U_{n+1} = U_n X^2 \frac{(2N-1)}{(2N+1)}$
30	$\frac{1}{1-x} = 1 + x + x^2 + x^3 + \dots$ $U_{n+1} = U_n X$

**Приклад виконання лабораторної роботи  
№6**



**№7**

**Лаб. раб.**

Студент Петренко В.И. гр. ЗБ1-02

Аргумент:

$\frac{1}{1+x} = 1 - x + x^2 - x^3 + \dots$

Рекуррентное соотношение:

$U_{n+1} = -U_n x$

Точность:

Результаты:

## **ДОСВІД ЗАСТОСУВАННЯ ВІРТУАЛЬНИХ МАШИН VMWARE ПРИ НАВЧАННІ МЕРЕЖНИМ ІНФОРМАЦІЙНИМ ТЕХНОЛОГІЯМ**

К.Г. Грищенко

м. Суми, Українська академія банківської справи

K.Gritsenko@academy.sumy.ua

Людина має певну вартість на ринку праці і ця вартість великою мірою залежить від якості її освіти і набутого досвіду практичної діяльності. Вузи орієнтуються в своєму розвитку на якісну підготовку спеціалістів, яка відповідає вимогам ринку праці. Щоб отримати якісну освіту, потрібно ефективно витратити час на навчання. В умовах ринкових відносин будь-яке діло, щоб бути успішним, потребує особливих знань, спеціальної підготовки, технічного оснащення і творчого ставлення до праці. Вузу доводиться враховувати ці вимоги в навчальному процесі. Головний критерій сучасного спеціаліста полягає не тільки в оволодінні знаннями з фундаментальних дисциплін, але й у виробленні допитливого, самостійного мислення. Формування такої здібності – найважливіша мета усіх дисциплін навчального плану спеціальності “Економічна кібернетика”.

Сьогодні наявність достатньої кількості інформації стала життєвою необхідністю для підприємств усіх форм власності та органів влади, а забезпечення, зберігання, захист, обробка та передавання інформації об’єктивно вимагають всебічного застосування сучасних мережних інформаційних технологій і комунікаційних засобів. У зв’язку з цим особлива роль у фундаментальній підготовці спеціалістів з “Економічної кібернетики” належить дисципліні “Основи операційних систем і комп’ютерних мереж”. Підготовка спеціалістів у цій області потребує суттєвого перегляду технології навчання у вузі в напрямку інтенсифікації використання комп’ютерної техніки. Особливу роль при цьому відіграє самостійна робота студента, яка дозволяє вирішувати основну проблему навчання – проблему становлення спеціаліста як особистості. Метою самостійної роботи є формування майбутнього фахівця, який може організувати та реалізувати свою діяльність без сторонньої допомоги, що особливо актуально в умо-

вах ринкової економіки. Повноцінну самостійну роботу студента при вивченні дисципліни “Основи операційних систем і комп’ютерних мереж” в умовах обмеженої кількості доступної для нього комп’ютерної техніки (один комп’ютер для одного студента) забезпечує програма VMware, для інсталяції якої потрібна ОС Windows 2000.

VMware представляє собою емулятор віртуальної машини. VMware повністю емулює роботу звичайного комп’ютера, починаючи з POST-тесту BIOS при включенні і до моменту вимикання віртуальної машини. Віртуальний комп’ютер може використовувати всі пристрої того комп’ютера, на якому запущений емулятор (host-комп’ютер), причому користувач може визначити, які пристрої будуть доступні, а які – ні. VMware надає можливість не тільки використовувати існуючі жорсткі диски, але і створювати віртуальні диски (які на host-комп’ютері зберігаються в вигляді звичайних файлів). Конфігурація віртуальної машини зберігається в вигляді текстового файлу. Розмір віртуального диска може перевищувати обсяг вільного простору на реальному диску. Річ у тому, що розмір файла, який моделює диск, визначається тільки реальними потребами віртуальної машини.

VMware дозволяє одночасно запускати декілька віртуальних машин, на яких можуть бути встановлені різні операційні системи (називані гостьовими ОС), причому це можна робити як у віконному режимі, так і в повноекранному режимі для повноти реалізму. Для кожного окремого віртуального комп’ютера створюється окремий каталог з конфігураційними файлами. В лабораторному практикумі з дисципліни “Основи операційних систем і комп’ютерних мереж” студенти виконували повну інсталяцію та настройку, як і в реальних виробничих умовах, таких розповсюджених в цей час ОС, як Windows 98, Windows 2000 Server і Linux, починаючи зі створення віртуальних дисків, розбивки їх на розділи, форматування розділів, настройки BIOS.

VMware підтримує комп’ютерну мережу в декількох режимах: **host only networking**, коли віртуальна комп’ютерна мережа існує тільки на host-комп’ютері (пов’язує між собою основну та гостьові ОС на одному комп’ютері), і **bridged networking**, коли віртуальна машина підключається до реальної комп’ютерної мережі навчального закладу. Така послуга VMware дозволяє студе-

нту самостійно на одному комп'ютері опанувати мережні інформаційні технології різних ОС. Хоча досвід проведення занять показує, що колективна робота студентів в мережі при вивченні мережних технологій є найбільш продуктивною.

При вивченні дисципліни “Основи операційних систем і комп'ютерних мереж” студенти постійно працювали у віртуальному середовищі (віртуальній мережі з віртуальними комп'ютерами), що дозволило забезпечити стабільну роботу комп'ютерної техніки під час проведення в комп'ютерних класах занять з інших дисциплін, а також дало можливість студентам максимально реалізувати їх творчі здібності в опануванні складного матеріалу (адміністрування ресурсів, користувачів і груп, маршрутизація, робота з доменами Active Directory тощо), суттєва частка якого була винесена на самостійне опрацювання.

Використання віртуальних машин дозволяє студентам постійно оновлювати їх знання і уміння в умовах, максимально наближених до практики. Сьогодні роботодавця цікавить не тільки теоретична основа підготовки випускника, а і його фахова спроможність розв'язувати поставлені завдання, а також те, наскільки швидко він пройде адаптацію на виробництві. Швидкість адаптації працівника, який щойно закінчив вуз, залежить безпосередньо від того, як він підготовлений до роботи в умовах частого оновлення техніки і технологій. Досвід застосування віртуальних машин VMware при навчанні мережним технологіям різних ОС показує, що в процесі навчання студент не просто засвоює навчальну дисципліну, а й вчиться використовувати інформацію, яку одержує, для формування нових знань. Таким чином процес навчання одночасно перетворюється в науково-дослідну діяльність, що є важливим аспектом підготовки сучасного фахівця в області інформаційних технологій. Будь-яка інформація відразу стає значущою, свідомо класифікується у важливу, актуальну зараз, або ж визначається, для чого вона може стати в пригоді в майбутньому, як та, що пов'язана з раніше відомою інформацією, а також яке значення вона може мати для збільшення знань в предметній області.

# ПОРІВНЯЛЬНИЙ ПІДХІД ДО НАВЧАННЯ ОСНОВАМ ПРОГРАМУВАННЯ СТУДЕНТІВ ТЕХНІЧНИХ СПЕЦІАЛЬНОСТЕЙ ВНЗ

Л.В. Гришко

м. Черкаси, Черкаський державний університет  
ім. Б. Хмельницького

Як зазначено в роботі Дж. Брукшира [1], задачею всієї галузі комп'ютерних наук є розробка алгоритмів. При цьому вивчення методів побудови та аналізу алгоритмів, можливостей різноманітних мов програмування повинні бути одними з фундаментальних задач при навчанні майбутніх програмістів.

Мови програмування – це основний засіб спілкування програміста з комп'ютером. Програміст повинен не лише вміти скласти програму на якійсь одній мові, він повинен також розуміти різні стилі програмування, притаманні різним мовам. На сучасному етапі розвитку інформаційних технологій від програміста вимагається вміння працювати з множиною різних мов і, навіть, стилів програмування одночасно. Володіння різними парадигмами програмування, знання декількох мов, розуміння головних рис мов програмування значно полегшує програмісту засвоєння нових мов і дозволяє крокувати разом з усією індустрією створення програмного забезпечення.

З якої ж методології або парадигми слід починати вивчення програмування? Програмування – це один з небагатьох видів людської діяльності, що потребує цілковитої точності. Прищеплення студентам звички дотримуватися досконалої точності при розробці власних програмних продуктів є найбільш складним у процесі навчання програмуванню [2]. Навчання програмуванню на базі імперативних (процедурних) мов, на нашу думку, дозволяє сформувати у майбутніх програмістів цю важливу рису і найбільш повно розв'язувати основні завдання їх професійної підготовки, які, зокрема, передбачають [3]:

- розвиток інтелектуальних здібностей студентів, зокрема, алгоритмічного стилю мислення;
- формування знань про:
  - стратегії проектування алгоритмів і програм;

- евристичні методи конструювання алгоритмів і програм;
- базові принципи побудови алгоритмів і програм;
- типові алгоритмічні конструкції: послідовність, вибір, повторення;
- критерії ефективності алгоритмів і програм;
- рекурсивні алгоритми та способи їх побудови;
- типи і структури даних та алгоритми їх обробки;
- модульний принцип розробки програм;
- методи структурного низхідного (спадного) програмування;
- методи динамічного розподілу пам'яті;
- засоби препроцесорної обробки програми;
- формування умінь
  - аналізувати поставлену задачу, будувати її інформаційну модель;
  - розробляти алгоритм розв'язування поставленої задачі методом покрокового уточнення;
  - використовувати низхідний метод для розв'язування складних задач;
  - аналізувати алгоритми на предмет правильності та ефективності;
  - віднаходити серед великої кількості гіпотетично можливих варіантів програми найбільш ефективні в певній конкретній ситуації;
  - застосовувати стратегію розуміння програм, яка узгоджує інформацію в тексті програми зі знаннями студента-програміста, програмування взагалі і предметної області задачі, що розв'язується, зокрема;
  - доводити правильність алгоритмів і програм;
  - використовувати стандартні та структуровані типи даних;
  - реалізувати багатомодульні програми;
  - складати і використовувати макрозасоби та засоби препроцесорної обробки програм.

Деякі вітчизняні і західні університети пропонують в якості базової використовувати об'єктно-орієнтовану методологію програмування [4, 5]. Однак в цьому випадку з розгляду практично

випадає алгоритмічна складова програмування, що, на нашу думку, є неприйнятними. Тому при підготовці майбутніх програмістів пропонується така послідовність навчання програмуванню: від імперативних мов Pascal і C до об'єктно-орієнтованих C++ і Java.

До мов програмування, які підтримують імперативну модель обчислень, відносяться мови програмування Pascal та C. Ми вважаємо, що навчання майбутніх програмістів основам програмування слід починати саме з цих мов програмування.

Цей вибір зумовлено декількома факторами. Вибір мови програмування Pascal обумовлено тим, що основи цієї мови, як правило, відомі більшості студентів з шкільного курсу інформатики. Завдяки цьому можна, по-перше, вдосконалювати знання мови Pascal, по-друге, починаючи вже з першого семестру навчання розв'язувати алгоритмічно складні задачі, по-третє, виходячи з того, що в конструкціях мов програмування Pascal і C є багато спільного, навчати мові програмування C у порівнянні з мовою Pascal, що сприяє, на нашу думку, більш глибокому засвоєнню кожної з цих мов.

Мова програмування C обрана з міркувань перспективності основних практичних областей програмування. Опитування, яке було проведене серед керівників американських комп'ютерних компаній в 1999 році показало, що 46,2% респондентів вважають найбільш перспективною галуззю програмування мову програмування C++ [4]. Автор мови програмування C++ Бьярн Страуструп пише наступне: "За виключенням несуттєвих деталей, C++ є надмножиною мови C" [5]. З метою залучення майбутніх програмістів з перших кроків навчання до найбільш перспективних напрямків програмування і було обрано мову C, що передбачено і в проєктах державних стандартів з комп'ютерних наук. Слід також зазначити, що мова програмування C була і залишається основною мовою системного програмування, тобто знання цієї мови не будуть зайвими в багажі майбутніх фахівців.

У Черкаському державному університеті проводиться педагогічний експеримент по впровадженню в навчальний процес методичної системи навчання основам програмування за експериментальною програмою курсу "Основи програмування та алгоритмічні мови" для студентів, які навчаються за спеціальністю

“Програмне забезпечення автоматизованих систем”. Під час експерименту перевіряється, зокрема, наступна гіпотеза: паралельне вивчення двох мов програмування на основі їх порівняння може виступати в ролі механізму інтенсифікації процесу навчання, активізації пізнавальної діяльності студентів, розвитку основних мислительних операцій і формування у них узагальнених знань з програмування.

На цей час є можливість зробити деякі висновки. В порівнянні з минулими роками викладання цієї дисципліни, коли викладання велося за ланцюжком Pascal → C → C++, більше часу витрачалося на вивчення синтаксису мов програмування, а під час експерименту цей етап вдалося скоротити і головною складністю для студентів стала алгоритмічна складова при розв’язуванні задач, чого ми і прагнули. Цікавими з цього приводу є думки студентів. По закінченні першого семестру вивчення курсу «Основи програмування та алгоритмічні мови» було проведено анкетування експериментальної групи студентів першого курсу спеціальності “Програмне забезпечення автоматизованих систем”.

Після обробки відповідей на запитання анкети отримані наступні результати:

- 77% студентів вивчали мову програмування Pascal у школі;
- 50% студентів до вступу в університет додатково займалися інформатикою, і, зокрема, програмуванням;
- 46% опитаних студентів вважають доцільним вивчення мови Pascal в університеті, з них, 13% опитаних студентів вважають так тому, що не вивчали цю мову в школі і 33% тому, що Pascal більш зрозумілий, ніж C;
- 10% студентів вважають, що знання мови C потрібні, як основа мов програмування C++, C#, Java;
- 100% студентів висловлюється за вивчення декількох мов програмування;
- жоден студент не сказав, що йому не подобається мова C (0%).

На екзамені з “Основ програмування та алгоритмічних мов” після першого семестру навчання для розв’язування задачі мову програмування Pascal обрали 79%. З них:



- 13% студентів тому, що ця мова відома їм із шкільного курсу інформатики;
- 23% – тому, що вона більше подобається;
- 45% – за звичкою;
- 23% – відповіли, що для них немає різниці, на якій мові писати програму.

Решта студентів (21%) розв’язували задачі на С і робили це з наступних причин:

- 6% – більше подобається мова С;
- 3% – приємно усвідомлювати, що можу писати програми мовою С;
- 3% – немає різниці, якою мовою писати програму, бо однаково володіють обома мовами;
- 3% – хотіли продемонструвати свої знання мови С;
- 6% – хотіли перевірити свої знання мови С.

Після підведення підсумків анкетування на формуючому етапі педагогічного експерименту з’ясувалось, що 81% студентів вважають доцільним паралельне вивчення мов Pascal і С на основі їх порівняння, при цьому 55% студентів вважають, що систематизуються знання мови Pascal і легше, на їх думку, вивчати мову С, а 42% студентів вважають, що це надає можливість робити порівняльний аналіз мов програмування; 13% респондентів вважають недоцільним паралельне вивчення мов Pascal і С. Для цієї категорії студентів послідовне вивчення мов програмування простіше; 6% – не вважають за потрібне вивчення мови Pascal у ВНЗ тому, що вивчали її у школі.

На основі аналізу роботи студентів на лабораторних заняттях, опитувань і тестування за матеріалами лекційного курсу, результатів проведеного іспиту, а також анкетування студентів, можна зробити попередній висновок про те, що паралельне вивчення мов Pascal і С на основі їх порівняння сприяє інтенсифікації процесу навчання, активізації пізнавальної діяльності студентів, розвитку у них основних мислительних операцій і формуванню узагальнених знань з програмування.

#### Література:

1. Брукшир Дж. Введение в компьютерные науки. Общий обзор, 6-е издание.: Пер. с англ. - М.: Издательский дом

- “Вильямс”, 2001.
2. Ершов А.П., Эстетический и человеческий фактор в программировании. – Кибернетика. – 1972. – №5.
  3. Триус Ю.В., Богатирьев О.О., Гришко Л.В. Особенности створення методичної системи навчання основам програмування для підготовки майбутніх інженерів-програмістів. – Вісник Черкаського університету. – Випуск 35. – 2002.
  4. Одинцов И.О. Профессиональное программирование. Системный подход. – СПб.: БХВ-Петербург, 2002.
  5. Страуструп Б. Язык программирования С++, спец. Изд. / Пер. с англ. – М.: СПб.: Издательство БИНОМ – «Невский диалект», 2001.

## ДОСВІД ВИКОРИСТАННЯ ВІДКРИТОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В КУРСІ ІНФОРМАТИКИ

О.В. Гришпенюк<sup>1</sup>, П.М. Григоруку<sup>2</sup>

<sup>1</sup> м. Хмельницький, Хмельницька філія Міжрегіональної Академії управління персоналом

<sup>2</sup> м. Хмельницький, Технологічний університет Поділля

На сьогодні однією з найактуальніших проблем при викладанні дисципліни “Інформатика та комп’ютерна техніка” є проблема вибору програмних продуктів для забезпечення навчально-виховного процесу. Ще декілька років тому рік тому цієї проблеми практично не існувало – усіх влаштовувало використання при навчанні “піратського” програмного забезпечення. Однак ситуація докорінно змінилась з початком перевірки ліцензійності встановленого програмного забезпечення. Але чи багато навчальних закладів може дозволити собі стовідсоткове забезпечення ліцензованими програмними продуктами? Останнім часом ця проблема набуває усе більшого масштабу, зважаючи на політику компаній у цьому напрямку та боротьбу із “піратським” програмним забезпеченням. Отже, користувач змушений або платити за програмне забезпечення або користуватися альтернативними “вільними” програмними продуктами.

Виходом з даного становища для закладів освіти є перехід до вільного програмного забезпечення, яке, звичайно, не потребує таких коштів.

Насамперед постає проблема вибору операційної системи, яка була б гідною альтернативою ОС Windows і у той же час вільно поширювалася. На думку фахівців, саме такою операційною системою на сьогодні є Linux.

У даній роботі хотілось би поділитися досвідом викладання дисциплін циклу “Інформатика та комп’ютерна техніка” з використанням зазначеної операційної системи.

В силу того, що вихідні коди Linux поширюються вільно і є загальнодоступними, до розвитку системи з самого початку підключилася велика кількість незалежних розробників. Завдяки цьому на сьогодні Linux є сучасною, стійкою операційною сис-

темою, яка розвивається досить швидкими темпами, майже миттєво вбираючи в себе усі останні технологічні новації.

Як позитивні якості системи, насамперед, слід зазначити не лише реальну багатозадачність, але й підтримку одночасної роботи багатьох користувачів. При цьому Linux може надавати усі системні ресурси користувачам, що працюють з хостом через різні віддалені термінали. Також для запуску різних програм розроблених для інших операційних систем, у тому числі і для ОС Windows, під Linux розроблені емулятори DOS та Windows.

Проте як і кожний програмний продукт Linux має і свої вади. Насамперед, це практична відсутність україномовних методичних розробок по застосуванню її в навчальному процесі. Досить суттєвою проблемою при забезпеченні ефективного навчально-виховного процесу для студентів спеціальностей економічного напрямку є неможливість запуску в емуляції програми “1С: Бухгалтерія” та інших програм економічного циклу.

Власне для викладання дисциплін циклу “Інформатика та комп'ютерна техніка” ми досить успішно використовували версію Linux, засновану на дистрибутиві Linux Mandrake 8.2 з графічною оболонкою KDE.

Кожний користувач має доступ до власної папки (Домашнього каталогу), до папки власної групи та до папки із завданнями.

При вивченні студентами I курсу спеціальності “Управління офісом” (молодший спеціаліст) команд ОС MS-DOS успішно використовували емулятор DOS, а для вивчення понять алгоритмізації та програмування FreePascal.

При вивченні офісних додатків використовувався пакет StarOffice 5.2:

- для створення та редагування текстових документів – текстовий процесор StarOffice Writer;
- для створення електронних таблиць та проведення обчислень – табличний процесор StarOffice Calc;
- для роботи з базами даних – StarOffice Base (Adabas D, dBase);
- для створення та редагування презентацій – StarOffice Impress;
- для створення макросів – StarOffice Basic.

Для перегляду та навігації Web-сторінками використовувався браузер Mozilla. Слід зазначити, що при інсталяції графічного середовища KDE встановлювати окрему програму-браузер взагалі нема потреби, оскільки в якості браузера у повній мірі може слугувати файловий менеджер Konqueror, за замовчуванням встановлений у складі KDE. Аналізуючи роботу із цим браузером, слід відзначити, що Konqueror дозволяє легко і зручно переглядати файли багатьох форматів, як текстових, так і графічних.

Пакет StarOffice має можливість розпізнання форматів документів створених у інших редакторах, звичайно, насамперед, маються на увазі документи створені процесорами та редакторами пакету Microsoft Office. Слід зазначити, що інтерфейс програм та набір основних операцій дуже схожий і це значною мірою спрощує перехід від пакету MS Office до StarOffice.

На жаль, для офісного пакету StarOffice також практично відсутня документація. Це ускладнює його опанування та розробку навчально-методичного забезпечення.

Проте, як показав досвід багато студентів, які працювали з продуктами Microsoft, досить важко адаптуються до аналогічних програмних продуктів під Linux, хоча з усіх інтегрованих графічних середовищ KDE найбільш близька до Windows як інтерфейсом, так і набором основних команд.

На нашу думку, ця проблема – наслідок поганого розвитку в студентів креативного мислення. Вони більше схильні бути пасивним користувачем, який лише повторює свої дії за викладачем, а не “відкриває” для себе щось нове самотужки. Саме тому ми намагаємося як найбільше урізноманітнювати завдання для студентів саме самостійним пошуком нового, навчити бачити загальні і спільні риси в різних програм, проводити аналогії, розвивати логічне мислення.

Отже, при переході до нових для студентів програмних продуктів слід приділити чимало уваги саме такому підходу до вирішення проблеми адаптації.

## **ВИКОРИСТАННЯ ЕОМ ЯК МОГУТНІЙ ЗАСІБ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ НАВЧАННЯ У ВНЗ**

А.І. Дзундза, Н.М. Лосева  
м. Донецьк, Донецький національний університет

Сучасний розвиток вищої освіти в Україні ґрунтується на досягненнях науково-технічного прогресу і характеризується збільшенням обсягу і поглибленням змісту знань, умінь і навичок, якими повинні володіти сучасні фахівці різних професій. Одним із перспективних засобів підвищення ефективності процесу навчання у ВНЗ є його автоматизація. Ефективність використання комп'ютерної техніки залежить від багатьох факторів.

Доцільність використання ЕОМ обумовлюється потребами:

1) індивідуалізації навчання, яка обумовлена залежністю результатів навчання від психічно-фізіологічних й інтелектуальних якостей студентів,

2) виконання численних досить одноманітних навчальних вправ і здійснення оперативного контролю правильності їхнього виконання,

3) здійснення перевірки рівня засвоєння знань за значним обсягом навчального матеріалу,

4) тренування різних навичок розумової і навчальної діяльності,

5) здійснювання тестового контролю знань студентів,

6) впровадження у навчальний процес ділових ігор різного роду, а також застосування елементів гри у навчанні;

7) організації самостійної навчальної діяльності студентів, що керується і контролюється викладачем,

8) забезпечення повторення й узагальнення отриманих знань,

9) здійснення ефективного консультування, видачу різного роду довідок,

10) накопичення статистичної інформації про хід навчального процесу і здійснення її обробки.

Поставлені вище завдання можуть бути з успіхом вирішені-ми через застосування автоматизованих навчальних систем. Автоматизована навчальна система (АНС) – новітній технічний за-

сіб, що здійснює навчання і контроль знань студентів за допомогою ЕОМ. У зв'язку із широким впровадженням в останні роки обчислювальної техніки в навчальні заклади АНС займає значне місце в навчальному процесі.

Навчальні програми складаються викладачем і містять у собі інформаційні матеріали для навчання, поточного контролю засвоєння знань студентом і керуючі директиви управління ЕОМ, що реалізують алгоритм навчання. По своїй сутності АНС у змозі забезпечити індивідуальну роботу кожного студента, тому основною її характеристикою і призначенням є індивідуалізація навчання. Крім того АНС здійснює навчання через ЕОМ, що приводить до строгої реалізації будь-якого алгоритму, будь-якої методики навчання, – ЕОМ “не утомлюється” і “не забуває” закладені в неї плани проведення занять і, звичайно, набагато швидше реагує на питання студентів. Усе це дозволяє здійснити ряд дидактичних прийомів, що не можуть бути реалізованими ніякими іншими засобами навчання:

- ведення навчального діалогу з кожним студентом окремо в прийнятному для нього темпі,
- видачу згідно запиту будь-якої довідкової інформації з теми навчання,
- можливості моделювання процесів і явищ, вивчення їхньої динаміки,
- збір, накопичення, обробку й аналіз даних про хід процесу навчання.

Інформаційна функція є основною при будь-якому навчанні, певні труднощі при цьому – корекція подачі інформації кожному студентові. АНС позбавлена цього недоліку. Навчальна програма обслуговує індивідуально кожного студента, як правило, вона розбита на кілька кроків, на кожному з яких обов'язково присутній кадр зворотного зв'язку. Навчання в АНС у більшому ступені, ніж при традиційному навчанні, зв'язане з активною розумовою діяльністю студентів. Навчальна програма постійно вимагає виконання студентами тих, чи інших розумових операцій, тому в АНС реалізується більш високий рівень інтелектуального розвитку, вона інтенсифікує і раціоналізує роботу студента. Іншою важливою перевагою АНС є можливість проведення адаптивного навчання. АНС дозволяє вибрати такі параметри адаптації:

- метод викладання навчального матеріалу,
- час наданий студенту для відповіді на запитання,
- наявність чи відсутність необхідної допомоги студенту при тестуванні.

АНС забезпечує студенту можливість постійного самоконтролю за засвоєнням необхідних знань, крім того, ця система видає оперативну інформацію про хід навчання, що створює позитивну мотивацію до скорішого оволодіння новим матеріалом, сприяє інтелектуальному настрою на активізацію пізнавальної діяльності. Навчальна програма передбачає можливість повторного викладання всього матеріалу або його частини.

Авторами розроблено тестуючі комплекси для ДЕК бакалаврів і декілька автоматизованих навчальних програм “Скінчені автомати”, “Скінчені розпізнавачі”, “Контекстно-вільні граматики” тощо, які застосовуються при навчанні і перевірці знань студентів з курсу «Системне програмування» у Донецькому національному університеті на математичному факультеті. Ці навчальні програми виконують ряд функцій викладача: служать джерелом інформації (надають студентам теоретичний матеріал, впорядкований певним чином, і приклади розв’язку задач за наданим матеріалом), організують навчальний процес (вибираючи індивідуальне завдання по даній темі), здійснюють контроль ступеня засвоєння матеріалу студентами (шляхом тестування, тобто на поставлене контрольне питання пропонується кілька варіантів відповідей, серед яких вірний лише один), регулюють темп процесу вивчення матеріалу, забезпечують зворотний зв’язок. Програми написані мовою програмування Delphi, і реалізовані на IBM PC.

Робота програми починається з появи на екрані основної форми, що служить для виклику інших форм за допомогою натискання однієї з кнопок: “Лекції”, “Приклади”, “Індивідуальні завдання”, “Тестування”. При натисканні на кнопку “Лекції” з’являється форма (Лекції), на якій міститься список лекцій, а також короткий зміст обраної лекції. Після вибору необхідної теми і натискання на кнопку “ОК” дана форма закривається, активною стає основна форма, в якій з’являється текст обраної теми. При натисканні на кнопку “ОК” ця форма закривається, активною стає основна форма, у якій з’являється текст обраної те-



ми.

При натисканні на кнопку “Приклади” в основній формі з’являється список прикладів, при виборі визначеного приклада і натисканні кнопки “ОК” з’являється його зміст.

При натисканні на кнопку “Індивідуальні завдання” з’являється форма (Індивідуальні завдання) на якій міститься список індивідуальних завдань, а також список варіантів. При виборі необхідного варіанту, а також при натисканні кнопки “ОК” з’являється текст індивідуальної роботи, яку можна роздрукувати, використовуючи кнопку «Друк». При необхідності студент може одержати текст вирішення варіанту свого індивідуального завдання.

При натисканні на кнопку «Тестування» з’являється форма (Тестування) з інформацією про тест і кнопку «ОК», при натисканні якої з’являється текст питання і чотири варіанти відповіді, один із яких є правильним. Після вибору відповіді і натискання на кнопку «Далі» з’являється наступний питання. Після вичерпання питань програма видає оцінку.

Зробимо зауваження з одного важливого приводу. Не можна вважати правильною дуже розповсюджену точку зору, начебто ключ до вирішення основних проблем комп’ютерного навчання - це розробка засобів, що дозволяють здійснювати перехід від сценарію навчальної програми до комп’ютерної програми. Таке представлення в ряді випадків негативно позначилося на розробці й оцінці ролі інструментарію для програмування навчальних курсів. Багато розроблювачів таких систем перебільшують не тільки можливості створених ними авторських систем, але й взагалі їхнє значення. Перебільшення можливостей навчальних програм часто сполучається з недооцінкою важливості тих психолого-педагогічних проблем, що виникають при розробці засобів програмного навчання. Не можна, маючи лише неясні уявлення про психолого-педагогічні особливості навчання, про зміст того чи іншого навчального предмета створити ефективну навчальну програму.

Поширення подібних поглядів вплинуло не тільки на теорію, але і на практику розробки навчальних програм. В ряді країн, наприклад, у США й особливо у Великобританії, протягом останніх 10–15 років з’явилося багато невеликих фірм (деякі з

них мають штат із двох-трьох програмістів), що розробляють навчальні програми, призначені для продажу. У нашій країні також нерідко серед одноособових розроблювачів навчальних програм були фахівці з обчислювальної техніки. Ця їхня діяльність, хоча і трохи відрізнялася від виконуваної раніше, проте по своїй суті залишалася звичною для них. У результаті створювалися численні, але малоефективні програми. Саме така практика стала основним джерелом ілюзій, начебто найбільші труднощі в розробці навчальних програм представляє кодування, як часто говорилося, або програмування навчальних курсів.

Стосовно програмованого навчання вираз «програмування навчальних курсів» став сприйматися як синонім «розробки навчальних курсів». А це привело до серйозних негативних наслідків: відвернуло увагу від найбільш важливих і трудомістких проблем – психолого-педагогічних проблем розробки навчальних програм (навчальних курсів), породило ілюзію, начебто створивши зручний інструментарій для кодування навчальних програм, можна за допомогою педагогів-ентузіастів вирішити проблему створення ефективних навчальних програм (навчальних курсів). Зрозуміло, провину за це не можна цілком покласти на перших розроблювачів навчальних програм і інструментарію для кодування (програмування) навчальних курсів. Просто вони, не будучи педагогами, не розуміли тих психолого-педагогічних проблем, що виникають при розробці навчальних програм.

У багатьох вищих навчальних закладах України розвиток програмованого навчання переживає стадію формування (створення навчальних програм, електронних підручників, дистанційне навчання) в умовах матеріально-технічних проблем, недосконалості теоретичної психолого-педагогічної бази. Але саме життя змушує нас рухатися у напрямку комп'ютеризації навчання.

## ПРОЕКТИРОВАНИЕ ЭЛЕКТРОННЫХ КОМПОНЕНТОВ ЭВМ ПРИ ПОМОЩИ IP CORE ГЕНЕРАТОРА

В.А. Долока, Л.А. Дубинец, Н.Л. Кириленко  
г. Черкассы, Черкасский государственный технологический уни-  
верситет  
cheti@cheti.cherkassy.ua

Повышение уровня интеграции электронных компонентов позволяет интегрировать на кристалле сложные функции проектируемых систем. В настоящее время имеется возможность реализации микроконтроллеров, памяти и специализированных устройств обработки данных на микросхемах программируемой логики (FPGA), содержащих от 5 до 30 тыс. логических вентилей. Однако изучение принципов функционирования сложных цифровых компонентов и разработка проектов на микросхемах FPGA требуют значительных временных затрат. С целью ускорения процесса проектирования используются готовые IP-ядра (Intellectual Property), реализованные в виде моделей отдельных компонентов цифровых систем.

Для повышения эффективности использования IP-ядер фирма Alatek создала новый инструментарий – IP CORE генератор, который позволяет пользователям адаптировать свои проекты для конкретных условий применения. Данный генератор интегрирован в среду Active-HDL и формирует тексты моделей отдельных функциональных модулей на двух языках описания аппаратных средств: VHDL и Verilog. Свойства проектируемых модулей могут определить пользователи перед началом проектирования программ моделей [1, 2].

IP CORE генератор обладает следующими особенностями:

- имеет дружелюбный интерфейс с пользователем, что позволяет выбрать нужный модуль, отрегулировать его параметры и сохранить файл с программой модели;
- сформированные коды оптимизированы со всеми популярными инструментариями синтеза электронных схем;
- имеется www-версия, которая позволяет работать с IP CORE генератором через Internet.

Все проектируемые объекты объединены в шесть групп:

комбинационные элементы, арифметические устройства, последовательностные схемы, фильтры, преобразователи кодов и промышленные контроллеры.

Исполняемый файл IP CORE Generator.exe можно загрузить с вебсайта [www.alatek.com](http://www.alatek.com). Систему ActiveX plug-in можно переписать с вебстраницы [www.esker.com](http://www.esker.com). Для работы с данным генератором требуется Internet Explorer 4.0 и выше или Netscape navigator 4/0. Требуется также 6 МВ свободного дискового пространства, операционная система – Windows 98/NT/2000.

В процессе установки IP CORE генератора к системе Active-HDL добавляется новая функция в меню инструментов. При помощи добавленной опции можно запустить генератор в открытой среде Active-HDL. Очень удобным и полезным является браузер IP CORE генератора, который обеспечивает стандартное представление объектов в виде папок и документов. Кнопки “плюс” и “минус” позволяют разворачивать и сворачивать дерево проекта.

Браузер IP CORE генератора дает возможность наладить связь с вебсайтом фирмы Alatek, где можно найти модернизированную версию данного генератора или дополнительные сетевые инструменты.

Представление проектов электронных компонентов на языках описания аппаратуры (Hardware Design Language) объединяет в себе возможность компактного представления очень сложных логических проектов с легкостью их понимания и небольшими затратами на их создание.

Использование IP CORE генератора позволяет повторно применять проектные решения и переносить их в различные среды, изменяя параметры проектируемых устройств, например, разрядность или подмножества реализуемых функций, а также осуществлять корпоративное проектирование программируемых цифровых систем.

### **Литература**

1. Вебсайт фирмы Alatek [www.alatek.com](http://www.alatek.com).
2. Вебсайт фирмы Aldec [www.aldec.com](http://www.aldec.com).

# АКТИВІЗАЦІЯ НАВЧАЛЬНО-ПІЗНАВАЛЬНОЇ ДІЯЛЬНОСТІ ПРИ ВИВЧЕННІ ПРОГРАМУВАННЯ З ВИКОРИСТАННЯМ МОДЕЛЬНО-СИМВОЛІЧНОЇ ТЕХНОЛОГІЇ

В.С. Єремєєв, С.М. Прийма

м. Мелітополь, Мелітопольський державний педагогічний університет

Реалізація основних положень Національної доктрини розвитку освіти України у ХХІ столітті вимагає оновлення освітньої парадигми, впровадження нових концепцій навчання, пошуку шляхів вдосконалення педагогічного процесу та посилення його технологічного аспекту. Особливої актуальності набувають вказані заходи у процесі вивчення такої фундаментальної науки як інформатика.

Дана публікація присвячена розробці основних принципів активізації самостійної навчально-пізнавальної діяльності студентів при вивченні інформатики, зокрема такого її розділу як основи алгоритмізації та програмування, з використанням модельно-символічної технології організації розвивального навчання.

В роботах автора даної педагогічної технології П.О. Барабохи показано ефективність та доцільність її використання у процесі вивчення дисциплін природничого циклу [1, 2].

Предметом нашого дослідження виступають шляхи активізації навчально-пізнавальної діяльності студентів вищого навчального закладу у процесі вивчення програмування на основі евристично-пошукової моделі, що реалізована використанням модельно-символічної технології.

Метою дослідження є обґрунтування ефективності та доцільності використання модельно-символічної технології у процесі вивчення курсу програмування.

Виходячи з мети дослідження, було сформульовано наступні завдання:

- розробити логіко-теоретичну основу модельно-символічної технології;

- виконати класифікацію проблемної символіки;
- розробити методичні рекомендації до організації лабораторних занять при вивченні курсу “Візуальне програмування”.

Система понять курсу програмування, що формується на фундаментальних основах інформатики, дозволяє краще відобразити глибинні причинно-наслідкові зв'язки, фундаментальні категорії дисципліни. Адже, будь-яке знання виражається в поняттях, категоріях, принципах, законах, закономірностях, символах, концепціях, теоріях. Відповідно, вільне володіння основними змістовними одиницями, які виступають орієнтиром для всього процесу вивчення курсу програмування, сприяє формуванню не тільки конкретних предметних знань, а й розвитку теоретичного, діалектичного мислення, що дозволить майбутнім фахівцям самостійно адаптуватися до швидкої зміни навколишнього інформаційного середовища.

Формуючи певне поняття предметного матеріалу, слід починати з такої логічної операції як поділ. Саме операція логічного поділу дозволяє розкрити обсяг поняття, розподіляючи його на види за певною ознакою. Чітке розуміння структури операції поділу, що включає ділене поняття, членів поділу та підставу, дозволяє уникнути помилок при виконанні даної операції [3, с. 154-155]. Поділ поняття на ділене поняття (рід) та види на підставі специфічного вияву ознаки в різних видах діленого називається поділом за видозмінюваною ознакою. Більш простішим діленням є дихотомічний поділ, за допомогою якого ділене поняття розподіляють на два суперечливі поняття. На практиці найчастіше використовують поділ за видозмінюваною ознакою, а дихотомічний поділ, за звичай, застосовують на початковій стадії вивчення поняття, коли є ясність відносно частини предметів, що обмежені діленням поняттям [4, с. 100].

Слід зауважити, що використовуючи будь-який з видів логічного поділу, необхідно дотримуватися певних правил, а саме:

- поділ поняття слід здійснювати за однією підставою;

- поділ повинен бути відповідним;
- члени поділу повинні виключати один одного;
- поділ повинен бути послідовним.

Отже, визначивши обсяг певного поняття, більш детальніше розглянемо членів його поділу, що надалі дозволять зрозуміти внутрішні взаємозв'язки та сутність поняття. Для цього скористаємося логічними відношеннями між поняттями.

Поняття, які мають спільну родову ознаку або спільне родове поняття називають порівнюваними [3, с. 142]. Наприклад, процедурне програмування – об'єктно-орієнтоване програмування, процедура – функція, цикл з передумовою – цикл з післяумовою, константа – змінна, інкапсуляція – наслідування, підпрограма – функція, сортування методом обміну – сортування методом вибору мінімального елементу. Якщо ж поняття не мають спільного родового поняття, то їх називають непорівнюваними. Наприклад, наукова дисципліна “інформатика” – мова програмування, комп'ютер – програма, об'єкт – об'єктно-орієнтоване програмування. Слід зауважити, що коли ми говоримо про непорівнюваність понять, то в цьому є доля умовності. Адже, визначаючи, що непорівнювані поняття не мають спільних родових ознак, ми вже порівнюємо їх за змістом. Окрім цього, при порівнянні можна завжди використати такий широкий рід як “предмет” чи “явище” чи “дещо”, який буде включати в себе об'єми будь-якого поняття, а вони по відношенню до нього будуть порівнюваними. Ось чому, говорячи про порівнювані та непорівнювані поняття, слід мати на увазі не потенційну можливість відшукування загального роду, а фактичну наявність в змісті понять спільних ознак, що безпосередньо визначають характеристики даного роду та дають підставу для ділення [4, с. 82].

Порівнювані поняття поділяють в свою чергу на сумісні і несумісні.

Сумісними називають поняття, видові ознаки яких забезпечують повне або часткове збігання їх обсягів.

Наприклад, програмування – об'єктно-орієнтоване програмування, підпрограма – процедура, цикл – цикл з умовою, цикл з умовою – цикл з післяумовою, програмне забезпечення – прикладне програмне забезпечення, функція – підпрограма, що

повертає значення результату через змінну, ім'я якої співпадає з її іменем.

Слід зазначити, що незважаючи на три види відношень (відношення тотожності, підпорядкування та часткового збігання), практичну цінність в умовах впровадження модельно-символічної технології, на нашу думку, має лише відношення підпорядкування. Дане відношення фіксує зіставлення родового і видового поняття. Наприклад, програмування – об'єктно-орієнтоване програмування, підпрограма – процедура, цикл з умовою – цикл з післяумовою. В даному випадку поняття, яке входить до обсягу іншого поняття, називають “підпорядкованим”, а поняття, яке вміщує у свій обсяг інше поняття, називається “підпорядковуючим” [3, с. 144].

Більший потенціал в межах модельно-символічної технології мають несумісні поняття, видові ознаки яких обумовлюють повне незбігання їх обсягів.

Несумісні поняття можуть знаходитися у трьох відношеннях:

- суперечності;
- протилежності;
- супідрядності [3, с. 145].

У відношенні суперечності знаходяться поняття, зміст яких заперечує один одного, а сума їх обсягів вичерпує обсяг родового поняття [3, с. 145]. Наприклад, значення логічної змінної true – значення логічної змінної false, процедура – функція, цикл з передумовою – цикл з післяумовою.

Протилежними називають поняття, зміст яких різниться у вищому ступені. Це означає, що вони виражають деякі протилежні характеристики у певному упорядкованому переліку властивостей, які постійно змінюються, і у свої сумі не вичерпують обсяг родового поняття. Наприклад, початкове значення лічильника – кінцеве значення лічильника.

Коли видові поняття одного роду не знаходяться ні у відношенні суперечності, ні у відношенні протилежності, а сума їх об'ємів становить лише частину об'єму родового поняття, то їм притаманне відношення супідрядності [3, с. 146]. Наприклад, цикл з передумовою – цикл з параметром, інкапсуляція – наслідування. Слід зауважити, що для відношення супідрядності



необхідна наявність більш загального, родового, поняття [4, с. 184].

Таким чином, знання логічних відношень між поняттями дозволяє краще збагнути зміст понять та допомагає визначити перелік операцій мислення, які можна виконати над поняттями, і, відповідно, уникнути помилок у побудові проблемно-символічних сигналів, що лежать в основі модельно-символічної технології.

Так, наприклад, до непорівнюваних понять можна застосувати команди впливу, взаємовпливу, взаємодії та встановлення причинно-наслідкового зв'язку. До порівнюваних понять, що знаходяться у відношенні підпорядкування, можна застосувати тільки команду встановлення причинно-наслідкового зв'язку. Побудова проблемно-диференційованого завдання, в основі якого знаходиться пара понять у відношенні підпорядкування, буде некоректною, якщо до нього застосувати команду порівняння (незважаючи на те, що ці поняття є порівнюваними).

Всі несумісні поняття, що знаходяться у відношенні суперечності, протилежності та супідрядності можуть комбінуватися з командами порівняння, впливу, взаємовпливу, взаємодії.

Для розуміння вищесказаного матеріалу скористаємося поняттями “цикл з передумовою” та “цикл з післяумовою”. Дані поняття утворилися в результаті поділу поняття “цикл з умовою” за такою ознакою як “місце умови”. В свою чергу поняття “цикл з умовою” є членом ділення поняття “цикл” за такою ознакою як “наявність умови”.

Враховуючи те, що поняття “цикл з передумовою” та “цикл з післяумовою” утворилися в результаті поділу родового поняття, вони будуть порівнюваними. Видові ознаки цих понять не забезпечують повного або часткового збігання їх обсягів, отже, поняття будуть несумісними. Можна стверджувати, що зміст одного з понять заперечує інше, а сума їх обсягів вичерпує обсяг родового поняття, а, отже, вони знаходяться у відношенні суперечності.

Встановлення відношення між даними поняттями дозволяє визначити одну з команд, наприклад, порівняння, та скласти

відповідне проблемно-диференційоване завдання, яке полягало б у порівнянні двох циклів з умовою, визначенні узагальнюючого слова (родового поняття) та 3 рис подібності і відмінності між ними, виділенні найголовніших з них (див. рис.1.).



Рис. 1 . Приклад проблемно-символічного сигналу

Подальший розгляд та формування поняття “цикл” можна здійснити на прикладі пари понять “цикл з передумовою” (“цикл з умовою”) та “цикл з параметром” (“цикл без умови”) (див. рис.2.).

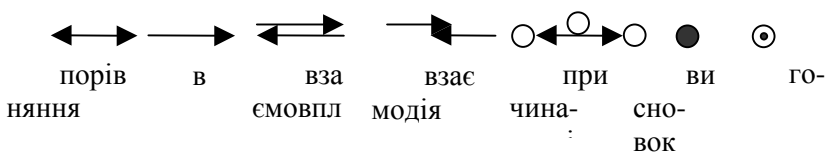
Таким чином, вказані проблемно-символічні сигнали реалізують план формування поняття “цикл”.

Слід зазначити, що масштабне впровадження технології вказало на необхідність класифікації символіки з метою її подальшого коректного використання. Для цього було проаналізовано блок символіки, визначено перелік основних символів (команди) та перелік додаткової символіки (параметри).

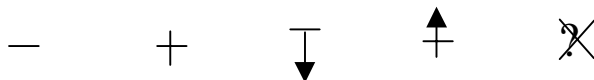


Рис. 2 . Приклад проблемно-символічного сигналу

До блоку команд ми віднесли символи, що відповідають загальним інтелектуальним операціям (порівняння, узагальнення, визначення причинно-наслідкових зв'язків та найголовнішого тощо):



Блок символів-параметрів, що доповнює та уточнює завдання, був складений з наступних символів:



подібність    відмінність    негатив    позитив    доказ    приклад

Символ К (в попередніх версіях конструктора проблемно-символічних сигналів символ З (три) ), що визначає кількісну

характеристику завдання, ми назвали ключем (опцією).

Наступний крок полягав у складанні можливих коректних комбінаціях, на базі яких було визначено загальну модель проблемної символіки:



– де квадратні дужки, за аналогією з форматом команд операційної системи MS DOS, вказують на обов'язковий елемент.

Наведена модель не тільки підтверджує правильність складеної моделі, але й дозволяють уникнути некоректності у поєднанні символів, що позитивно відображається на такому критерії технологічності як відтвореність.

Запропонований нами підхід до організації практичних робіт в умовах впровадження модельно-символічної технології базується на реалізації наступних етапів самостійності: від повного керівництва вчителя через дозовану допомогу до самокерування пізнавальною діяльністю, під час якої самостійність реалізується повністю. Звісно, процес проведення занять у відповідності до такого підходу вимагає від викладача високої кваліфікації, уваги до процесу організації самостійної діяльності, до логіки побудови та організації кожного заняття та всієї спільної діяльності [5, с. 456]. Реалізація основних принципів такого підходу передбачає кардинальні зміни в організації практичних робіт, зокрема, у вирішенні питань диференціювання. Відомо, що провідним принципом диференціювання освіти, а особливо в умовах впровадження розвивального навчання, повинно виступати не диференціація змісту завдань (одним-простіше, іншим-складніше), що домінує при традиційному навчанні, а диференціація допомоги студентам з боку викладача без істотного зниження складності змісту (одні студенти потребують більшої допомоги, інші - меншої). Слід зазначити, що завжди можна виділити групу студентів, котрим можна надати повну самостійність [6, с. 247-248]. Такий підхід дасть змогу кожному студенту досягти максимального прояву своїх можливостей та здібностей, творчого самоствердження та самовизначення.

Залишаючи традиційні три типи диференційованих рівнів А, В, та С, що передбачають визначення рівня оволодіння студентами знань, вмінь та навичок, слід більше уваги звертати на рівень самостійності.

Так, використання рівня А підносить студентів на рівень усвідомленого, творчого та подальшого застосування знань. Цей рівень передбачає вільне володіння фактичним матеріалом, прийомами навчальної роботи й розумових дій, можливість кожному студентові повністю виявити себе через самостійну пізнавальну діяльність, поміркувати над проблемою.

Робота над рівнем В передбачає осмислення й усвідомлення матеріалу. Але для оволодіння такими прийомами навчальних та розумових дій, які необхідні для вирішення питань програми рівня А, в програмі рівня В містяться загальні методичні рекомендації виконання пізнавальних завдань.

Рівень С передбачає засвоєння навчального матеріалу на рівні відтворення і включає багаторазове повторення, членування матеріалу на смислові групи, визначення головного, застосування прийомів запам'ятовування. В зміст цього рівня вводиться детальний інструктаж про те як навчатися, на що звертати увагу, який з вивченого впливає наслідок. При роботі на рівні С студентам дозволяється користуватися підручником, лекційним матеріалом та додатковою літературою, оскільки прямої відповіді на запитання там немає. Виконання рівня С передбачає ознайомлення із комп'ютерною реалізацією завдання – кінцевою метою кожного заняття (ніякої творчості, головна мета - досягти конкретного результату).

Запропонована нами методика організації практичних занять в умовах впровадження модельно-символічної технології полягає в наступному.

На початку вивчення курсу студентам пропонується базовий комплект до практичних занять, що містить *завдання до всього курсу, перелік практичних робіт із зазначенням теми та мети даного заняття, вимоги до оформлення практичних робіт, контрольні запитання для всіх трьох рівнів після кожної практичної роботи.*

Завдання до всього курсу. По-перше, завдання повинні бути практично значущими для студентів. Необхідно встановити

зв'язок навчання із життям та практичною професійною діяльністю. По-друге, завдання повинні бути творчими, тобто сприяти творчому розвитку кожної особистості.

Перелік практичних робіт із зазначенням теми та мети даного заняття. В цьому розділі рекомендовано більш детально вказати на завдання та кінцевий результат окремого заняття, звертати увагу на обов'язковий рівень реалізації (опис загальної структури, перелік функціональних можливостей та наявних компонентів). Слід звернути увагу на те, що цього матеріалу досить для роботи на рівні А.

Контрольні запитання для всіх трьох рівнів після кожної практичної роботи. Метою контрольного завдання рівня С є перевірка осмислення алгоритму виконуваних дій. Враховуючи те, що детальна інструкція складається з певної послідовності кроків, то завдання можна побудувати на визначенні негативних наслідків зміни даної послідовності. Пояснимо це на прикладі п.п. 7 (I<sub>7</sub>), 8 (I<sub>8</sub>) та 9 (I<sub>9</sub>) практичної роботи “Середовище швидкої розробки додатків Delphi. Створення головного та контекстно-залежного меню” курсу “Візуальне програмування”, що формулюється як:

*“... 7. Додайте до Вашої форми компонент класу TPopupMenu та, скориставшись Конструктором меню (Menu Designer), організуйте контекстно-залежне меню за зразком, що наведений у п.4.*

*8. За допомогою Інспектора об'єктів, встановіть значення властивості PopupMenu компоненту Main\_button в “PopupMenu1”.*

*9. Завантажте проект на виконання та, скориставшись натисненням правої кнопки миші на Головній кнопці, перевірте функціональність створеного Вами контекстно-залежного меню”.*

Зміст завдання відповідно буде наступним:

*“До яких негативних наслідків, що вплинуть на результат виконання завдання, призведе вилучення з інструкції практичної роботи п.8 (I<sub>8</sub>)?”*

Наведене завдання у формі проблемно-символічного сигналу матиме вигляд, зазначний на рис. 3.

Таким чином, виконання даного завдання передбачає розуміння виконуваних дій, їх взаємозв'язок, визначення необхідності дотримання саме такої послідовності. Все це сприяє не простому бездумному виконанню інструкції практичного завдання даного рівня, а осмисленню кожного етапу завдання.

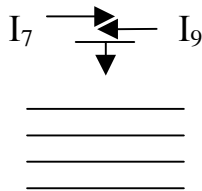


Рис. 3

Як було вказано раніше, програма рівня В містить перелік методичних рекомендацій до виконання завдання. Контрольне завдання даного рівня полягає у ретельному опрацюванні тексту цих рекомендацій, визначенні в ньому пари найголовніших понять, положень чи дій, адекватному підборі проблемно-диференційованого символу, узагальнюючого слова та самостійному складанні на їх основі проблемно-символічного сигналу з подальшим його вирішенням.

Для прикладу скористаємося наступним фрагментом рекомендацій щодо роботи з компонентами-меню:

*“Компонент класу TMainMenu визначає головне меню форми. На формі можна розташувати скільки завгодно примірників даного компонента, але відобразатися в смузї меню у верхній частині форми буде тільки той із них, що зазначений у властивості Menu форми. Натомість, компонент класу TRorirMenu використовується для створення контекстно-залежних меню. На відміну від головного меню, даний компонент може бути створений для будь-якого віконного компонента. Для пов'язування даного меню з будь-яким компонентом, необхідно у властивості RorirMenu компонента вказати ім'я компонента-меню. Робота щодо формування та редагування змісту меню аналогічна роботі з головним меню форми.”*

Завдання повинно бути сформульоване наступним чином:

*“На основі п.2 рекомендацій до виконання практичної роботи визначте пару термінів, підберіть до неї узагальнююче слово та виконайте порівняння термінів між собою з метою встановлення 3 рис подібностей та відмінностей.”*

Виконане студентами завдання у площині проблемної графіки матиме вигляд, зазначений на рис.4.



Рис. 4

Завдання даного рівня передбачає вміння опрацювати текст, вільне володіння основними положеннями використання проблемної символіки, самостійність при виконанні завдання.

Виконання програми рівня А вимагає від студента самостійного опрацювання теоретичного матеріалу, визначення алгоритму власних дій, застосування творчого підходу до вирішення проблеми. У зв'язку з цим контрольне завдання цього рівня полягає у перенесенні основних етапів власної роботи у площину проблемної символіки. Студенти самостійно аналізують суттєві етапи (на одне заняття – від 2 до 5 етапів) власних дій, встановлюють в них найголовніше та, аналогічно до завдання рівня В, складають проблемно-символічний сигнал. Під час виконання даного завдання слід рекомендувати студентам використовувати 2-5 проблемні символи, а від так і складати завдання на порівняння, взаємодію та встановлення причинно-наслідкових зв'язків.

Далі вказані методичні рекомендації до організації роботи студентів під час лабораторних занять з курсу програмування.

Безпосередньо практичному заняттю передують самостійна підготовка студентів. Маючи загальну направленість (завдання до всього курсу) та мету конкретного заняття студенти обирають стратегію його реалізації. Опрацьовується теоретичний матеріал підручника та лекцій, додаткова література; складається план роботи безпосередньо в аудиторії; опрацьовуються контрольні запитання та формується перелік запитань до викладача.



На початку лабораторного заняття студентам надається можливість отримати відповіді на запитання, які вони підготували заздалегідь. На даному етапі слід звернути увагу на те, що не викладач ставить запитання, відповіді на які в нього вже є, а самі студенти проявляють пізнавальну активність, шукаючи відповіді на питання, що виникли у них під час вирішення завдання.

Якщо опрацювавши матеріал та отримавши відповіді на свої запитання, студент все ж таки відчуває невпевненість в можливості самостійного виконання завдання (рівень А), то на занятті він отримує методичні рекомендації рівня В з його відповідними критеріями оцінювання.

Після отримання програми рівня В у студента з'являється вибір: самостійно виконувати завдання цього рівня або ж зробити запит детальної інструкції виконання завдання, і тим самим обрати рівень С.

Слід зазначити, що надання можливості вільного вибору рівня складності доцільне при свідомому ставленні студентів до рівня своїх знань. На початковому етапі бажано запровадити контрольні завдання, успішне виконання яких дозволяло б працювати на певному диференційованому рівні.

Завершальним етапом роботи студента на лабораторному занятті є опрацювання контрольних завдань, зміст яких був поданий нами раніше.

Переваги такої форми організації лабораторних робіт полягають в можливості особистості проявити такі індивідуальні риси мислення як самостійність та критичність; активізації самостійної навчально-пізнавальної діяльності, що проявляється в самостійній підготовці до кожного практичного заняття; можливості переходу студентів від одного диференційованого рівня на інший в залежності від підготовленості до певного заняття; економії навчального часу та раціонального його використання на заняттях за рахунок самостійної підготовки вдома; можливості вільного вибору, що сприяє розвитку критичного ставлення до своїх знань та відповідальності за зроблений вибір; обов'язковому контролю в кінці кожного заняття, що виключає можливість списування алгоритму розв'язку або результатів роботи.

Отже, в ході проведеного нами дослідження було отримано наступні результати:

- розроблено логіко-теоретичну основу модельно-символічної технології, на основі якої сформульовано нові принципи підбору опорних пар понять програмного матеріалу;
- виконано класифікацію проблемної символіки з визначенням переліку основних символів та сукупності додаткової символіки;
- складено коректні комбінації символів-команд з блоком символів-параметрів;
- визначено загальну модуль проблемної символіки;
- розроблено методичні рекомендації до організації лабораторних занять при вивченні курсу “Візуальне програмування” у вищому навчальному закладі.

Результати педагогічного експерименту підтверджують доцільність та ефективність застосування модельно-символічної технології при вивченні курсу програмування для активізації навчально-пізнавальної діяльності студентів, підвищення рівня знань та розвитку логічного мислення.

#### Література.

1. Барабоха П.А. Методика применения системы проблемно-графических сигналов (ПГС) в преподавании географии: Дис...канд.пед.наук: 13.00.02. – С.-Пб., 1993. – 158 с.
2. Барабоха П.А. Программа системного применения проблемно-символических сигналов (ПСС) в преподавании географии. – Уч.-метод. пособие. – К.: Реформа, 1998. – 48 с.
3. Конверський А.С. Логіка: Підручник для студентів вищих навчальних закладів. – К.: Український Центр духовної культури, 1999. – 400 с.
4. Логика: Учебное пособие / В.Ф. Берков, Я.С. Яскевич, В.И. Павлюкевич. – Мн.: ТетраСистемс, 1998. – 480с.
5. Кушниренко А.Г., Лебедев Г.В. 12 лекций о том, для чего нужен школьный курс информатики и как его преподавать. Методическое пособие. – М.: ЛБЗ, 2000. – 464 с.
6. Фурман А.В. Психолого-педагогічна теорія навчальних проблемних ситуацій: Дис... д.-ра психол.наук: 19.00.07. – К., 1993. – 449 л.

## **ИСПОЛЬЗОВАНИЕ СИСТЕМЫ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ В УЧЕБНОМ ПРОЦЕССЕ КАФЕДРЫ ФУНДАМЕНТАЛЬНЫХ ДИСЦИПЛИН**

В.И. Засельский, С.В. Швед

г. Кривой Рог, Криворожский металлургический факультет Национальной металлургической академии Украины

Компьютерные технологии развиваются и совершенствуются быстрыми темпами и все больше заполняют наш быт и трудовую деятельность.

Будущее промышленных технологий инженерных и научных разработок – за компьютерной обработкой всего потока информации. Однако говорить о переходе к качественно новому подходу в использовании компьютеров пока еще рано. Знания и навыки, приобретенные студентами с использованием компьютерных технологий, во многом будут определять пути развития общества.

В ВУЗах, научно-исследовательских институтах сосредоточено большое количество прогрессивных научно-педагогических кадров, которые способны внедрять новые информационные технологии в учебных заведениях, но без перестройки всего учебного процесса сделать это невозможно.

На наш взгляд, начинать работу с компьютером необходимо уже в первые дни учебы. Опыт показывает, что это должна быть комплексная работа, рационально включающая в себя как элементы «ручной» деятельности, так и деятельности с машиной. Если таким образом будет преподаваться каждая дисциплина (не исключая общественные), то компьютерная деятельность студента охватит широчайший диапазон изучаемых предметов.

Накопленный на кафедре опыт, например, при освоении курса теоретической механики, показывает, что, сочетая классические методы образования и новый подход, можно добиться значительного прогресса в уровне подготовки прилежных студентов, в конечном счете, создавая условия для нивелирования их будущих профессиональных возможностей, вплоть до самого высокого уровня.

Из опыта преподавания фундаментальных дисциплин следу-

ет отметить, что в направлении развития обучения студентов с использованием компьютерных технологий предстоит проделать еще много работы. Дело хотя бы в том, что все существующие базовые учебники разработаны задолго до бурного внедрения компьютеров в быт. В лучшем случае, они относятся к тому периоду времени, когда ученые и инженеры стояли в длинных очередях за получением «машинного времени». Поэтому все задачи в них рассчитаны на возможности калькуляторов. Бесспорно, среди этих задач есть бесценные по своему уровню воздействия на студентов. Однако на нынешнем этапе развития техники появляется возможность введения в практику расчетных работ студентов более сложных задач, расширяющих их технический кругозор. Эта возможность и является резервом методических разработок. Так, например, пользуясь возможностями нынешней компьютерной математики, можно быстро рассчитать поле траекторий системы материальных точек, находящихся под действием пространственной, гармонически изменяющейся системы сил. При этом компьютеру задается система векторных интегральных уравнений. Более того, средства анимации математических программ позволяют создать видеоролик возникающих колебаний. Производить решение такой задачи вручную в учебных целях невыносимо. Студенты могут увидеть виртуальную модель колебаний материальной точки в ее составном движении, наблюдать этот процесс, как в абсолютной системе координат, так и в относительной.

Весьма перспективной представляется концепция совмещенного лабораторно-компьютерного практикума, в рамках которого студент на каждом занятии выполняет дидактически законченный цикл лабораторных экспериментов, способствующий более полному раскрытию учебной темы. При этом часть экспериментов выполняется на реальном оборудовании, а часть - на компьютере с использованием программ имитационного моделирования, с максимальной степенью детальности воспроизводящих условия реального эксперимента для физики, химии, сопромата и других дисциплин.

Внедрение в учебный процесс системы AutoCAD (для дисциплины «Начертательная геометрия и инженерная графика») позволяет не только автоматизировать построение графических

моделей инженерной информации, но и преобразовывать и исследовать их.

В последнее время получила развитие автоматизация конструкторской деятельности на основе создания трехмерных геометрических представлений проектируемых изделий. А решение геометрических и других задач для построения модели обеспечивает большую достоверность, реально расширяет пространственный кругозор и позволяет перейти на качественно новый уровень проектирования.

Дальнейшее развитие компьютерных технологий в учебном процессе должно идти в направлении создания электронных версий учебников, лекций и т.д., в том числе интерактивных. Такие разработки позволяют, во-первых, добиться максимально высокого качества представления материалов, что не всегда удается при работе с доской и мелом, особенно при воспроизведении рисунков, и, во-вторых, дополнить лекционный материал большим количеством примеров, рассмотрение которых позволяет студенту не только уяснить практические методы применения теоретических положений, но и ознакомиться с решением многих классических задач, на решение которых в аудиторной работе не хватает времени.

Еще десятилетие назад студенческая аудитория разделялась на две четко выраженные группы: «практиков» и «теоретиков». Это разделение происходило самопроизвольно, соответственно уровню их математической подготовки. Такое разделение продолжалось и дальше, вплоть до перехода их в профессиональную деятельность. Нынешние студенты, обладающие компьютерной грамотностью, представляют собой более однородный коллектив. Даже «троечники» по классической «ручной» математике получают возможность производить сложные вычисления и реализовывать свои профессиональные навыки. Увеличение универсальности инженерных кадров в будущем непременно будет способствовать сращению науки с производством, а также поднимет качество подготовки специалистов на современный технический уровень.

## УЧЕБНЫЙ ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ, НАПРАВЛЕННЫЙ НА УГЛУБЛЕНИЕ ЗНАНИЙ ПО ТЕРМОДИНАМИКЕ

З.Г. Зуйкова

г. Донецк, Донецкий национальный университет

В курсе молекулярной физики высшей школы при изучении термодинамики доказывается теорема Карно: коэффициент полезного действия (к.п.д.) цикла Карно больше, чем к.п.д. любого равновесного цикла, у которого максимальная температура нагревателя и минимальная температура холодильника равны соответственно температуре нагревателя и температуре холодильника цикла Карно [1]. Опыт показывает, что студенты обычно воспринимают ее формально, как некое качественное утверждение, не приложимое к решению конкретных задач о работе тепловых машин. Сложившуюся ситуацию легко объяснить: вариантов циклических процессов, в которых к.п.д. можно рассчитать аналитически (именно они предлагаются в сборниках задач для практических занятий, например, [2]) немного. Говорить о «любом равновесном цикле», как упоминается в теореме, при этом не приходится. Величину к.п.д., для которого получаются аналитические выражения, сравнить с его значением для цикла Карно не всегда просто.

Представляется целесообразным вернуться к данному материалу в ходе изучения курса «Численные методы», предусмотренного в некоторых учебных планах (например, для специализации «Физика и основы информатики» физического факультета университета).

Студентам, уже изучившим интерполирование функций и численное интегрирование, и владеющим каким-либо языком программирования (например, PASCAL), предлагается составить программу для расчета к.п.д. произвольного цикла, совершаемого идеальным газом.

Цикл представляется в плоскости  $(V, p)$  (здесь  $V$  – объем,  $p$  – давление) в виде дискретного набора точек  $(V_i, p_i)$ ,  $i=1, 2, \dots, N$ , количество которых  $N$  можно менять в некотором диапазоне с целью повышения точности дискретизации. Для подсчета рабо-

ты газа при изменении объема от значения  $V_a$  до  $V_b$ ,

$$A = \int_{V_a}^{V_b} p dV$$

проводится интегрирование по методу Симпсона или трапеций [3]. На каждом шагу сетки вычисляется количество теплоты, полученное системой. Если оно положительно, то используется для подсчета количества теплоты, полученного в цикле от нагревателя. Параллельно рассчитывается абсолютная температура и находятся ее максимальное  $T_{max}$  и минимальное  $T_{min}$  значения. В результате расчета получаются два значения к.п.д.: для данного цикла  $\eta$  и для соответствующего ему цикла Карно

$$\eta_K = 1 - T_{min}/T_{max}$$

Особый интерес представляет вычислительный эксперимент с готовой программой: меняя произвольным образом форму изучаемого цикла (т.е. используя различные наборы точек  $(V_i, p_i)$ ), студент убеждается, что в любом случае расчет дает  $\eta < \eta_K$ , как и утверждает теорема Карно. Увеличивая число расчетных точек, можно получить результат с требуемой точностью и убедиться в стремлении отношения  $\eta/\eta_K$  к некоторому пределу, характерному выбранному циклу.

#### Литература

1. Румер Ю.Б., Рывкин М.Ш. Термодинамика, статистическая физика и кинетика. – М.: Наука, 1972. – 400с.
2. Иродов И.Е. Задачи по общей физике: Учеб. пособие. – М.: Наука, 1988. – 416 с.
3. Калиткин Н.Н. Численные методы. – М.: Наука, 1978. – 512 с.

## **ФОРМУВАННЯ КОНКРЕТНО-ПРЕДМЕТНИХ ЗНАТЬ З ІНФОРМАТИКИ ЯК ПЕРШООСНОВИ РОЗВИТКУ ПРОФЕСІЙНОЇ КУЛЬТУРИ СТУДЕНТІВ-АГРАРІЇВ**

О.В. Клочко

м. Вінниця, Вінницький державний аграрний університет

Застосування сучасних інформаційних технологій – показник, за яким можна визначити стан тієї чи іншої галузі знань. Без широкого застосування інформаційних технологій вже неможливо уявити ні наукову сферу, ні виробничу, ні науку управління. Сучасні комп'ютерні технології для розв'язання професійних економічних задач вивчаються студентами п'ятого курсу Вінницького державного аграрного університету (ВДАУ) в дисципліні “Автоматизація робочого місця спеціаліста” (АРМС). Предметом АРМС для спеціальності “Менеджмент організацій” сільськогосподарського вузу є галузь наукових знань про комп'ютерні технології ведення економічних розрахунків в агропромислових формуваннях, а саме про основи проектування, організації і експлуатації автоматизованих систем обробки економічної інформації. Під методом курсу слід розуміти сукупність способів, принципів і методичних прийомів проектування організації і експлуатації автоматизованих систем обробки інформації. Завдання курсу – навчання методам проектування, організації і експлуатації систем машинної обробки економічної інформації сільськогосподарських підприємств.

Зміст економічних задач залежить від характеристики їх складових: змісту інформації, алгоритму обчислень, структури вихідної документації тощо. Управління підприємством реалізується рядом загальних функцій: плануванням, обліком, економічним аналізом і регулюванням. Таким чином, склад економічних задач і процеси обробки їх даних є різноплановими як за інформаційним забезпеченням, так і за алгоритмом розв'язання. Разом з тим для них властивий певний інформаційний взаємозв'язок тому, що вони всі входять до складу задач управління сільськогосподарським виробництвом.

Даний курс ґрунтується на знаннях дисциплін: економічних, математичних, дисциплін що стосуються автоматизації та інфо-



рматизації обробки економічної інформації, основи комп'ютерних технологій обробки економічної інформації та досвід їх застосування в сфері економіки, основні принципи використання комп'ютерів, інших технічних засобів.

Організація викладання цієї дисципліни передбачає лекційний курс, аудиторні лабораторні та практичні заняття під керівництвом викладача, а також виконання самостійних індивідуальних завдань. Це забезпечує закріплення теоретичних знань та сприяє отриманню практичних навичок і розвитку самостійного наукового мислення.

О.М. Леонтєв підкреслював, що для того, щоб оволодіти знаннями і уміннями, необхідно здійснити діяльність, адекватну тій, яка втілена в цих знаннях і уміннях.

Ефективним засобом формування вказаних вище навичок є нові інформаційні технології, які розглядаються під час курсу АРМС.

Засоби розв'язання різних задач: текстові редактори, електронні таблиці, СУБД, генератори звітів і вивчаються на другому курсі на дисципліні “Програмне забезпечення” в розрізі прикладного застосування, тобто в напрямку автоматизації деяких задач, але вони не вивчається в розрізі автоматизації робочого місця спеціаліста, тобто автоматизації розв'язання комплексних задач. Не вивчається конкретне ППЗ: фінансові аналітичні системи, програми статистичної обробки даних, бухгалтерські системи. За 10 годин можна ознайомитись тільки з одним конкретним видом засобів. Тому доцільно збільшити кількість годин на вивчення даного курсу.

Слід відмітити необхідність коригування учбових програм в зв'язку з поновленням технічних та програмних засобів, які використовуються в учбовому процесі та відповідності програми комп'ютерних дисциплін сучасним вимогам.

Особливістю організації викладання цієї дисципліни є те, що застосування ППЗ сьогодні треба розглядати не відокремлено, а як предмет, тісно пов'язаний з основами менеджменту, маркетингу, обліку, аналізу, фінансовою діяльністю та бізнесом. Використання сучасної комп'ютерної техніки та програмних засобів залежить не тільки від рівня комп'ютерної грамотності, а й від уміння використовувати сучасне ППЗ для вирішення практичних

задач виробництва. Тому майбутні економісти повинні розуміти відмінності інформації від даних, знати сутність форм адекватності інформації (синтаксична, семантична і прагматична), уміти оцінити інформацію як за якісними, так і кількісними характеристиками, мати уявлення про систему класифікації та кодування інформації, розуміти сутність і цілі інформатизації суспільства, знати структуру ринку інформаційних продуктів і послуг, сутність інформаційних технологій опрацювання даних, управління, автоматизації офісу, підтримки прийняття рішень.

Особливу увагу при вивченні дисципліни АРМС слід приділяти спеціалізованим програмним продуктам, орієнтованим на дослідження готових економічних моделей. Project Expert фірми PRO-INVEST CONSULTING є програмою, що втілює імітаційну модель, за допомогою якої можна відтворити діяльність компанії. Програма Project Expert, як інструмент фінансового аналізу, виконує дві основні функції: по-перше, переводить опис діяльності підприємства з мови користувача у формалізований опис грошових потоків; по-друге, обчислює показники, по яких фінансовий менеджер може судити про результативність прийнятих рішень. Програма Project Expert призначена для використання на комп'ютерах IBM PC або сумісних з ними. Інтерфейс програми реалізований у стилі MS Office 97. Програма підтримує роботу з Інтернет.

Найважливішим результатом застосування програми Project Expert для розробки інвестиційного проекту є створення бізнес-плану, що задовольняє стандартам UNIDO. United Nations Industrial Development Organization (UNIDO) займається виробленням стратегії економічного розвитку країн з перехідною економікою. Одним з результатів її діяльності є створення стандартів підготовки інвестиційних рішень.

Project Expert підготує фінансові звіти, по яких можна визначити стан компанії в будь-який момент часу. При цьому балансовий звіт і звіт про прибутки і збитки формуються у відповідності зі стандартами IAS, загальноприйнятими в міжнародній практиці. International Accounting Standard (IAS) визначає вимоги до бухгалтерського обліку, прийняті в більшості розвинутих країн. При висновку міжнародних угод, представлення звітності компанії в стандарті IAS є обов'язковим.

Таким чином, вивчивши курс Project Expert студенти опановують знаннями структури фінансово-аналітичних систем, їх можливостей, схемою обробки інформації в них, а також набувають вмінь:

- розробляти детальний фінансовий план і визначати потреби в коштах на перспективу;
- визначати схему фінансування підприємства, оцінювати можливість і ефективність залучення коштів із різних джерел;
- розробляти план розвитку підприємства, реалізації інвестиційного проекту, визначивши найбільш ефективну стратегію маркетингу, а також стратегію виробництва, що забезпечує раціональне використання матеріальних, людських і фінансових ресурсів;
- проробляти різні сценарії розвитку підприємства, варіюючи значення факторів, що можуть впливати на його фінансові результати;
- формувати стандартні фінансові документи, розрахувати найбільш розповсюджені фінансові показники, проводити аналіз ефективності поточної і перспективної діяльності підприємства;
- підготувати бездоганно оформлений бізнес-план інвестиційного проекту.

У Project Expert реалізований аналіз проекту за методом Монте-Карло. Розрахувавши проект із використанням очікуваних значень цін, обсягів продажу, тимчасових факторів, студенти проводять додатковий аналіз.

Для будь-якого продукту, що входить у проект, проводиться аналіз точки беззбитковості – визначається мінімальний обсяг виробництва, що забезпечує окупність даного продукту.

За допомогою модуля What-If студенти аналізують різні варіанти проекту.

Наприклад, перед студентом ставиться завдання:

На основі отриманого варіанту побудувати імітаційну фінансову модель підприємства на 2 роки за допомогою програми Project Expert. Модель має бути побудована за схемою, наведеною на рис. 1.

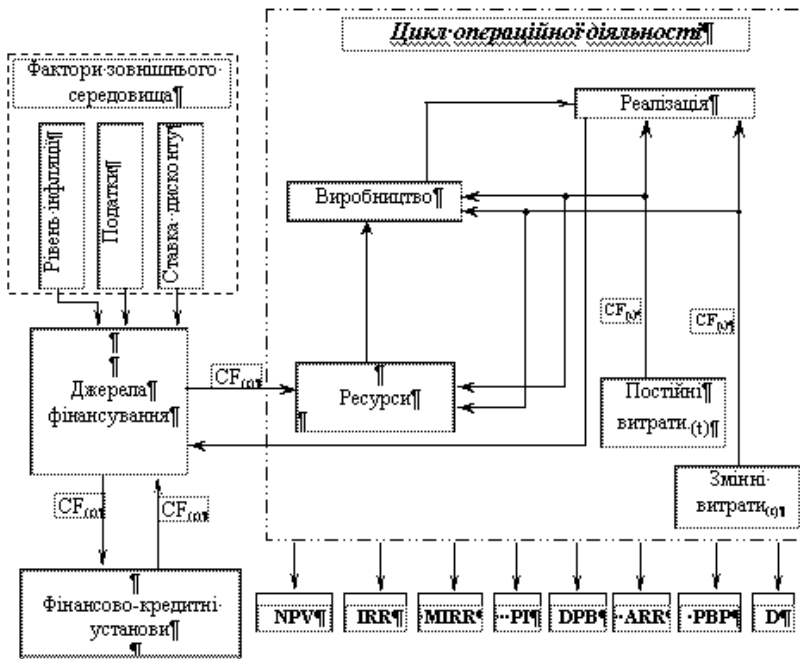


Рис. 1. Модель фінансової діяльності підприємства

Для виконання завдання студент повинен:

1. Визначити фактори зовнішнього середовища.
2. Визначити джерела фінансування і фінансово-кредитні установи.

3. Сформулювати цикл операційної діяльності.

Наприклад. Продукція, що виробляється підприємством: - ячмінь; гречка (рис. 2).

Продукты/Услуги		
Наименование	Ед. изм.	Нач. продаж
▶ Ячмінь	кг	01.03.2003
Гречка	кг	01.03.2003

Buttons: OK, Отменить, Справка

Рис. 2. Продукція, що виробляється підприємством.

В частині *ресурси* студент описує:

- номенклатуру ресурсів для виробництва;
- кількість ресурсів в розрізі номенклатури і вартість за

одиноцю.

Ресурси:

№ п/п	Вид ресурсів	Одиниця виміру	Вартість за одиницю
1	Дизельне паливо	Л	0,85 грн.
2	Нітрофосфор (добрива)	Кг	2,80 грн.
3	Насіння гречки	Кг	3,60 грн.
4	Насіння ячменю	Кг	1,00 грн.
5	Селітра	Кг	0,32 грн.
6	Техніка з інших господарств	грн./га	35 грн.

В частині *виробництво* студент має описати технологічний цикл виробництва продукції (по тривалості технологічних операцій) і визначити його вартість на основі технологічних карт, вартості і кількості ресурсів, а також нормативно-довідникових джерел. Технологічна карта має бути надана у звіті.

Опис технологічних операцій на один рік:

Назва	Тривалість (днів)	Вартість операції (грн.)
Боронування зябу (ячмінь та гречка)	5	<b>780,37</b>
Внесення мінеральних добривши	3	<b>5933,92</b>
Культивація	2	<b>350</b>
Посів гречки	5	<b>27921,69</b>
Посів ячменю	3	<b>7492,97</b>
Внесення фосфорних добрив	3	<b>301,43</b>
Збір врожаю (ячмінь)	4	<b>533,86</b>
Збір врожаю (гречка)	3	<b>710,04</b>
<b>Перевезення та складування</b>	2	<b>133,47</b>

4. Студент вивчає основні критерії прогнозування фінансової діяльності підприємства.

5. Після проектування моделі і вивчення основних показників ефективності проекту студент представляє схему моделі фінансової діяльності підприємства в розгорнутому вигляді з розшифровкою по кожному окремому блоку. Всі дії занотуються у звіт по підпунктам.

Система АРМ сприяє інтегруванню різних видів обліку (бухгалтерський, внутрішньогосподарський (управлінський), статистичний, податковий) у єдину систему господарського обліку, що дає змогу уникнути дублювання показників, підвищує оперативність і ефективність використання даних обліку в управлінні підприємством.

Економіст отримує інформацію з бухгалтерської документації. Тому спеціалісту-економісту доцільно вивчати комп'ютерні системи бухгалтерського обліку. Компонентами бухгалтерських програм є "1С Бухгалтерія Проф. для Windows", "Парус". Комп'ютерні системи бухгалтерського обліку (КСБУ) розрізняються за повнотою та інтеграцією облікових функцій: від засобів автоматизації локальної задачі бухобліку до повнофункціональних КСБУ в складі інформаційної системи підприємства.

Отже, метою курсу АРМС є також вивчення можливостей електронних таблиць для розвитку дослідницьких навичок, формування критичного стилю мислення.

Незаперечною перевагою MS Excel є можливість застосування графічного способу рішення багатьох завдань, який дозволяє отримати результат без математичних розрахунків (рис. 3).

**Кореляційне поле**

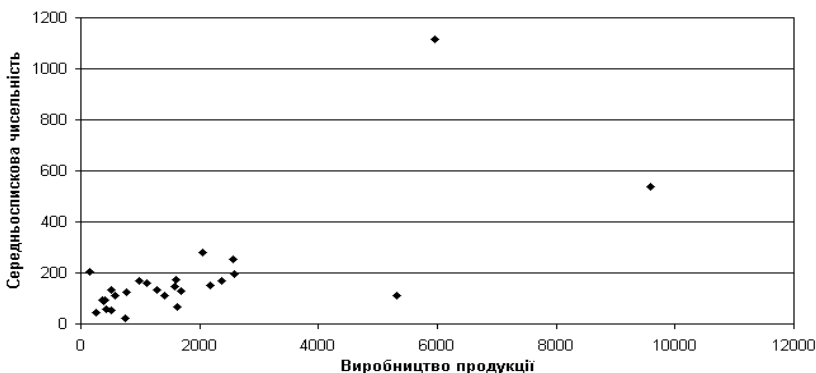


Рис. 3. Кореляційне поле по виробництву продукції і середньоспискової чисельності підприємств.

Цей метод рішення демонструє переваги комп'ютерної гра-

фіки, причому використовуються її ілюстративна та когнітивна (що сприяє пізнанню) функції. Ілюстративна функція надає можливість вирішити задачу навіть студентам із слабкою математичною підготовкою, забезпечує економію часу (що є важливим чинником в навчальній діяльності), дозволяє зосередитися на аналізі отриманого рішення, формулюванні відповідних висновків. Когнітивна функція полягає в тому, що за допомогою графічного зображення можна отримати нове знання – про тенденцію показника, що досліджується, його наступні значення (наприклад, за допомогою лінії тренда). Загалом, застосовуючи графічні можливості MS Excel, студенти не тільки набувають знань про використання діаграм та графіків у табличних процесорах, побудову лінії тренда, а також отримують навички прогнозування та формулювання гіпотез відносно закономірностей, що досліджуються, уміння вирішувати задачу різними методами; вчать аналізувати одержані в обчислювальних та графічних експериментах результати, робити висновки.

Наприклад: Студент повинен розрахувати собівартість продукції (цукру) за допомогою електронних таблиць Microsoft Excel (рис. 4).

Microsoft Excel - Книга1		
Файл Правка Вид Вставка Формат Сервис Данные Окно ?		
Arial Cyr 10 Ж К Ц [буліт] [квадрат] [квадрат] [квадрат] % , +,00 +,00		
B8 =		B7
A		
1	НВп - незавершене виробництво на початок калькуляційного періоду, грн.;	15000
2	ВВп – виробничі витрати періоду (місяць, квартал, сезон, рік), грн.;	2000
3	ЗВ – зворотні витрати і відходи, грн.;	1000
4	ПП – побічна продукція, грн.;	3000
5	СП – супутна продукція, грн.;	2000
6	БП – брак продукції;	1000
7	НВк – незавершене виробництво на кінець калькуляційного періоду, грн.	4000
8	<b>ВСП = НВп + ВВп - ЗВ - ПП - СП + БП - НВк , грн.,</b>	<b>10000</b>
9		

Рис. 4. Розрахунок виробничої собівартості цукру.

Виробнича собівартість продукції (ВСП) розраховується за формулою:  $ВСП = НВп + ВВп - ЗВ - ПП - СП + БП - НВк$ , грн., де НВп – незавершене виробництво на початок калькуляційного періоду, грн.; ВВп – виробничі витрати періоду (місяць, квартал, сезон, рік), грн.; ЗВ – зворотні витрати і відходи, грн.; ПП – по-

бічна продукція, грн.; СП – супутна продукція, грн.; БП – брак продукції; НВк – незавершене виробництво на кінець калькуляційного періоду, грн.

Процес побудови і дослідження моделей активізує та розвиває дослідницький стиль мислення студентів.

Обґрунтування вибору оптимальних рішень із запропонованих програмою потребує вміння виявляти та враховувати вплив всіх умов та чинників. Важливим є те, що під час виконання завдання студенти набувають навичок моделювання – одного з найважливіших навичок економіста, вчатья висувати різні варіанти рішення, порівнювати, оцінювати, виявляти недоліки і переваги кожного, вибирати оптимальний. У процесі колективного обговорення моделей виробляються уміння аргументовано доводити свою точку зору, враховувати чужу думку, тобто формулюються комунікаційні якості, навички ділового спілкування, культура мови. Моделювання сприяє розвитку творчих здібностей студентів, вчить пошуку засобів перевірки гіпотези. Якщо в процесі моделювання отримані дані, які не відповідають побудованій моделі, тобто модель реалізовано з помилками, - тобто необхідно переглянути побудовану модель або досліджувати її іншими методами. Таким чином студенти отримують навички дослідження.

Крім розвитку навичок пошуку інформації, вміння формулювати запити, визначати критерії та області пошуку, сортувати та аналізувати знайдену інформацію, виконання запропонованих завдань сприяє формуванню рефлексивного мислення, тобто усвідомлення необхідності перевірки, аналізу отриманого рішення, порівняння його з передбачуваним результатом. Ці якості є одними з найважливіших показників дослідницького, критичного стилю мислення, навички якого, як показує практика, не вироблені у багатьох студентів.

Розроблений комплекс завдань спрямований на досягнення декількох педагогічних цілей:

– навчально-пізнавальну: накопичення фонду базових знань, умінь і навичок роботи в середовищі однієї з найбільш масових інформаційних технологій – табличних процесорів; знайомство з сучасними методами представлення, зберігання, пошуку, обробки економічної інформації;



– пропедевтичну: освоєння інструменту, знаряддя праці, що закладає основу широкого використання комп'ютерних технологій в подальшій навчальній та професійній діяльності. Реальний практичний зміст завдань забезпечує ілюстрацію практичної цінності і значущості набутих знань;

– розвивальну: підвищення інтелектуального рівня студентів, формування дослідницьких вмінь та навичок, розвиток критичного мислення.

Студенти на лабораторних заняттях навчаться створювати ключі в системі PGP, передавати відкритий ключ PGP кореспондентам, передавати захищені і підписані повідомлення за допомогою системи PGP, відкривати захищені в підписані повідомлення за допомогою системи PGP, шифрувати дані на жорсткому диску за допомогою системи PGP.

Оволодівши знаннями курсу АРМС, спеціаліст зможе працювати з ПЕОМ, використовувати отримані знання для розв'язання задач обліку, аналізу, планування, при вивченні спеціальних дисциплін, курсового та дипломного проектування.

Таким чином, досвід викладання дисциплін комп'ютерного циклу при підготовці економістів АПК в ВДАУ показав, що вирішення завдань навчання сучасного спеціаліста потребує не тільки застосування новітніх технічних та програмних засобів, а й підтримки рівня їх викладання у відповідності з вимогами часу.

#### Література

1. Джонассен Д.Х. Компьютеры как инструменты познания: изучение с помощью технологии, а не из неё. // Информатика и образование. – 1996. – №4. – С. 117-125.
2. Дусавицький О.К. Система розвивального навчання: засади становлення // Початкова школа. – 1996. – №11. – С. 4-7.
3. Кларин М.В. Столкновение с проблемой: Обучение исследованию путем выдвижения и проверки гипотез. // Директор школы. – 1995. – №5. – С. 66-72.
4. Орвис В. Дж. EXCEL для учёных, инженеров и студентов: Пер. с англ. – К.: Юниор, 1999. – 528 с.
5. Полат Е.С. Новые педагогические технологии и компьютерные коммуникации // Изд. центр «Академия», 2000. – 224 с.

6. Разумовский В.Г. Обучение и научное познание // Педагогика. – 1997. – №1. – С. 7-13.
7. Роберт И.В. Современные информационные технологии в образовании: дидактические проблемы, перспективы использования. – М.: Школа-Пресс, 1994. – 205 с.
8. Уваров А.Ю. Новые информационные технологии и реформа образования // Информатика и образование. – 1994. – №3. – С. 3-14.

## **ПРИМЕНЕНИЕ ТЕХНИЧЕСКИХ СРЕДСТВ ОБУЧЕНИЯ ПРИ ИЗЛОЖЕНИИ ДИСЦИПЛИН КОМПЬЮТЕРНОГО ПРОФИЛЯ**

Г.К. Кожевников, В.В. Сосновский, Л.А. Панова  
г. Харьков, Украинская инженерно-педагогическая академия

Повышение педагогической эффективности учебного процесса невозможно без применения технических средств обучения (ТСО). Используя традиционные средства преподавания (мел, доска) преподаватель вынужден тратить часть занятий на изображение графической или текстовой информации, что снижает производительность педагогического труда. В настоящее время при преподавании дисциплин технического профиля возникает необходимость подачи большого объема информации за ограниченный промежуток времени. Вместе с тем изложение дисциплин компьютерного профиля имеет ряд особенностей. Во-первых, для доступного изложения материала необходима демонстрация большого количества копий экрана компьютера (чтобы убедиться в этом, достаточно открыть любую книгу по компьютерной тематике). Во-вторых, обучаемым важно проследить за динамикой изменения результатов работы рассматриваемого программного приложения. В-третьих, обучаемые должны видеть реакцию программных приложений на действия пользователя. Исходя из перечисленных выше особенностей применения ТСО, при изложении дисциплин компьютерного профиля необходимо определить набор технических средств для решения данной дидактической задачи и разработать методику применения выбранных технических средств с учетом их специфики.

Наиболее целесообразно для решения данной задачи использовать следующие устройства:

- 1 – Графопроектор с пакетом фолиограмм;
- 2 – Комплекс графопроектор + жидкокристаллическая проекционная панель + персональный компьютер (ПК).

Применение графопроектора с пакетом фолиограмм (вариант 1) даёт возможность наложения изображений. Используется основная фолиограмма и несколько вспомогательных, накладываемых поверх основной.

При использовании второго варианта возможно отображать на большом экране всё, что происходит на экране ПК по мере работы с программным приложением.

Недостатком первого случая является ограниченный объём фолиограмм и невозможность отображения всех нюансов работы с программным приложением

Во втором случае недостатком является громоздкость и достаточно высокая стоимость комплекса ТСО.

При демонстрации копий экранов компьютерного изображения необходимо определить оптимальные размеры экранной проекции. Так, например, оптимальное расстояние для восприятия информации с экрана компьютера составляет  $D = (3,3 * Q \div 9 * Q)$ , где  $Q$  – диагональ экрана телеприёмника или монитора [1, с. 114]. Тогда несложные расчеты показывают, что при использовании в качестве ТСО компьютера с монитором 32 см по диагонали позволяет обеспечить ясное восприятие информации не более чем для 2 – 4 студентов.

В [2, с. 34] предлагается формула описывающая соотношение размеров проекционного экрана и длины учебного помещения

$$H = \left( \frac{1}{5} \div \frac{1}{6} \right) * L, \quad (1)$$

где  $H$  – высота экранного изображения, а  $L$  – длина учебного помещения. В свою очередь от размеров экранного изображения зависит его яркость и, соответственно, ясная видимость изображения с последнего ряда зрителей. Яркость изображения, формируемого графопроектором, рассчитывается по формуле  $F = K * \Phi$ , где  $\Phi$  – полезный световой поток проектора,  $K$  – коэффициент снижения яркости изображения. Данный коэффициент может быть рассчитан по формуле  $K = e^{-\lambda * S_{\text{э}}}$ , где  $\lambda$  – коэффициент, учитывающий прозрачность дидактического материала и тип проекционного аппарата,  $S_{\text{э}} = b * H$  – площадь экранного изображения,  $b$  – ширина экранного изображения. Поскольку наиболее удобным для наблюдения является соотношение высоты и ширины экрана (зрительного поля) соответственно  $H : b \approx 3 : 4$ , то величина  $b$  может быть определена как

$$b = \frac{H * 4}{3} = 1,33 * H.$$

Ниже приведен график зависимости яркости экрана при использовании графопроектора «Литер 1400» от размеров экрана.

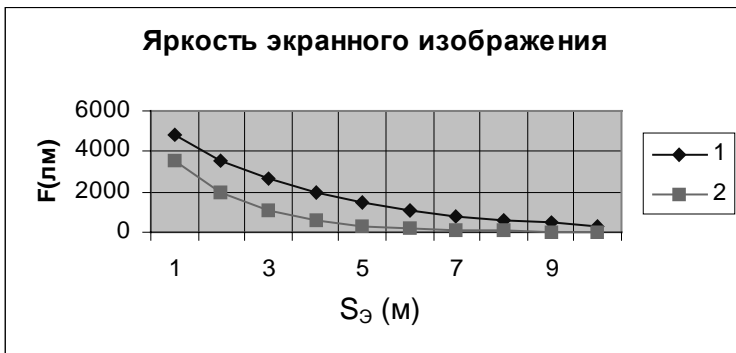


График 1 рассчитан для случая затемненного помещения, график 2 – для незатемненного помещения. Используя эти графики, можно определить размеры экрана, обеспечивающие ясную видимость изображения. Тогда на основании формулы (1) можно рассчитать предельные размеры аудитории, в которой можно проводить занятия с использованием данного аппарата статистической проекции. Таким образом, при изложении дисциплин компьютерного профиля будут обеспечены такие необходимые параметры дидактических материалов (ДМ) как достаточные размеры изображения, обеспечивающие его ясную видимость с последнего ряда зрителей, и высокая контрастность изображения.

Такие параметры ДМ, как наглядность, фрагментарность, полное использование площади и применение цветовой гаммы [2. с. 36-37], обеспечиваются в данном случае за счет использования копий экранов компьютера. При этом копии экрана не должны быть перегружены сложной детализировкой и пояснениями. Если требуется разместить большое количество учебного материала, следует использовать несколько копий экрана.

При подготовке в качестве ДМ копий экранного изображения наряду с определением размеров экранного изображения необходимо определить размер шрифта на экране компьютера. При размещении текстовой информации на проекционном экра-

не основным параметром является высота символа. Данная величина может быть определена по формуле 2 [3 с. 72];

$$H_{\text{э}} = 0,005 * L \quad (2)$$

Если предположить, что высота символа на экране ( $H_{\text{э}}$ ) связана с высотой символа на ДМ  $h_{\text{ом}}$  так же, как ширина экранного изображения ( $b$ ) с шириной носителя информации ( $b_1$ ), то, используя выражение  $\frac{b}{b_1} = \frac{p}{f}$ , где  $p$  – расстояние между экраном и аппаратом,  $f$  – фокусное расстояние аппарата [1 с. 91], можно получить формулу (3) позволяющую вычислить высоту символа на ДМ:

$$h_{\text{ом}} = \frac{H_{\text{э}} * f}{p} \quad (3)$$

Приведенные выше рекомендации по применению ТСО для изложения дисциплин компьютерного профиля могут быть полезны преподавателям компьютерных дисциплин в ВУЗах, школах, ПТУ, техникумах и коллежах, а также для преподавателей компьютерных курсов и курсов повышения квалификации.

#### Литература

1. Мелецинек Адольф. Инженерная педагогика. – М.: МАДИ(ТУ), 1998. – 185 с.
2. Молибог А.Г., Тарнопольский А.И. Технические средства обучения и их применение. [Учебное пособие для преподавателей и учебно-вспомогательного персонала высших и средних специальных учебных заведений по курсу «Технические средства обучения и методика их применения»]. – Мн.: Изд-во «Университетское», 1985. – 208 с.
3. Уманский В.С. Применение технических средств обучения в учебном процессе средних профтехучилищ. – М.: Высш. школа, 1979. – 151 с.

## РОЗМІЩЕННЯ ЗАДАЧ НА МОДЕЛЮВАННЯ В ШКІЛЬНОМУ КУРСІ ІНФОРМАТИКИ

В.В. Коломенська

м. Донецьк, Донецький національний університет

Шкільний курс “Основи інформатики і обчислювальної техніки” в останні роки вийшов на якісно новий етап свого розвитку. За комп’ютерною грамотністю в наш час стоїть проблема ефективного використання сучасних комп’ютерних технологій, а не лише вміння програмувати. Зараз уже практично всіма усвідомлено, що шкільна інформатика не повинна бути курсом програмування. Однак так само ж помилково було б низводити курс інформатики тільки до чисто орудійного використання комп’ютера: редагування текстів, малювання та ін. Тоді інформатика швидко загубила б своє значення як самостійна навчальна дисципліна.

Оволодіння засобами комп’ютерних інформаційних технологій є тільки першою практичною задачею інформатики. Комп’ютерна техніка, інформаційні системи та технології, програмування – це всього лише засоби для розв’язку прагматичних задач, тобто для інформаційного моделювання в самому широкому розумінні. Моделювання по суті і є кінцевою метою інформатики.

Процес навчання моделюванню в шкільному курсі інформатики має і такий важливий аспект: грамотний фахівець в будь-якій області повинен уміти поставити задачу, тобто так здійснити її формалізацію, щоб суть задачі однозначно зрозумів кожний, хто буде брати участь у її розв’язку.

У програмі шкільного курсу інформатики в темі “Інформаційна модель” на знайомство з поняттям інформаційної моделі і принципами її побудування відведено лише дві години. Цього часу досить лише для формального визначення основних термінів. Більш того, надзвичайна загальність поняття інформаційної моделі і багатогранність технологій моделювання потребує, щоб в цьому питанні практика випереджала теорію: спочатку учень працює з моделями, потім з допомогою вчителя усвідомлює суть своєї діяльності як моделювання, і лише після цього

обговорюється зміст поняття моделі, проводиться класифікація моделей і розглядаються основні етапи моделювання.

В узгодженні з вказаним, модельні задачі необхідно вводити в курс інформатики ще до розглядання теми “Інформаційне моделювання”. Спочатку вони слугуватимуть допоміжним засобом для демонстрації можливостей прикладних програм або для з’ясування принципу дії алгоритмічних конструкцій. Такі модельні задачі вводяться поступово, на протязі всього курсу, тому засоби для їх розв’язку повинні відповідати меті навчання і рівню знань учнів. Крім того, такі задачі мають бути нескладними (відповідні програмні або алгоритмічні засоби ще тільки засвоюються учнями) і мати виключно чітку постановку (моделювання грає тут все ж таки підлеглу роль). Отже, подібні модельні задачі вводяться одночасно і для надання матеріалу, що вивчається, практичного змісту, і для поступового знайомства з основними поняттями моделювання: “об’єкт”, “модель”, “система”, “елемент” та ін.

Безпосередньо до теми “Інформаційне моделювання” увійдуть теоретичні основи технології моделювання і більш складні модельні задачі. По-перше, це задачі, основною метою яких є вивчення властивостей моделі: задачі з інших дисциплін, більш складні прикладні задачі. По-друге, це задачі, пов’язані з моделюванням власних об’єктів інформатики. Тут моделі редакторів, електронних таблиць, та ін. розглядаються з точки зору принципу їхньої дії як програм. По-третє, це нечіткі прикладні задачі, розглянуті з метою навчити саме моделюванню. При їх розв’язку учень знайомиться з основними етапами моделювання: постановка задачі, пошук методів та комп’ютерних засобів для її вирішення, аналіз результатів, визначення похибок.

Таким чином, інформаційне моделювання із часткового питання перетворюється на фундамент шкільного курсу інформатики. Питання про місце конкретної задачі в курсі вирішується в залежності від цілей навчання, а також від властивостей задачі і моделі.



# ФОРМУВАННЯ ІНФОРМАЦІЙНОЇ КУЛЬТУРИ МАЙБУТНІХ ЕКОНОМІСТІВ НА ОСНОВІ ПРЕДМЕТНОГО МОДЕЛЮВАННЯ

М.Г. Коляда

м. Донецьк, Донецький інститут соціальної освіти

## **1. Вступ**

Однієї з важливих задач вищої школи є розробка стандартів навчання. У зв'язку з цим особливого значення набуває побудова моделі фахівця. У сформованій у сьогоденні термінології, ця робота стосується моделювання студента.

Під моделлю студента, розуміють знання про студента: знання про те, який він є зараз, знання про те, яким ми хочемо його бачити, і знання про те, яким ми можемо його побачити [1, 2]. Уявлення про те, яким повинний стати студент, чи, передбачуваний образ студента, після закінчення навчання у вузі, визначена, як *нормативна модель студента*. Вона збігається з моделлю фахівця.

Частина нормативної моделі студента, що визначає систему умінь і знань по окремому навчальному предметі одержала назву *предметної моделі студента*. Предметна модель студента, – це багатокomпонентне утворення. Її складовими виступають *тематичний, семантичний, процедурний, операційний і функціональний* компоненти. *Тематичний* предметний компонент предметної моделі дає уявлення про структуру навчального предмета, *семантичний* передає її зміст, *функціональний* показує, яку роль грають ті чи інші предметні знання, *процедурний* описує порядок і характер перетворення об'єктів предметної області, а *операційний* визначає уміння, що повинні бути засвоєні при вивченні курсу. Кінцевою метою навчання є освоєння способу дій, що реалізується через уміння [1], тому особливого значення набувають процедурна й операційна предметні моделі. Інші моделі, до складу яких входять знання й метазнання [2], відіграють допоміжну роль, забезпечуючи формування умінь.

У Донецькому інституті соціальної освіти ведеться комплексна робота з моделювання студентів по різних дисциплінах, зокрема, по інформатиці для економістів. У роботах [3; 4] опубліко-

вані тематична й операційна предметні моделі при формуванні інформаційної культури майбутніх економістів, ця робота, присвячена експериментальній перевірці результативності використання предметного моделювання.

## ***2. Педагогічний експеримент по визначенню ефективності формування інформаційної культури майбутніх економістів***

Такий експеримент проводився в Донецькому інституті соціальної освіти (ДІСО) і в Донецькому державному університеті економіки і торгівлі ім. Туган-Барановського з 1998 по 2002 рік. У ході експерименту були виділені експериментальні і контрольні групи. В експериментальних групах вивчення загальної й економічної інформатики велося із застосуванням нової структури побудови курсів «Інформатика та комп'ютерна техніка» і «Інформаційне забезпечення аналізу даних і прогнозування в економіці», а також із застосуванням усього арсеналу методичних і організаційних прийомів і методів використовуваних при діяльній підході в навчанні на основі предметного моделювання.

У ході першого експерименту була простежена динаміка росту формування основних знань і умінь студентів по інформатиці. При вивченні базових питань інформатики, кожен студент мав можливість проаналізувати на занятті від 60 до 70 ключових питань курсу і спостерігати аналіз приблизно 180–210 відповідей, виконаних його товаришами. Колективний аналіз ключових питань і результатів розбивки задач на підзадачі дозволив врахувати не тільки свої помилки, але і помилки інших студентів групи, що безсумнівно відбивалося на якості аналізу визначень і рішенні задач надалі.

Рівень якості того чи іншого знання й уміння оцінювався викладачем по дванадцятибальною шкалою. У результаті спостереження й аналізу більш 840 занять була встановлена залежність між рівнем сформованості знань та умінь і кількістю проаналізованих базових питань (прямо пропорційних кількості занять). При побудові структури предметних понять головною була інформаційно-пошукова діяльність студентів, компонентом і результатом якої є інформаційно-пошукові уміння. В ході експерименту оцінювалися наступні знання й уміння:

- знання базових понять і визначень по інформатиці;
- знання особистого матеріалу по інформатиці;
- уміння встановлювати необхідні й достатні ознаки понять;
- уміння аналізувати поняття й відносини між ними;
- уміння виробляти план виконання перетворення об'єкта (орієнтування на виконавчу частину способу дії);
- уміння перетворювати об'єкт (виконавча частина способу дії);
- уміння співвідносити продукт дії з поставленою перед студентами задачею (контрольна частина способу дії).

Експеримент показав, що для формування різних умінь потрібно різний час, причому це розходження в часі може бути дуже значним. Поряд з уміннями, що формуються досить швидко, є уміння, якими студенти в достатньому ступені, так і не опанували (наприклад, уміння перетворювати об'єкт). Найбільше складно формувати контрольну частину способу дії.

Показовими є результати порівняльного аналізу динаміки формування знання базових понять, визначень, а також уміння встановлювати необхідні й достатні ознаки понять, аналізувати поняття й відносини між ними. Показники для рівня сформованості знання виявилися краще, ніж показники, що описують рівень сформованості умінь аналізувати ці знання.

Характерно, що залежність від часу рівня сформованості умінь має яскраво виражений шпильчастий характер, оскільки не можна розуміти те, що не знаєш, а вище названі уміння саме і свідчать про розуміння студентами базових понять і визначень. Добре відомо, що «знати» й «розуміти» – не синоніми. Не можна розуміти, не знаючи; але досить часто ми маємо знання без розуміння, іншими словами, у тих випадках, коли ми розуміємо, ми повинні обов'язково знати, але немає необхідності у зворотному [5; С. 35], по відтворенню тексту не можна з повною вірогідністю судити про якість його розуміння, що й підтверджується в експерименті.

Результати наступного експерименту (парного порівняння) представлені в табл. 1. Ця таблиця являє собою матрицю парного порівняння студентів-економістів за рівнем сформованості в них основних інформаційних знань і умінь, отриманих ними в

2001/2002 навчальному році. Зокрема, за основу узяті знання й уміння по роботі в операційній системі Windows.

Методика парного порівняння полягає в тім, що за основу порівняння береться одна характеристика студента по обраному питанню. Якщо ці рівні однакові для двох студентів, то кожний з них одержує по 1 балу. Якщо один із них має перевагу, то він одержує 2 бали. При цьому студент, що уступає за рівнем прояву уміння, одержує нуль. Проаранжував прізвища, що коштують у списку, по ступені сформованості якості одержують таблицю, у якій студенти, що одержали однакову кількість балів, мають різні ранги (абсолютне рангове місце  $R_A$ ).

Щоб зрівняти їх по рангах, розраховують відносне рангове місце ( $R_O$ ), а потім процентний ранговий показник ( $R_P$ ) (див. формулу (1))

Для цього відповідні ранги складаються, і загальна сума поділяється на число рангів. Таким чином, оброблений матеріал дозволяє розрахувати процентний ранговий показник ( $R_P$ ), що дозволяє співвіднести результати по одному показнику з результатами за іншими показниками, що отримані в такий же спосіб, чи проаналізувати їх у залежності від інших показників. Розрахунок процентного рангового показника проводиться по формулі:

$$R_P = [(2 R_O - 1) / 2 N] * 100, \quad (1)$$

де  $R_O$  - відносне рангове місце даного випробуваних;

$N$  - кількість членів даного колективу.

Це дозволяє співвіднести результати по одному показнику з результатами за іншими показниками, що отримані в такий же спосіб, чи проаналізувати їх у залежності від інших показників.

У табл. 2, побудованої на підставі табл. 1, представлені рангові показники студентів, знання й уміння яких порівнювалися. Прізвища студентів експериментальної групи відзначені зірочкою. Видно, що ці студенти знаходяться наприкінці представленої таблиці, тобто їхні рангові показники значно вище, ніж у студентів контрольної групи. Це говорить про те, що застосовуючи в навчанні діяльнісний підхід, побудований на основі предметної моделі студента, ефективність формування компонентів інформаційної культури на багато зростає.

Таблиця 1

Матриця парного порівняння студентів за рівнем сформованості у них основних інформаційних знань і умінь (2001/2002 навчальний рік)

Студенти, які порівнюються	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	Заг. бал	
1. Андрущенко Е.*	X	2	2	2	0	2	2	2	0	0	2	2	2	2	2	2	2	2	0	2	2	2	2	2	2	38
2. Бойченко В.	0	X	0	0	1	0	0	0	1	0	0	0	0	2	1	0	0	0	1	1	0	0	1	0	0	8
3. Верба Е.*	0	2	X	1	0	2	1	1	2	0	1	1	2	2	0	0	1	1	2	2	0	0	1	2	2	24
4. Воронцов Н*	2	2	1	X	2	2	2	2	1	2	2	2	1	2	2	2	2	2	2	2	2	1	2	2	2	42
5. Григор'єв Я.	2	1	2	0	X	1	1	0	0	0	0	0	2	0	0	0	0	0	0	2	0	0	0	0	0	11
6. Долженко О.	0	2	0	0	1	X	1	0	0	1	1	1	1	2	1	1	1	1	1	2	1	1	1	1	1	21
7. Дроботова Ю.	0	2	1	1	1	1	X	0	0	1	1	1	1	1	0	1	1	1	0	2	1	0	1	1	1	19
8. Єрмілова А.*	0	2	1	0	2	2	2	X	2	2	1	2	2	2	1	1	2	2	2	2	2	2	2	1	2	37
9. Юхимович А.*	2	1	0	1	2	2	2	0	X	1	1	1	1	2	1	1	1	1	1	2	1	1	1	1	1	27
10. Іволга О.*	2	2	2	2	2	1	1	0	1	X	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	41
11. Капкова М.*	0	2	1	2	2	1	1	1	0	X	1	2	2	1	2	1	2	1	2	1	2	2	1	2	1	31
12. Коваленко Ю.	0	2	1	2	2	1	1	0	1	0	X	1	1	1	1	1	1	1	1	1	1	1	1	1	1	23
13. Конвісаров А.	0	2	0	1	2	1	1	0	1	0	0	X	2	1	0	1	0	1	1	2	0	1	0	1	0	18
14. Крилова І.	0	0	0	0	0	0	1	0	0	0	0	1	0	X	0	0	1	0	0	0	0	0	0	1	0	4
15. Паршин Д.*	0	1	2	2	2	1	2	1	1	0	1	1	1	2	X	1	1	1	1	2	1	1	1	1	1	27
16. Перегудова Т.*	0	2	2	2	2	1	1	1	1	0	0	1	2	2	1	X	2	2	2	2	2	2	2	2	2	34
17. Піхтіна І.	0	2	1	2	2	1	1	0	1	0	1	1	1	1	1	0	X	1	1	2	1	1	1	1	1	23
18. Пшенична Н.*	0	2	1	2	2	1	1	0	1	0	0	1	2	2	1	0	1	X	2	1	1	2	1	2	2	26
19. Сакун Д.*	2	1	0	2	2	1	2	0	1	0	1	1	1	2	1	0	1	0	X	2	2	2	2	2	2	28
20. Сгибнева Ю.	0	1	0	0	0	0	0	0	0	0	0	1	0	2	0	0	0	1	0	X	0	0	0	0	0	5
21. Скічко А.	0	2	2	1	2	1	1	0	1	0	0	1	2	2	1	0	1	1	0	2	X	1	1	1	1	23
22. Слюсаренко Т.*	0	2	2	2	2	1	2	0	1	0	1	1	1	2	1	0	1	0	0	2	1	X	2	2	2	26
23. Смеян Е.	0	1	1	2	2	1	1	1	1	0	0	1	2	1	1	0	1	1	0	2	1	0	X	0	0	20
24. Тертичний В.	0	2	0	2	2	1	1	0	1	0	1	1	1	2	1	0	1	0	0	2	1	0	2	X	2	21
Студенти, які порівнюються	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	Заг. бал	

Таблиця 2

Рангові показники парного порівняння знань і умінь по інформації студентів у 2001/2002 навчальному році

Прізвища студентів	Загальний бал	R <sub>A</sub>	R <sub>O</sub>	R <sub>P</sub>
Крилова І.	4	1		
Сгибнева Ю.	5	2	2	6
Бойченко В.	8	3		
Григор'єв Я.	11	4		

Прізвища студентів	Загальний бал	R <sub>A</sub>	R <sub>O</sub>	R <sub>P</sub>
Конвісаров А.	18	5	5,5	1
Дроботова Ю.	19	6		
Смеян Е.	20	7		
Долженко О.	21	8		
Тертичний В.	21	9		
Коваленко Ю.	23	10	10	0
Піхтіна І.	23	11		
Скичко А.	23	12		
Верба Е.*	24	13		
Пшенична Н.*	26	14		
Слюсаренко Т.*	26	15	15,5	3
Юхимович А.*	27	16		
Паршин Д.*	27	17		
Сакун Д.*	28	18		
Капкова М.*	31	19		
Перегудова Т.*	34	20	20	1
Єрмілова А.*	37	21		
Андрущенко Е.*	38	22	22	0
Іволга О.*	41	23	23	4
Воронцов Н*	42	24	24	8

Це підтверджують підсумки задачі іспитів із курсів «Інформатика та комп'ютерна техніка» і «Лабораторний практикум із комп'ютерних технологій». Результати наведені в табл. 3. Абсолютна успішність і в експериментальних і в контрольних групах збігаються, зате, якість знань за курсом в експериментальних групах значно вище, ніж у контрольних.

Таблиця 3

Результати задачі студентами іспитів з інформатики (2001/2002 навчальний рік)

Навчальна дисципліна	Експериментальна група		Контрольна група	
	Середній бал	Якість знань	Середній бал	Якість знань
Інформатика та КТ	4,2	21%	4,2	24%
Лаб. практикум з комп. технологій	4,3	31%	4,3	24%

Діяльнісний підхід навчання інформатиці на основі предметної моделі студента, був апробований у двадцяти чотирьох групах студентів при вивченні курсів «Інформатика та комп'ютерна техніка» і «Лабораторний практикум з комп'ютерних технологій». При цьому спостерігалось достовірне поліпшення екзаменаційних результатів. Це було визначено за допомогою методу  $\chi^2$  («хі-квадрат»).

Метод  $\chi^2$ , був обраний не випадково. При малій вибірці, коли розподіл не є нормальним (такими, є наші виміри), поряд із використанням методу парного порівняння ефективне використання непараметричного методу  $\chi^2$  [6; 7; 8].

При використанні методу  $\chi^2$  досить порівняти число випробуваних в експериментальній і контрольній групі, у яких змінився досліджуваний показник. Метод  $\chi^2$  дозволяє установити, залежить чи ні зміна від застосування особливих методів (у нашому випадку, за допомогою діяльнісного підходу навчання інформатиці на основі предметної моделі студента) в експериментальній групі, чи носить випадковий характер.

За результатами вимірювання визначають емпіричні частоти (Э) для студентів контрольних і експериментальних груп, тобто число студентів, що поліпшили свої показники і ті, у яких показники або погіршилися, або залишилися без змін. Далі обчислюють теоретичні частоти (Т), що були б отримані, якби всі розходження були чисто випадковими.

За результатами вимірювання визначають емпіричні частоти (Э) для студентів контрольних і експериментальних груп, тобто число студентів, що поліпшили свої показники (Э<sub>лэ</sub> – число студентів експериментальної групи, що поліпшили свої показники; Э<sub>лк</sub> – число студентів контрольної групи, що поліпшили свої показники) і ті, у яких показники або погіршилися, або залишилися без змін (Э<sub>хэ</sub> – число студентів експериментальної групи, що погіршили чи не змінили свої показники; Э<sub>хк</sub> – число студентів контрольної групи, що погіршили чи не змінили свої показники). Далі обчислюють теоретичні частоти (Т), що були б отримані, якби всі розходження були чисто випадковими. Вони розраховуються по формулах (див. формулу (8) [10]:

$$T_{лэ} = \frac{(\text{Э}_{лэ} + \text{Э}_{лк}) * n_{э}}{N}, \quad T_{лк} = \frac{(\text{Э}_{лэ} + \text{Э}_{лк}) * n_{к}}{N} \quad (2)$$

$$T_{x_3} = \frac{(\Theta_{x_3} + \Theta_{x_k}) * n_3}{N}, \quad T_{x_k} = \frac{(\Theta_{x_3} + \Theta_{x_k}) * n_k}{N}$$

де  $n_3$  – число студентів експериментальної групи;

$n_k$  – число студентів контрольної групи;

$N = n_3 + n_k$  – загальне число студентів.

При  $n_3 = n_k$ ;  $T_{л_3} = T_{л_k}$  і  $T_{x_3} = T_{x_k}$ .

Метод  $\chi^2$  полягає в тому, що оцінюють, наскільки подібні між собою розподіли емпіричних і теоретичних частот. Якщо різниця між ними невелика, то можна думати, що відхилення емпіричних частот від теоретичних обумовлені випадковістю. Якщо ж, навпроти, ця різниця велика, то можна вважати, що розходження між ними значимі й існує визначений зв'язок між застосуванням особливих методів в експериментальній групі і розподілом емпіричних частот.

Для обчислення  $\chi^2$  використовують формулу:

$$\chi^2 = \frac{(\Theta_{л_3} - T_{л_3})^2}{T_{л_3}} + \frac{(\Theta_{x_3} - T_{x_3})^2}{T_{x_3}} + \frac{(\Theta_{л_k} - T_{л_k})^2}{T_{л_k}} + \frac{(\Theta_{x_k} - T_{x_k})^2}{T_{x_k}} \quad (3)$$

У табл. 4 як приклад надані результати задачі екзаменів із курсу «Інформатика та комп'ютерна техніка» (четвертий семестр) і «Лабораторний практикум із комп'ютерних технологій» (п'ятий семестр) студентами експериментальних і контрольних груп лише денної форми навчання. Видно, що в 17 із цих студентів успішність підвищилася, причому 12 із них навчалися інформатиці із застосуванням діяльнісного підходу на основі предметної моделі студента. Значення емпіричних частот представлені в табл. 5. Далі порівняємо ці дані з теоретичними частотами (розраховані по формулах (2), що були б отримані, якби всі розходження були чисто випадковими. Вони представлені в табл. 6.

Для розрахунку числа ступенів свободи число рядків у табл. 5 (експериментальна й контрольна групи) за винятком одиниці множать на число стовпців (краще; без зміни, чи гірше) за винятком одиниці. Таким чином, у нашому випадку число ступенів свободи дорівнює  $(2 - 1) * (2 - 1) = 1$ .



Таблиця 4

Результати здачі іспитів з інформатики студентами-економістами в 2001/2002 навчальному році (значком \* відзначені прізвища студентів експериментальної групи)

Прізвища, імена студентів	Учбові курси	
	Інформатика та комп'ютерна техніка	Лабораторний практикум із комп'ютерних технологій
1. Бердник С. *	3	4
2. Бистра Н.	3	4
3. Варич Ю. *	4	5
4. Власенко Т. *	4	5
5. Гігієнова М.	5	5
6. Епельман Ю. *	4	5
7. Дорофєєва Е. *	3	4
8. Загребін В. *	4	5
9. Іноземцев А.	3	3
10. Ісакова М. *	3	4
11. Калинчева А. *	3	4
12. Калієвська Е.	3	3
13. Кібальнічко А.	4	4
14. Ковровська М.*	5	5
15. Козулько А.	3	3
16. Копас Д. *	3	4
17. Кузнєцов Е.	3	5
18. Кульбака І.	3	4
19. Мішукова В. *	5	4
20. Оборотова Н.	5	5
21. Панов А. *	3	3
22. Півовар Т.	3	3
23. Синенко Д.	4	5
24. Ткачова Я. *	4	5
25. Токаренко А.	3	3
26. Толстов С. *	4	5
27. Трофіменко О.	3	4
28. Фесенко О.	4	4
29. Шахрова Ж. *	4	5
30. Яриш А.	5	5

Таблиця 5

## Емпіричні частоти

Умови	Результати		
	Краще	Без змін, чи ще гірше	Усього:
Експериментальна група	12	3	15
Контрольна група	5	10	15
Усього:	17	13	30

Таблиця 6

## Теоретичні частоти

Умови	Результати		
	Краще	Без змін, чи ще гірше	Усього:
Експериментальна група	8,5	6,5	15
Контрольна група	8,5	6,5	15
Усього:	17	13	30

Табличне значення  $\chi^2$  для рівня значимості 0,05 і одного ступеня свободи складає 3,841, а для рівня значимості 0,01 і одного ступеня свободи складає 6,635 [6; 7; 8]. На підставі табл. 5 і 6 по формулі (3) визначимо  $\chi^2$ , що відповідає експериментальним даним. Обчислене нами значення  $\chi^2 = 6,652 > 6,635 > 3,841$ . Таким чином, наше значення  $\chi^2$  перевищує границі критичних значень, що мовою математики позначає, що розбіжності між розподілами статистично достовірні, отже, застосування діяльнісного підходу в навчанні інформатиці на основі предметного моделювання сприяє підвищенню рівня інформаційної культури студентів.

Після виявлення розходження в розподілі ознаки (метод  $\chi^2$ , по іншому називають критерій Пірсона, чи критерій згоди розподілів), визначимо, яка частка спостережень у даній вибірці (експериментальній і контрольній групах) характеризується цікавлячим нас ефектом, і яка частка цим ефектом не характеризується. Для цього скористаємося багатофункціональним статистичним методом – критерієм  $\phi^*$  Фішера (кутове перетворення Фішера) [6; С. 158].

Суть методу критерію Фішера складається у визначенні того, яка частка спостережень у даній вибірці характеризується цікавлячим нас ефектом і якою часткою цим ефектом не харак-

теризується.

Перетворимо таблицю емпіричних частот (табл. 5) у так названу чотирипільну таблицю (табл. 7), що являє собою таблицю емпіричних частот по двох значеннях ознаки «є ефект» - «немає ефекту».

Таблиця 7

Групи	«Є ефект»		«Немає ефекту»			Сума:
	Кількість випробуваних	% доля	Кількість випробуваних	% доля		
1. Експериментальна група	12	А (80 %)	3	Б (20%)		15
2. Контрольна група	5	В (33,3%)	10	Г (66,7%)		15
Сума:	17		13			30

Беруть участь у зіставленні тільки поля (чарунки) А і В, тобто процентні частки по стовпцю «є ефект».

Табличне значення величини  $\varphi$ , що відповідають процентним часткам у кожній із груп рівні:

$$\varphi_1(80\%) = 2,214$$

$$\varphi_2(33,3\%) = 1,320$$

$$\varphi^* = (\varphi_1 - \varphi_2) \cdot \sqrt{\frac{n_1 \cdot n_2}{n_1 + n_2}} \quad (4)$$

де  $\varphi_1$  – кут, що відповідає більшій % частці;

$\varphi_2$  – кут, що відповідає меншій частці % частці;

$n_1$  – кількість спостережень у вибірці 1;

$n_2$  – кількість спостережень у вибірці 2.

На підставі табл. 7 по формулі (4) вважаємо емпіричне значення  $\varphi^*$ :

$$\varphi^*_{\text{емп}} = 2,70$$

яке відповідає рівню значимості  $\rho = 0,09$ .

Відповідно до встановлених критичних значень  $\varphi^*$ , прийнятими у психолого-педагогічній практиці як рівні статистичної значимості приймаються:

$$\varphi^*_{кр} = \begin{cases} 1,64(\rho \leq 0,05) \\ 2,31(\rho \leq 0,01) \end{cases}$$

Зіставимо ці граничні параметри з нашим емпіричним значенням  $\varphi^*_{\text{емп}} = 2,70$ :

$$\varphi_{\text{эмп}} = 2,70 > \varphi_{\text{кр}} = 2,31$$

Знайшовши цю точку на графіку (див. [6; С. 163]), можна зробити висновок, що ми попадаємо в «зону значимості», отже, метод кутового перетворення Фішера також дає позитивний результат, і вказує на те, що частка студентів, що навчалися на основі діяльнісного підходу на основі предметного моделювання, характеризується цікавлячим нас ефектом. Необхідно помітити, що, згідно авторитетного дослідження Є.В. Сідоренко [6; С. 163], при зіставленні двох вибірок, по якій-небудь якійсь ознаці критерій  $\varphi^*$  більше засмучує дослідників, чим радує. Ми ж одержали позитивний результат, що ще раз підтверджує правильність нашої гіпотези.

Педагогічний експеримент ефективності діяльнісного підходу навчання студентів інформатиці на основі предметної моделі студента, є нетрадиційним експериментом. Його основна мета складалася не у прямому визначенні результативності навчання інформатики (хоча мимоволі, ця задача була теж виконана), а у відстеженні динаміки росту основних умінь і навичок по предмету, і в тім, як ці уміння й навички сприяли більш ефективному викладанню інших спеціальних економічних дисциплін. Наша задача складалася у визначенні того впливу, що робила інформатика, завдяки формуванню нових інформаційних умінь і навичок, виявлених на основі предметної моделі студентів, тих причинно-наслідкових зв'язків, що виникали у процесі формування цих навичок, що тотожно формуванню загальної інформаційної культури майбутнього економіста.

### **3. Висновки**

На основі предметного моделювання студента, розроблена нова організація навчального процесу по інформаційним курсам для економічних спеціальностей вузів, що відкриває наступні можливості для проектування ефективного навчального процесу.

Вона дозволяє:

- більш детально визначати зміст навчання;
- підсилити практичну сторону навчального процесу;
- забезпечує реалізацію діяльнісного підходу в навчанні;
- підвищує ефективність контролю навчальної діяльності;
- розробляти нові види навчальної діяльності.

За допомогою педагогічного експерименту підтверджено,

що вивчення інформатики сприяє:

- формуванню знань, умінь і навичок, необхідних для вивчення спецкурсів по економічній інформатиці;
- прискоренню формування всіх частин способу дії;
- виробленню нових якостей, що дозволяють виконувати найважливіші професійні функції економіста, тобто підвищує загальну інформаційну культуру майбутніх економістів.

Використовуючи методи математичної обробки педагогічних даних, експериментально підтверджені ефективність і результативність запропонованого предметного моделювання тих, яких навчають, на основі діяльнісного підходу по формуванню в них інформаційної культури.

### Література

1. Атанов Г.А. Деятельностный подход в обучении. – Донецк: ЕАИ-пресс, 2001. – 160 с.
2. Атанов Г.А., Пустынникова И.Н. Обучение и искусственный интеллект, или основы современной дидактики высшей школы. / Под ред. Г.А. Атанова. – Донецк: Изд-во ДООУ, 2002. – 504 с.
3. Коляда М.Г. Программа курса информатики для экономических специальностей вузов // ПостМетодика. – Полтава, 2002. – №4 (42). – С. 90–94.
4. Коляда М.Г. Формирование умений по информатике студентов экономических специальностей вузов // Наук.-пр. друга Міжнар. конф. «Ідеї Вернадського та взаємообумовленість стратегічного партнерства вищого навчального закладу і школи в освітньому просторі регіону» (18-20 квітня 2002 р.). – Полтава, ПостМетодика. – 2002. – №2-3 (40-41). – С. 88-92.
5. Нишанов В.К. Феномен понимания: когнитивный анализ. – Фрунзе: Илим, 1990. – 228 с.
6. Сидоренко Е.В. Методы математической обработки в психологии. – СПб.: ООО «Речь», 2001. – 350 с.
7. Смирнов Н.В., Дунин-Барковский И.В. Курс теории вероятностей и математической статистики для технических приложений. – М.: Наука, 1969. – 512 с.
8. Суходальский Г.В. Основы математической статистики для психологов. – СПб.: Изд-во СПб. ун-та, 1998. – 428 с.

## **ВИКОРИСТАННЯ ЗАДАЧ ДОСЛІДНИЦЬКОГО ХАРАКТЕРУ В ПРОЦЕСІ НАВЧАННЯ ІНФОРМАТИКИ У ВИЩИХ ЗАКЛАДАХ ОСВІТИ ЕКОНОМІЧНОГО ПРОФІЛЮ**

Ю.М. Красюк

м. Київ, Київський національний економічний університет

Важливим показником рівня підготовки випускника кожного навчального закладу виступає його вміння перспективно використовувати досить великий обсяг фактологічної інформації, яка була ним засвоєна в процесі навчання, не тільки до відомих йому проблем, але й у нестандартних, непередбачуваних життєвих та виробничих ситуаціях. Розвиток цих умінь зумовлює дидактичну необхідність організації на всіх етапах навчання самостійної пізнавальної діяльності студентів дослідницького характеру. Це перш за все вимагає розробки відповідних комплектів навчальних задач, які повинні відзначатися такими загальними характеристиками:

- процес розв'язування задачі повинен сприяти підтримці інтересу до предмету та мотивації навчально-пізнавальної діяльності студентів;
- завдання повинні мати практичну значимість та носити фаховий характер;
- процес розв'язування задачі повинен вимагати залучення отриманих знань з інших навчальних дисциплін;
- завдання повинні носити творчий характер, що надавало б можливість розкрити особистісний потенціал кожного студента.

Розглядаючи дану проблему на прикладі курсу інформатики у вищих закладах освіти економічного профілю, необхідно відмітити, що використання нових інформаційних технологій навчання (НІТН) дає можливість значно розширити коло задач дослідницького характеру, які можуть бути використані в процесі навчання. Адже залучення НІТН дає можливість зосередити головну увагу студентів на вирішенні поставленої проблеми, розробці математичної моделі, переклавши рутинні технічні операції на комп'ютер.

При цьому проблема включення задач дослідницького характеру в процес навчання повинна розв'язуватися комплексно, з дотриманням загальних принципів дидактики. Задачі обов'язково повинні відповідати темі заняття або серії занять. Розв'язувати їх необхідно як під час подання нового матеріалу, в ході виконання практичних та лабораторних робіт, так і в процесі самостійної роботи студентів у позаурочний час. При цьому викладач, згідно даних психолого-педагогічного моніторингу та результатів навчально-пізнавальної діяльності студентів, повинен надавати кожному з них необхідний рівень допомоги. Адже, як відзначав Ю.К. Бабанський в [1], якщо диференціюється не тільки і не стільки зміна обсягу і складності навчального матеріалу, скільки допомога студенту, то такий підхід дозволить йому досягнути максимуму власних можливостей в даний момент.

Наведемо приклад однієї з навчальних задач дослідницького характеру, яку доцільно використовувати під час вивчення теми "Системи опрацювання таблично поданих даних".

Задача. У файлі *завдання\_1.xls* (частина електронної таблиці представлена на рис. 1) наведені показники реалізації підприємством великої кількості партій товару (147 партій) семи різних видів протягом минулого року. Необхідно підрахувати виручку та отриманий прибуток від кожної відправленої партії продукції. Побудувати діаграму поквартального отримання прибутку за кожним видом продукції. З'ясувати ступінь нерівномірності отримання прибутку підприємством від кожного виду продукції протягом чотирьох кварталів минулого року.

	A	B	C	D	E	F	G	H
1	№	Номенклатурний номер продукції	Собівартість одиниці продукції, грн	Дата реалізації	Кількість реалізованої продукції, тис. шт	Відпускна ціна одиниці продукції, грн	Виручка від реалізації, грн	Прибуток, грн
2	1	23524	20,05 грн.	09.01.01	2,5	22,06 грн.		
3	2	26059	35,62 грн.	10.01.01	3,6	39,18 грн.		
4	3	30782	56,75 грн.	15.01.01	1,2	62,43 грн.		
5	4	41210	40,11 грн.	18.01.01	4,7	44,12 грн.		
6	5	50724	86,24 грн.	22.01.01	5,8	94,86 грн.		
7	6	50751	69,01 грн.	22.01.01	6,5	75,91 грн.		
8	7	55073	60,12 грн.	23.01.01	3,002	66,13 грн.		
9	8	23524	20,05 грн.	25.01.01	8,5	22,06 грн.		
10	9	26059	35,62 грн.	29.01.01	9,4	39,18 грн.		

Рис. 1.

*Розв'язування:* Розглянемо основні етапи розв'язування поставленої задачі.

Спочатку слід розв'язати досить просте завдання - підрахувати отриману виручку та прибуток від реалізації продукції підприємства. Записавши відповідно до комірок G2, H2 формули  $=E2*F2$  та  $=G2-C2*E2$  (рис. 1), а також скопіювавши їх для наступних рядків даних, отримуємо необхідні результати.

Далі для побудови діаграми поквартального отримання прибутку за кожним видом продукції необхідно визначити номер кварталу в якому була реалізована кожна партія продукції. З цією метою доцільно ввести до таблиці між стовпчиками "Дата реалізації" та "Кількість реалізованої продукції" ще один стовпчик "Квартал" та за допомогою формули  $=ЯКЩО(МІСЯЦЬ(D2)>=1; МІСЯЦЬ(D2)<=3); 1; ЯКЩО(МІСЯЦЬ(D2)<=6; 2; ЯКЩО(МІСЯЦЬ(D2)<=9; 3; 4))$  обчислити номер кварталу (рис. 2).

	A	B	C	D	E	F	G	H	I
	№	Номенклатурний номер продукції	Собівартість одиниці продукції, грн	Дата реалізації	Квартал	Кількість реалізованої продукції, тис. шт	Відпускна ціна одиниці продукції, грн	Виручка від реалізації, грн	Прибуток, грн
2	1	23524	20,05 грн.	09.01.01	1	2,5	22,06 грн.	55,14 грн.	5,01 грн.
3	2	26059	35,62 грн.	10.01.01	1	3,6	39,18 грн.	141,06 грн.	12,82 грн.
4	3	30782	56,75 грн.	15.01.01	1	1,2	62,43 грн.	74,91 грн.	6,81 грн.
5	4	41210	40,11 грн.	18.01.01	1	4,7	44,12 грн.	207,37 грн.	18,85 грн.
6	5	50724	86,24 грн.	22.01.01	1	5,8	94,86 грн.	550,21 грн.	50,02 грн.
7	6	50751	69,01 грн.	22.01.01	1	6,5	75,91 грн.	493,42 грн.	44,86 грн.
8	7	55073	60,12 грн.	23.01.01	1	3,002	66,13 грн.	198,53 грн.	18,05 грн.
9	8	23524	20,05 грн.	25.01.01	1	8,5	22,06 грн.	187,47 грн.	17,04 грн.
10	9	26059	35,62 грн.	29.01.01	1	9,4	39,18 грн.	368,31 грн.	33,48 грн.
11	10	30782	56,75 грн.	30.01.01	1	7,5	62,43 грн.	468,19 грн.	42,56 грн.

Рис. 2.

Зважаючи на те, що в списку електронної таблиці знаходиться 147 записів, у яких представлена інформація про реалізацію семи різних видів продукції протягом чотирьох кварталів минулого року, а необхідно побудувати діаграму поквартального отримання прибутку за кожним видом продукції, доцільно спочатку скласти відповідний зведений звіт. Для виклику *Майстра зведених таблиць або діаграм* задаємо вказівки Дані / Зведена таблиця. Далі на першому кроці побудови виконуємо: включення режиму *Створити таблицю на основі даних, які знаходяться в списку або базі даних MS Excel*; вибір такого виду звіту як *Зведена діаграма (зі зведеною таблицею)*. На другому кроці вказуємо діапазон комірок, які містять вихідні дані (для нашого випадку це  $\$A\$1:\$I\$148$ ). І на останньому кроці роботи *Майстра діаг-*



рам вибираємо режим розміщення таблиці на новому листі. Після цього до робочої книги додається два робочих листа (лист для зведеної діаграми та для самої зведеної таблиці). В довільному з них визначаємо *Поле категорій* як стовпчик *Номенклатурний номер продукції*, *Поле рядів* як *Квартал*, а до *Елементів даних* відносимо стовпчик *Прибуток*. Отримаємо відповідну зведену діаграму та зведену таблицю (рис. 3, рис. 4).

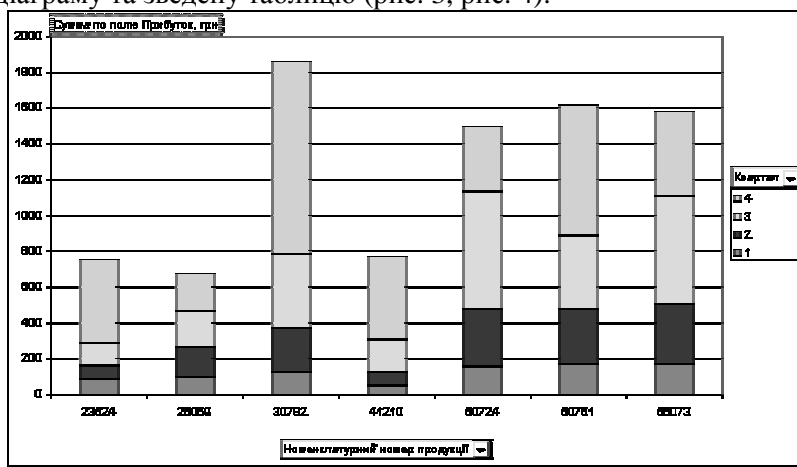


Рис. 3.

	A	B	C	D	E	F
1	Сумма по полю Прибуток, грн	Квартал				
2	Номенклатурний номер продукції	1	2	3	4	Общий итог
3	23524	80,288	80,32	125,194	467,079921	752,881921
4	26059	97,26262	164,2968	200,72325	211,089195	673,371865
5	30782	122,06925	243,552	418,45686	1080,1824	1864,26051
6	41210	46,96881	72,55815	180,3358125	470,923425	770,7861975
7	50724	150,112295	324,8426	656,30225	361,16874	1492,425885
8	50751	167,093535	308,1976	409,21947	729,839355	1614,34996
9	55073	164,247072	334,0272	607,26794	474,467304	1580,009516
10	Общий итог	828,041582	1527,79435	2597,499583	3794,75034	8748,066855

Рис. 4.

Перед тим як перейти до визначення степеня нерівномірності отримання прибутку підприємством від кожного виду продукції протягом чотирьох кварталів минулого року, необхідно побудувати криву Лоренца – залежність процентів прибутку від проценту відпрацьованих робочих днів (крива *OAB* (рис. 5)). У даному випадку на осі абсцис відкладається відсоткове значення кількості відпрацьованих днів (100%, кількість відпрацьованих

днів за рік, приймається за 1), а на осі ординат – відсоткове значення кількості отриманого прибутку на поточний момент (загальна кількість отриманого прибутку за рік приймається за 1). Графік кривої Лоренца досить наглядно демонструє, що якщо прибуток отримувався рівномірно, то крива Лоренца виродиться в пряму  $Y=X$ . Тому площа фігури  $OBA$  між бісектрисою  $OB$  та кривою Лоренца віднесена до площі трикутника  $OBC$  характеризує степінь нерівномірності надходження прибутку протягом чотирьох кварталів, який ще називають коефіцієнтом Джині. Таким чином, чим більший коефіцієнт Джині, тим більш нерівномірно відбувалося надходження прибутку.

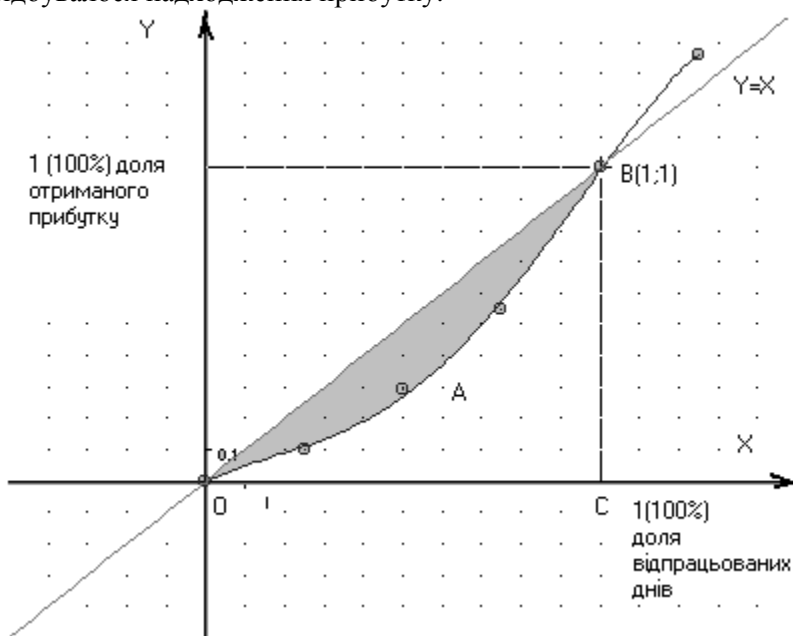


Рис. 5.

Отже, скористаємося числовими даними зведеної таблиці про щоквартальне надходження прибутку від кожного виду продукції та перерахуємо їх як відсоткову ставку до річного прибутку від реалізації даного виду товару. В свою чергу перший квартал буде складати 0,25 % від всього року, два квартали – 0,5%, три квартали – 0,75%, а весь рік відповідно 1 (рис. 6).

	A	B	C	D	E
12	Співвідношення відсотку прибутку до відсотку відпрацьованих днів				
13	Номенклатурний номер продукції	0,25	0,5	0,75	1
14	23524	0,11	0,21	0,38	1,00
15	26059	0,14	0,39	0,69	1,00
16	30782	0,07	0,20	0,42	1,00
17	41210	0,06	0,16	0,39	1,00
18	50724	0,10	0,32	0,76	1,00
19	50751	0,10	0,29	0,55	1,00
20	55073	0,10	0,32	0,70	1,00

Рис. 6.

Далі для побудови кривої Лоренца та визначення коефіцієнта Джині, зважаючи на те, що в MS Excel немає необхідних засобів для обчислення площ геометричних фігур, доцільно скористатися можливостями ППЗ GRAN 1 [2], який був розроблений на кафедрі інформатики Національного педагогічного університету імені М.П. Драгоманова. Дана програма розроблена для середовища Windows, має стандартний зручний інтерфейс та не вимагає потужних технічних ресурсів комп'ютера.

У ППЗ GRAN 1 функцію можна задати явно ( $Y=Y(X)$ ), в параметричній формі ( $Y=Y(t)$ ,  $X=X(t)$ ), через полярні координати ( $R=R(F)$ ), неявно ( $0=G(X, Y)$ ), таблично ( $X_i, Y(X_i)$ ). Оскільки в задачі, яка розглядається, функція Лоренца визначена через п'ять точок координатної площини, то задамо її таблично. Для цього у вікні *Список об'єктів* виберемо *Таблична* ( $X_i, Y(X_i)$ ), виконаємо вказівки *Об'єкт / Створити*. Далі занесемо відповідні числові дані стосовно товару з номенклатурним номером 23524 та вкажемо степінь полінома – 4 (рис. 7).

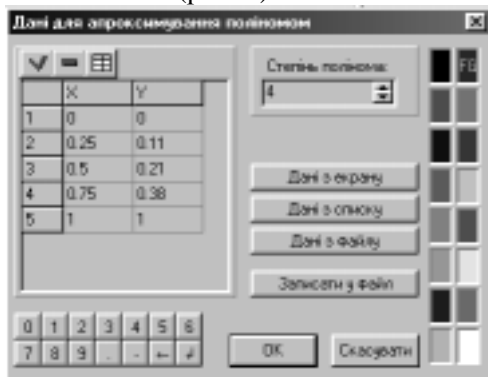


Рис. 7.

Для задання прямої  $Y=X$  у вікні *Список об'єктів* виберемо *Явна* ( $Y=Y(X)$ ), виконаємо вказівки *Об'єкт / Створити* та введемо відповідну функцію. Таким чином у вікні *Список об'єктів* будуть знаходитися дві вище введені функції. З метою побудови відповідних графіків виконуємо вказівки *Графік / Побудувати* (рис. 8). Встановивши вказівник мишки на точку перетину кривої Лоренца та прямої  $Y=X$ , з'ясуємо, що вона має координати (1; 1). Отже, вершини трикутника, площу якого потрібно визначити, мають координати (0; 0), (1; 0) та (1; 1), а це означає, що його площа дорівнює 0,5 кв. один.

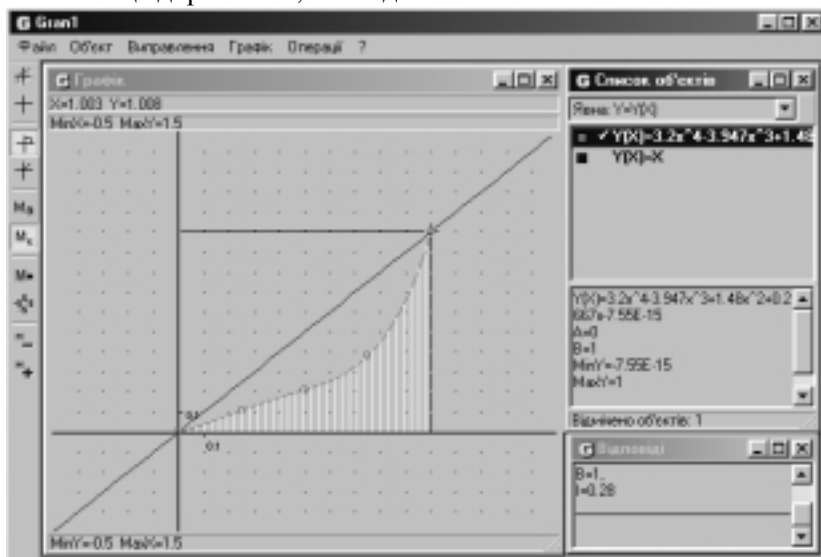


Рис. 8.

Залишається визначити площу фігури, яка обмежена кривою Лоренца та прямою  $Y=X$ . Вона обчислюється як різниця площі вище зазначеного трикутника (яка рівна 0,5 кв. один.) та площі криволінійної трапеції, яка утворена при перетині кривої Лоренца та прямих  $Y=0$ ,  $X=1$  (обчислюється через визначений інтеграл від функції Лоренца на проміжку від 0 до 1). Для обчислення площі даної криволінійної трапеції виконуємо вказівки *Операції / Інтеграл* / *Інтеграл* (у даному випадку виділеною у вікні *Список об'єктів* повинна бути тільки функція Лоренца) та задаємо відповідні межі інтегрування. У вікні *Відповіді* побачимо резуль-

тат обчислень – 0,28 кв. один. Аналогічні обрахунки проводимо для всіх інших видів продукції та заносимо отримані результати до робочого листа *Результати обчислень* файлу *Завдання\_1.xls* (рис. 9). Підрахуємо коефіцієнт Джині для товару з номенклатурним кодом 23524 за формулою  $= (0,5 - B2) / 0,5$  та скопіюємо її для товарів інших видів (рис. 9).

	А	В	С
1	Номенклатурний номер продукції	Площа криволінійної трапеції	Коефіцієнт Джині
2	23524	0,28	0,44
3	26059	0,4249	0,1502
4	30782	0,2787	0,4426
5	41210	0,2591	0,4818
6	50724	0,4262	0,1476
7	50751	0,3476	0,3048
8	55073	0,4049	0,1902

Рис. 9.

Проаналізуємо отримані результати дослідження. Оскільки коефіцієнт Джині найбільший для товару з номенклатурним номером 41210, то рівномірність надходження прибутку від реалізації цього товару найменша. Відповідно найбільш рівномірно прибуток надходив від реалізації товару з номенклатурним кодом 50724. Це означає, що відділу маркетингових досліджень підприємства потрібно провести додаткові дослідження по вивченню попиту на продукцію з такими номенклатурними номерами як 41210, 30782 та 23524.

Як свідчить досвід, систематичне використання в курсі інформатики навчальних задач дослідницького характеру сприяє розвитку прийомів самостійної пізнавальної діяльності студентів, реалізації їх творчого потенціалу. Успіх, який вони відчували в результаті надання навчально-пізнавальної діяльності творчого, дослідницького характеру, давав значний імпульс підвищення пізнавальної активності.

#### Література

1. Бабанский Ю.К., Поташник М.М. Оптимизация педагогического процесса (в вопросах и ответах). – К: Радянська школа, 1982.
2. Жалдак М.І. Комп'ютер на уроках математики: Посібник для вчителів. – К.: Техніка, 1997. – 304 с.: іл.

## ИЗ ОПЫТА ОБУЧЕНИЯ СТУДЕНТОВ ВУЗОВ ПРОГРАММИРОВАНИЮ НА ЯЗЫКЕ ПАСКАЛЬ

И.А. Кривель<sup>1</sup>, А.Н. Моргун<sup>2</sup>

<sup>1</sup> г. Черкассы, Черкасский государственный университет имени  
Богдана Хмельницкого

<sup>2</sup> г. Черкассы, Черкасский институт пожарной безопасности име-  
ни Героев Чернобыля

Основные проблемы с обучением программированию в современной высшей школе связаны с низким «средним стартовым уровнем знаний студентов по информатике», а также с его значительной неравномерностью, что, в частности, отмечено в [1]. Глубинные причины этого явления кроются в снижении уважения к сфере образования на фоне общего падения общественной значимости знаний. Учащиеся массы всё менее склонны относиться к своим учебным заведениям как к «храмам науки». И отмеченные тенденции постепенно становятся характерными не только для школ, но и для вузов. В складывающихся условиях серьёзно возрастают требования к педагогическому мастерству преподавателя, который должен суметь заинтересовать аудиторию, увлечь её, превратить утомительный процесс поглощения и переваривания информации в почти удовольствие. Таким образом, современное учебное заведение всё более нуждается в педагоге. К сожалению, всё это неминуемо приводит к существенному снижению специального уровня преподаваемых дисциплин. И, очевидно, поэтому в школьных учебниках информатики постепенно минимизируются разделы, требующие интеллектуальных усилий. Подтверждением этого может служить переориентация с «программистов» на «пользователей».

Вместе с тем в любой школе, в любом вузе всегда находится определённая группа учащихся, которые получают удовлетворение от самого процесса получения знаний. Занятия, ориентированные на основную массу учащихся, оказываются для них и не интересными, и не нужными. Такие дети нуждаются в помощи уже не педагога, а специалиста, основательно владеющего предметом. Возможно, когда-нибудь в учебных заведениях будут предусмотрены должности как для педагогов, так и для специа-

листов высокой квалификации исключительно для работы с заинтересованными детьми. Существующая же организация учебного процесса на такое разделение не ориентирована. И поэтому в большинстве случаев такие дети оказываются предоставленными сами себе.

Пожалуй, единственным разумным путём разрешения сложившейся ситуации является создание комплекта учебных пособий, специально ориентированных на самообразование. Такие учебные пособия должны обладать особыми качествами с точки зрения подбора и доступности материала, а также последовательности его изложения, поскольку вряд ли во всех без исключения случаях учащийся сможет рассчитывать на помощь специалиста соответствующей квалификации. В них следует избегать как поверхностности, так и «наукообразия» излагаемого материала, а также тонко учитывать интеллектуальные возможности и запросы заинтересованных учащихся.

В отношении школьного курса информатики тематическая направленность таких учебных пособий должна лежать в русле требований, изложенных в [2]. Характерно, что аналогичные требования могут быть предъявлены и к вузовскому обучению, хотя бы потому, что задача «обеспечения равномерного качественного базового уровня подготовки учащихся по информатике» на уровне школ «в настоящее время решается слабо» [1].

Как указывалось в [2], особая роль в изучении информатики принадлежит разделам, прямо или косвенно связанным с математикой, алгоритмизацией и программированием. При этом особенно важно методически правильно ориентировать учащихся, впервые приступающих к изучению указанных разделов. В частности, пятнадцатилетний школьный и вузовский опыт преподавания программирования на языке Паскаль позволил авторам выработать в этом направлении ряд методически целесообразных рекомендаций.

1. Использование алгоритмического языка на этапе разработки алгоритма решения задачи. Такой язык оказывается удобным для предварительного ознакомления с основными понятиями в области программирования, позволяет в доступной форме представить последовательность решения задачи, не требует выполнения жёстких синтаксических правил, способствует выра-

ботке навыков структуризации алгоритма.

2. Отказ от блок-схем как средства алгоритмизации. Их применение целесообразно только для иллюстрации готовых сравнительно небольших по размеру алгоритмов на начальных этапах обучения.

3. Математическо-логический уклон в подборе решаемых задач. Программист, не знающий математики, теряет право считаться представителем этой профессии.

4. Определение достаточно полного перечня базовых алгоритмов решения типичных задач, на примере которых демонстрируются основные приёмы алгоритмизации.

5. Представление алгоритмов в виде процедур и функций алгоритмического языка с возможностью последующего формализованного преобразования их в соответствующие подпрограммы на языке Паскаль.

6. Ориентация на, так называемые, комбинированные задачи, при решении которых должны использоваться базовые алгоритмы.

7. Реализация принципа «от задачи к программе», то есть изучение средств языка Паскаль по мере необходимости в них для решения всё усложняющихся задач.

8. Представление исполняемой части программы на языке Паскаль в виде линейной последовательности трёх групп операторов: ввода исходных данных, обращения к подпрограмме решения задачи, вывода результатов.

9. Чёткое выделение теоретических аспектов алгоритма (на алгоритмическом языке) и практических особенностей его реализации (на языке Паскаль).

10. Выработка навыков анализа алгоритмов (с помощью таблиц исполнения) и показ механизма процесса их усовершенствования с целью повышения эффективности.

Перечисленные методические рекомендации нашли своё практическое отражение в соответствующем учебном пособии [3]. Книга предназначена, в первую очередь, для использования учащимися средних учебных заведений, желающими научиться правильному программированию, а не просто записи операторов языка Паскаль. Книга может быть полезна и тем студентам младших курсов высших учебных заведений, которым получен-



ная в школе подготовка по информатике не позволяет успешно преодолевать вузовские барьеры. В книге подробнейшим образом рассмотрен процесс построения около пятидесяти линейных, разветвляющихся и циклических базовых алгоритмов, а также приведены соответствующие программы на языке Паскаль. При этом большое внимание уделяется решению комбинированных задач, а также особенностям машинной реализации. По мнению авторов, книга может быть одним из примеров того, как следует учить программированию правильно.

#### Литература

1. Глинський Я.М., Анохіна В.Є., Рязська В.А. Про навчальні посібники та наступність вивчення інформатики в школі і вузі // Теорія та методика навчання математики, фізики, інформатики: Збірник наукових праць, т. 3. – Кривий Ріг: Видавничий відділ НацМетАУ, 2002. – С. 69-72.
2. Моргун А.Н. О содержании школьного курса информатики // Теорія та методика навчання математики, фізики, інформатики : Збірник наукових праць, т. 3. - Кривий Ріг: Видавничий відділ НацМетАУ, 2002. – С. 180-183.
3. Моргун А.Н. Решение задач средствами языка Turbo Pascal 7.0. – К.: Юниор, 2002. – 216 с., ил.

## КОНЦЕПЦИЯ КУРСА "ОСНОВЫ ЗАЩИТЫ ИНФОРМАЦИИ"

С.В. Куклина, Н.В. Моисеенко  
г. Кривой Рог, Криворожский государственный педагогический  
университет  
nvg@kpi.dp.ua

*«Каждый, кто думает, что избрал непробиваемую схему шифрования, – или невероятно редкий гений, или просто наивен и неопытен.*

*Каждый программист воображает себя криптографом, что ведет к распространению исключительно плохого криптообеспечения...»*

*Ф. Зиммерман (автор шифра PGP)*

С каждым годом усиливается значение информационной составляющей в жизни нашего общества. Информационные ресурсы и технологии становятся все более ценными, что ведет к повышению значимости информационной безопасности. Специалисты в области информационной безопасности необходимы в государственных структурах, в научных учреждениях и коммерческих предприятиях.

В наше время проблема защиты информации в информационных системах особенно актуальна. Во-первых, расширилось использование компьютерных сетей (в частности Интернет), по которым передаются огромные объемы информации государственного и частного, военного и коммерческого характера, доступ к которой посторонних лиц крайне нежелателен. Во-вторых, появление мощных компьютеров, технологий сетевых и нейронных вычислений развеяло миф о надежности криптографических систем еще недавно считавшихся практически непробиваемыми.

Проблема защиты информации от прочтения посторонними лицами волнует людей с незапамятных времен. История криптографии – ровесница истории человеческого языка. Более того,

первоначально письменность сама по себе была криптографической системой, так как в древних обществах ею владели только избранные.

С широким распространением письменности криптография стала формироваться как самостоятельная наука. Первые криптосистемы встречаются уже в начале нашей эры. Так, Цезарь в своей переписке использовал уже более или менее систематический шифр, получивший его имя.

Бурное развитие криптографические системы получили в годы первой и второй мировых войн. Начиная с послевоенного времени и по нынешний день, появление вычислительных средств ускорило разработку и совершенствование криптографических методов.

Проблемой защиты информации путем ее преобразования занимается *криптология* (*kryptos* – тайный, *logos* – наука). Криптология – наука из двух ветвей: *криптографии* и *криптоанализа*. Цели их прямо противоположны.

*Криптография* – занимается поиском и исследованием математических методов преобразования информации. Следует различать криптографию и теорию кодирования сигналов, которая занимается преобразованием импульсов, идущих по каналам связи. При преобразовании информации криптографическими методами по каналу связи передаётся не сама защищаемая от извлечения противником информация, а результат её преобразования с помощью шифра.

Сфера интересов *криптоанализа* – исследование методов и способов расшифровывания информации без знания ключей.

Порядок изучения этих составляющих прямо следует из их целей, поскольку невозможно проводить криптоанализ без знания криптографических алгоритмов.

Для профессионального понимания криптографических алгоритмов и умения оценивать их слабые и сильные стороны, необходима предварительная серьезная математическая подготовка. Это объясняется тем, что современная криптография основана на результатах таких серьезных разделов математики, как теория сложности вычислений, теория чисел, алгебра, теория информации и др. Поэтому изучение элементов криптологии рекомендуется на старших курсах ВУЗов, специальность которых

связана с математикой. Однако нельзя не отметить, что знакомство с азами науки о шифровании будет полезно даже учащимся школ, проявляющим склонность и способности к математике и информатике.

Подготовленный нами курс “Основы защиты информации” разработан для студентов специальностей “прикладная математика” и “компьютерные сети и системы”.

В первом разделе курса изучаются основные понятия криптографии:

1. Терминология.
2. Подстановочные и перестановочные шифры.
3. Простое *xor*.
4. Одноразовые блокноты.
5. Компьютерные алгоритмы.
6. Большие числа.

Во втором разделе изучаются следующие вопросы криптографических протоколов:

1. Элементы протоколов:
  - введение в протоколы;
  - передача информации с использованием симметричной криптографии;
  - однонаправленные функции;
  - однонаправленные хэш-функции;
  - передача информации с использованием криптографии с открытыми ключами;
  - цифровые подписи;
  - генерация случайных и псевдослучайных последовательностей.
2. Основные протоколы:
  - обмен ключами;
  - удостоверение подлинности;
  - удостоверение подлинности и обмен ключами;
  - формальный анализ протоколов проверки подлинности и обмена ключами;
  - криптография с несколькими открытыми ключами;
  - разделение секрета;
  - совместное использование секрета.

В третьем разделе изучаются следующие криптографические

методы:

1. Длина ключа.
2. Управление ключами:
  - генерация ключей;
  - передача ключей;
  - проверка ключей;
  - использование ключей;
  - обновление ключей;
  - хранение ключей;
  - резервные ключи;
  - скомпрометированные ключи;
  - время жизни ключей;
  - разрушение ключей.

Четвертый раздел изучает криптографические алгоритмы:

1. Математические основы.
2. Блочные шифры.
3. Поточковые шифры.
4. Однонаправленные хэш-функции.
5. Алгоритмы с открытыми ключами.
6. Цифровая подпись с открытым ключом.
7. Схемы идентификации.
8. Алгоритмы обмена ключами.
9. Специальные алгоритмы для протоколов.

В последнем разделе изучаются основы криптоанализа.

В качестве литературы для самостоятельной работы учащихся можно рекомендовать основополагающую работу Шеннона [1], а также русскоязычные [2-4] и энциклопедическую работу Шнейера [5].

Теоретический материал данного курса достаточно сложен для восприятия, поэтому особое внимание следует уделять закреплению на практических занятиях. Некоторые разделы позволяют закрепление только посредством решения задач. В разделах криптографические алгоритмы и основы криптоанализа, есть широкие возможности для проведения закрепления и зачетов в нестандартной форме.

На начальном этапе полезно использовать специальный комплекс учебного программного обеспечения, имитирующего гипотетического противника (шифрующего или дешифрующего

сообщения).

Зачет по теме можно проводить в виде деловой игры.

Предлагается разбить учащихся на две команды. Одна команда будет выполнять функции криптографов, а другая – криптоаналитиков.

Задача криптографов за определённый промежуток времени зашифровать сообщение одним из ранее изученных методов. Криптоаналитики занимаются расшифровкой шифротекста предоставленного противниками. Криптографы должны предоставить команде соперников алгоритм шифрования и, возможно, открытый ключ.

Результат действий второй команды обратно пропорционален успешности работы первой команды, поэтому для равных условий далее возможно изменение функций команд и проведение второго тура игры. Успешность действий обеих команд оценивает преподаватель или судейская комиссия.

Поскольку проведение игры требует достаточно большого времени и дает большие возможности для сюжетной постановки, то возможно ее включение в неделю информатики, проводимую в учебном заведении.

В заключение следует отметить, что программа курса является ориентировочной и может изменяться в зависимости от специальности и уровня подготовленности контингента.

#### Литература:

1. Шеннон К. Работы по теории информации и кибернетике. – М.: ИЛ, 1963. – С. 333-369.
2. Жельников В. Криптография от папируса до компьютера. – М., 1996.
3. Введение в криптографию / Под общ. ред. В.В. Яценко. – М.: МЦНМО: «ЧеРо», 2000. – 288 с.
4. Анохин М.И., Варновский Н.П., Сидельников В.М., Яценко В.В. Криптография в банковском деле. – М.: МИФИ, 1997.
5. Schneier B. Applied Cryptography: Protocols, Algorithms and Source Code in C. – NY.: John Wiley & Sons, 1993.

## **ИСПОЛЬЗОВАНИЕ ПАКЕТА СИМВОЛЬНОЙ МАТЕМАТИКИ МАХИМА В КУРСЕ “КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ В НАУЧНЫХ ИССЛЕДОВАНИЯХ”**

Н.А. Леонова

г. Кривой Рог, Криворожский государственный педагогический университет

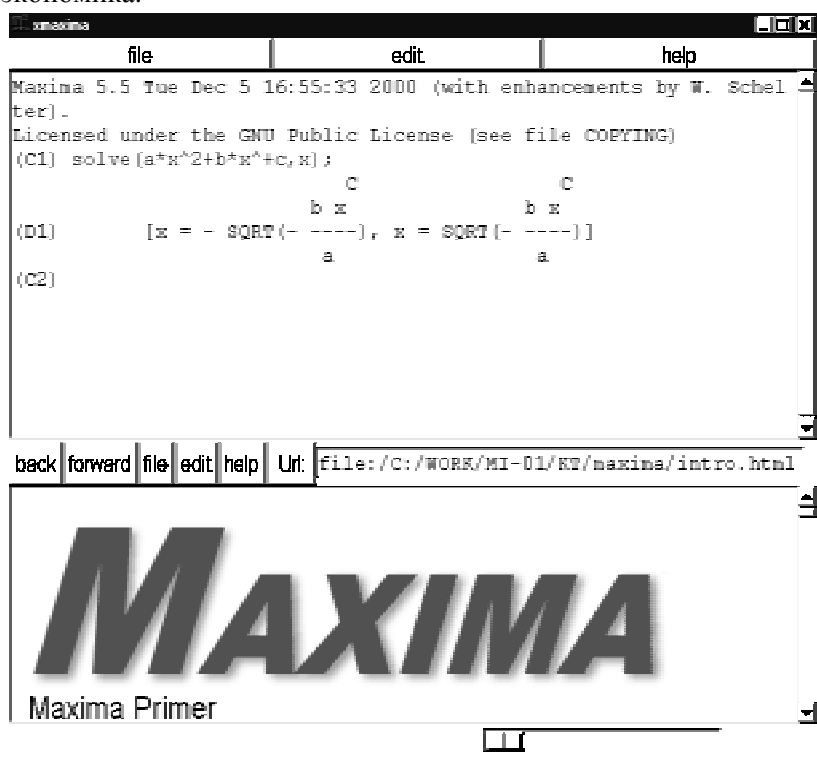
В Криворожском государственном педагогическом университете на физико-математическом факультете (специальность «Математика и основы информатики») в 2002–2003 учебном году была введена новая дисциплина «Компьютерные технологии в научных исследованиях». Цель этого курса – научить студентов применять возможности современных компьютерных технологий в научных исследованиях, например, при написании курсовых и дипломных работ, использовании различных методов расчетов, статистической обработки данных, графической интерпретации результатов эксперимента и т.д.

Еще совсем недавно перед любым учебным заведением стояли проблемы, связанные с отсутствием необходимой вычислительной техники или несоответствия ее современным требованиям [3]. Теперь эти проблемы отходят на второй план, но появляются новые и не менее простые. Одна из таких проблем – это использование лицензионного программного обеспечения. Поэтому все чаще наше внимание привлекают те программные продукты, которые имеют свободное распространение. Для проведения указанного курса было решено в качестве основы использовать пакет символьной математики *Mathima*.

*Mathima* – одна из программ для выполнения математических вычислений, символьных преобразований, а также для построения разнообразных графиков. Сложные вычисления оформляются в виде отдельных процедур, которые могут затем использоваться при решении других задач. Система распространяется под лицензией GPL и доступна как пользователям ОС Linux, так и пользователям Windows.

*Mathima* дает возможность специалистам решать большое количество достаточно сложных задач, не вдаваясь в тонкости программирования. Благодаря этому программа получила широ-

кое распространение в таких областях, как физика, биология, экономика.



Умение проводить аналитические расчеты – одно из главных достоинств этой программы. Maxima «умеет» преобразовывать и упрощать алгебраические выражения, дифференцировать и вычислять определенные и неопределенные интегралы, вычислять конечные и бесконечные суммы и произведения, решать алгебраические и дифференциальные уравнения и системы, а также разлагать функции в ряды и находить пределы. Кроме того, Maxima имеет стандартные дополнения для аналитических расчетов.

Для тех задач, которые невозможно решить аналитически, Maxima располагает большим количеством эффективных алгоритмов для проведения численных расчетов. Maxima позволяет решать задачи оптимизации (линейного программирования, нахождения экстремумов функций), а также задачи математиче-



ской статистики. В Maxima реализован адаптивный контроль точности, основанный на выборе внутренних алгоритмов, позволяющих ее максимизировать.

Пакет имеет, встроенную справочную систему с примерами использования тех или иных функций.

Система настолько гибка и универсальна, что может оказать неоценимую помощь в решении математических задач как школьнику, постигающему основы математики, так и будущему научному работнику, который использует математические методы для решения различных прикладных задач.

Курс рассчитан на 32 часа – 16 лабораторных работ. Предлагается следующая методика проведения лабораторных занятий: студент работает с документом-файлом, который представляет собой компьютерную систему заданий по теме занятия. Эта система готовится преподавателям средствами текстового редактора.

Методические рекомендации для каждой работы состоят из трех частей: теоретического материала с готовыми примерами работы, системы практических заданий различной степени сложности и контрольных вопросов по теме занятия [1]. Благодаря этому студенты могут выполнять каждую работу не только во время аудиторных занятий по расписанию, а и во время самостоятельной работы в компьютерном классе во внеурочное время.

В файле с компьютерной системой заданий, для акцентирования внимания студентов на существенных моментах, а также для наглядного отражения структуры представленного теоретического материала (разделы, подразделы, вопросы, подвопросы) используются различные средства выделения – цвет, размер символов и т.д.

Благодаря такой организации учебного процесса студенты при выполнении практических задач многократно обращаются к теоретической части материала, с помощью которого оказывается эффективное содействие по его усвоению. Кроме того, необходимое переключение из режима выполнения работы в режим просмотра теоретического материала закрепляет навыки работы со справочной системой. Таким образом, повышается и уровень индивидуализации обучения.

Предлагаем ориентировочную тематику и количество часов на каждую тему. Количество часов можно как увеличивать за счет добавления тем, так и уменьшать.

<b>№</b>	<b>Тема лабораторной работы и краткое содержание</b>	<b>Кол-во часов</b>
1	Знакомство с пакетом символьной математики <i>Math</i> (знакомство со средой, основные принципы работы, вызов справки и примеры использования функций).	2
2	Работа с выражениями (основные арифметические операции и их приоритет, работа с числами в экспоненциальной форме, работа с числами произвольной длины и точности, выбор точности).	2
3	Тригонометрические функции.	2
4	Алгебраические преобразования (раскрытие скобок, нахождение частного и остатка от деления многочленов, определение общего делителя многочлена, разложение многочлена на множители).	2
5	Алгебраические преобразования (учет дополнительных условий в неравенствах, тригонометрические выражения, работа с комплексными выражениями).	2
6	Решение уравнений (решение уравнения и системы алгебраических уравнений, тригонометрические уравнения, поиск решений в комплексной области).	2
7	Определение функций.	2
8	Построение графиков (построение графиков, построение изображений трехмерных объектов).	2
9	Операции математического анализа (нахождение производной, вычисление кратных производных функций многих переменных, вычисление неопределенных и определенных интегралов).	2
10	Операции математического анализа (вычисление конечных и бесконечных сумм, нахождение разложения функций в ряд Тейлора, вычисление пределов, исследование функций на непрерывность).	2
11	Матричные вычисления (сложение, умножение, вычисление определителя матрицы, нахождение транспонированной и обратной матрицы).	2

№	Тема лабораторной работы и краткое содержание	Кол-во часов
12	Решение систем уравнений матричным способом.	2
13	Программирование в системе Maxima	6
14	Зачет	2

При составлении данного курса ставилась задача собрать в одном дидактическом блоке набор сведений об основных функциях и приемах работы, а также сформировать практические умения и навыки путем проведения цикла лабораторных работ.

Курс ориентирован на краткое и четкое изложение основ работы с пакетами символьной математики и не претендует на полноту изложения материала – для этого существуют справочники, документация и контекстная помощь.

#### Литература

- 1 Шибанов В.А., Лумпова Р.И, Степанова О.А. Основы работы в системе MathCad 2000 Professional. // Информатика и образование. – 2002. – №9. – С. 15-19.
- 2 Леонова Н.А., Соловьев В.Н. Формирование научного мировоззрения средствами математического моделирования // Комп'ютерне моделювання та інформаційні технології в природничих науках: Зб. наук. праць. – Кривий Ріг: Видавничий відділ КДПУ, 2000. – С. 153–159.
- 3 Стрюк А.М. Особливості викладання інформатики для некомп'ютерних спеціальностей у вищих навчальних закладах. – Теорія та методика навчання математики, фізики, інформатики: Зб. наук. праць. – Кривий Ріг: Видавничий відділ КДПУ, 2001. – С. 164–169.

## ДЕКІЛЬКА ПИТАНЬ ПРО МЕТОДИ ВИКЛАДАННЯ ОЗНАЙОМЧИХ КУРСІВ ІНФОРМАТИКИ У ВНЗ

Р.В. Ліхачов

м. Кривий Ріг, Криворізький металургійний факультет Національної металургійної академії України

### **Старість – не радість, або “корінь зла”!**

Більшість сучасних методів викладання інформатики на етапі першого знайомства слухачів з обчислювальною технікою складаються з зашкарублених переказів “сивої давнини”, та марного витрачання дорогоцінного часу на спроби розтлумачити нікому не потрібні мови програмування.

Різноманітна література та методичні вказівки, що видаються й сьогодні, як в Україні взагалі, так і в кожному окремому ВНЗ, вбачають лише єдиний шлях вивчення предмету:

- розповіді про дві (максимум три) системи числення;
- абияке ознайомлення з апаратною частиною (зазвичай на прикладі легендарних IBM PC AT/XT, добре хоч не на PDP-11 чи ENIAC);
- легкого натяку на існування інформаційних мереж (частіше згадують ARPANET, ніж INTERNET);
- ознайомлення з операційною системою MS DOS (купка базових команд з сумнівним роз’ясненням щодо їх використання);
- поверховий розгляд простих мов програмування (основною багатьох методик є BASIC, проте останнім часом все більше авторів схиляються до PASCAL).

Відносна більшість цих видань є лише переспівом класичних підручників кінця 80-х – початку 90-х років минулого століття, коли такий підхід до вивчення курсу ще давав деякі результати. Звісно, декотрі викладачі (читай автори) пропонують вивчати більш розширений, на їх думку, матеріал, додаючи до перерахованого вище ознайомлення з ОС сімейства MS Windows (частіше 3.11, хоча є тенденція розглядання MS Windows 98 і навіть MS Windows 2000). Проте і такий “новаторський” підхід є лише окомилуванням та порожніми балачками, що викликано відсутністю бажання самозростання, небажанням перетнути старі кордо-

ни, та відсутністю нормальних лабораторій для практичних занять.

Величезна кількість годин, що могли б бути використані для практичних робіт, віддається на розгляд та усвідомлення слухачами термінів та понять, що ніколи не знадобляться їм у майбутньому, як при більш детальному вивченні обчислювальної техніки, так і при роботі з комп'ютером на рівні користувача.

Така ситуація не може не обурювати мислячу людину будь-якого віку чи фаху. Тому останнім часом деякі викладачі починають вказувати на необхідність вивчення альтернативних операційних систем чи навіть інших платформ. Звісно, це ще не революція, але маленький заколот – це точно.

Розглянемо ситуацію більш детально.

Є система вивчення ознайомчого курсу інформатики, що цілком прив'язана до певної операційної системи, яка використовується на певній платформі (мається на увазі ОС MS DOS чи MS Windows, та платформа IBM PC). Є давно схвалена і відпрацьована схема викладання. А ще є викладачі, згодні з таким станом діла.

Проте є, так би мовити, опозиція, яка складається з ентузіастів, що пропонують вивчати інші операційні системи та платформи. Мотивація останніх не безпідставна. Так, базовим питанням є відсутність коштів на придбання, як з'ясувалося, комерційних ОС, що складають основу старого методу навчання. Та намір перейти на безкоштовні (звісно ж у певній мірі) системи, що їх тепер пропонують різні компанії (мова йде про такі ОС, як Linux та FreeBSD). Деякі радять перейти на іншу платформу (Apple Macintosh), як більш досконалу і самодостатню, проте й дорожчу від тієї, що використовується сьогодні.

Зовні все це, й справді, виглядає як намір змінити типovu для багатьох ВНЗ ситуацію, проте – це лише заглиблення проблеми та гра на публіку, а можливо – віяння моди.

Зробимо деякі попередні підсумки.

Ми маємо дві абсолютно схожі цукерки, що відрізняються обгортокою, а на смак однакові. Кардинальних же змін в двох, нерідко ворожих, таборах не помітно.

А й справді, яка різниця, чи вивчати одну операційну систему, чи іншу, чи навіть обидві разом? А яка різниця, на якій плат-

формі все це відбувається? Проблема лишається – за деревами не видно лісу!

Вивчаючи щось конкретне та специфічне, що не завжди можна зустріти у звичайному житті, студенти не розглядають предмет в цілому, не бачать спільності процесів, схожості, зовні різних, систем.

### **Що день наступний нам готує?**

Яка користь з фахівців, які вміло користуються деякими програмами певний час, а потім, коли “вітер змінився”, не в змозі навіть знайти аналогічні, не кажучи вже про їх використання, під управлінням іншої операційної системи, чи на іншій платформі.

Використовуючи старий підхід для ознайомлення з обчислювальною технікою, ми підписуємо “смертний вирок” тим користувачам, що прийшли нам на зміну і, без сумніву, надходять ще. Люди, що вивчають предмет старими методами, у наш швидкозмінний час, будуть не пристосовані і знаходитимуться обабіч еволюції комп’ютерної техніки.

А час диктує дедалі жорстокіші умови. Кількість реально конкуруючих операційних систем та платформ (включаючи КПК, що стрімко розвиваються і перехоплюють ринок у настільних систем) зростає з кожним роком. Можливості сучасної техніки вже не обмежуються чотирма математичними діями, а все частіше стають потрібні там, де раніше ми цього не передбачали.

Все це дає поштовх для пошуку нових методів викладання інформатики у ВНЗ.

### **Не питай мене чому...**

Власне, існує ще декілька проблем, однією з яких є небажання чи невміння пояснити базові поняття. Більшість методик пропонують лише короткі відомості про достатньо велику область знань, замість акцентування уваги на спільності та взаємозв’язку процесів.

Так, декотрі пропонують вивчення базових команд та простих програм операційної системи MS DOS (чи будь якої іншої, як то Linux або Windows), навіть не пояснюючи, що власне є операційні системи та які їх спільні риси.

Студентам пропонують сліпо вірити в те, що наданий їм ознайомчий матеріал є догмою, що окрім розглянутого у курсі більше не існує альтернатив, а якщо вони є, то недостатньо розвинені.

Викладачі пояснюють двійкову систему числення, не пояснюючи, чому саме її покладено в основу курсу, та навіть не акцентуючи уваги на тому, що системи числення можна створювати за власним бажанням, з огляду на те, яку апаратну базу використано та які цілі переслідуються.

Volkov або Norton Commander вивчають як “Отче наш ...”, не пояснивши, що таке взагалі файлові менеджери та психологічний аспект їх використання (вести мову про ергономіку чи юзабіліті взагалі вважається недоречним). Це призводить до ситуації, коли студент не відрізняє операційну систему від файлового менеджера і формує у власній уяві стійку асоціацію, що MS DOS – це “дві синенькі панелі”.

Ситуація, коли методика не поновлюється декілька років, є критичною сама по собі, а в сфері комп’ютерних технологій – утопічною.

### **Мімікрія**

Вміння пристосовуватись до швидко змінних умов повинно стати єдиною і найнеобхіднішою рисою нової методики викладання інформатики.

### **Що робити?**

Звісно, не можна зіштовхнути з місця будь-яке діло, поклавши декілька прямих запитань, та побачивши опісля них наявність проблеми потрібно як най швидше її вирішувати.

Змінювати концепцію викладання треба не сьогодні, а вчора!

Нова методика повинна враховувати взаємозв’язок та спільні риси як апаратних платформ, так і теоретичних відомостей про операційні системи, що їх тепер повно.

Потрібно робити акцент на те, що в інформаційній сфері весь час йде еволюція, нові поняття – це не щось взяте “з неба”, а необхідність (як, наприклад, перехід до багатозадачності, чи підвищення безпеки систем). Проте все нове зовсім не заперечує існуючих раніше концепцій (за деяким винятком) і розуміючи

фундаментальні речі, можна перекваліфікуватись та працювати на будь-якій системі. Таким фундаментом є поняття про засоби зберігання інформації (файли та деревовидна модель каталогів), можливі маніпуляції з нею (копіювання, переміщення, правка тощо), спільність графічних інтерфейсів та розуміння статусу операційної системи як інтерфейсу між користувачем та “залізом”.

Студенти повинні знати, що машина створена для людини, а не людина для машини. Потрібно пояснити, що внутрішня структура комп'ютера повторює структуру мислення людини (оперативна пам'ять – для тимчасового зберігання інформації, будь-який носій постійного збереження – альтернатива нашій пам'яті, системи зв'язку між ними). Операційні системи калькують з нашого побуту засоби роботи з інформацією (файли – окремі документи, що складаються в папки та іменуються за типом інформації).

Тобто маємо такі прості тези:

- кожен комп'ютер має апаратну частину, що складається з засобів зберігання інформації та її вводу/виводу (необхідно підкреслити схожість різних платформ у виборі концепції побудови обчислювальних машин);
- рівнем вище, як інтерфейс користувача з апаратною частиною, йде операційна система (пояснити різницю та необхідність створення одно- та багатозадачних систем);
- далі будь-який менеджер, що відображає інформацію у зручній для користувача формі, та дозволяє з нею працювати (зауважити, що саме він є зв'язком між людиною та ОС);
- різноманітні програми, що відповідають за конкретні функції (наголосити на їх сумісності навіть між різними ОС, та різноманітності альтернативних варіантів);
- вищий рівень – протоколи обміну інформацією між різними робочими станціями незалежно від платформи чи встановленої ОС.

Такий простий погляд на речі, що здаються значно складнішими, ніж є насправді, дозволяє вагомо підвищити рівень знань студентів і не є зацикленним на конкретні системи, що робить його більш динамічним та прозорим для розуміння, не залежно від



часу.

### **Перечитуємо історію, або “I’ll be back...”**

Спільність різних систем легко помітити, заглянувши у минуле. Так, для розтлумачення графічних інтерфейсів користувача (GUI) просто необхідно згадати, звідки вони з’явилися, хто перший почав їх впровадження та як вони змінювались з часом. Це достатньо вдалий прийом для пояснення необхідності розширеного вивчення курсу.

Переглянувши вклад Хероx у розвиток графічного інтерфейсу (їх розробка інтуїтивно зрозумілих елементів, що були підказані життям – робочий стіл, корзина тощо), подальше впровадження цієї ідеї компанією Apple (задум Хероx вперше з’являється у настільних системах, як реально працююча та зручна ОС), спроба перехопити ринок компанією Microsoft власною реалізацією концепцій Хероx та Apple (вихід у світ Windows), можна легко зрозуміти, як тісно пов’язані всі системи, що ми їх маємо сьогодні. Це зовсім не кінець еволюційного шляху.

Так, на тепер існує цілий ряд конкуруючих компаній, що впроваджують у життя свої реалізації графічних оболонок, не зважаючи на постійне оновлення систем від двох безсумнівних лідерів – Apple та Microsoft. На суд користувачів виходять нові дизайнерські рішення та переосмислення концепцій зручності все більше і більше в останній час (рядок Mac OS 8, 9; лінійка Windows 95, 98, 2000; графічні оболонки KDE та GNOME для UNIX-подібних систем; неймовірна за оригінальністю та простою Beos; апогей творчої думки – Mac OS X, KDE 3.0 та Windows XP). Лідером у створенні нових інтерфейсів та впровадження сміливих рішень, як апаратних, так і програмних досі залишається компанія Apple, хоча, нажаль, це і не дуже помітно в наших умовах.

### **Ось і казочці кінець...**

Звісно, неможливо зараз відмовитись від багатьох напрацьованих матеріалів та деяких елементів курсу, що стосуються практичних занять. Проте це не привід не розглядати альтернативних варіантів взагалі.

Нехай відсутність матеріальної бази зв’яже нам руки, проте

вона не повинна зв'язувати наших думок.

Викладачам, як нікому іншому не можна залишатись обабіч нових комп'ютерних технологій, що стрімко розвиваються. В першу чергу вчитися потрібно самим, увесь час бути у формі, розглядати новітні технології одразу з їх появою, треба завжди дивитися у майбутнє, мріяти і шукати втілення своїх мрій, бо саме таке ставлення до справи призводить до прекрасних результатів.

Мрійники та ентузіасти з Хегох, що не мали під рукою технологій, достатніх для втілення своїх мрій, замислювались над зручністю використання майбутніх машин і призвели до революції настільних систем, які ми використовуємо зараз.

Мрія, творчий пошук, постійне навчання, спроби зрозуміти спільність усіх ідей – ось єдина методика вивчення курсу інформатики у ВНЗ, що потрібна нам сьогодні.

## ВОЗМОЖНОСТИ НАПИСАНИЯ ОБУЧАЮЩИХ ПРОГРАММ ПОД WINDOWS

С.С. Лихман

г. Кривой Рог, Криворожский металлургический факультет Национальной металлургической академии Украины

Операционная система Windows компании Microsoft является на данный момент широко распространенной операционной системой. Обладая такими удобствами, как современный пользовательский интерфейс, богатый сервис, широкие возможности настройки, хорошая справочная система, она нашла своих пользователей во всем мире.

Эта ОС предоставляет пользователю графический интерфейс (GUI) и мультимедиа средства, позволяющие наиболее эффективно проводить обучение, поэтому не удивительно, что часто в учебных заведениях можно встретить именно эту операционную систему.

Программисты компании Microsoft предусмотрели возможность создания приложений, направленных на интерактивное обучение пользователей. Для этого в состав Windows API включены специальные функции, однако для их использования необходимо познакомиться с системой ловушек Windows и понятием о сообщениях.

Так как Windows является мультизадачной операционной системой и построена на концепции окон, то для общения между задачами и окнами используются сообщения.

Сообщение передается в виде заранее определенного кода, к которому часто добавляются дополнительные параметры. Любое приложение под Windows обрабатывает сообщения и переданные с ним параметрами в специальной функции обработки сообщений.

Пользователь взаимодействует с операционной системой с помощью клавиатуры и мыши. Драйверы этих устройств посылают сообщения активным окнам.

Система перехвата сообщений позволяет анализировать сообщения еще до того, как они попадут к целевым окнам, решать, передавать ли их вообще. Таким образом, имеется возможность

наблюдать и контролировать действия пользователя. Например, при ошибке приложение может подсказать, как поступить или засчитать ошибку, если это тест.

В операционной системе Microsoft Windows хуком (хук – от оригинального названия *hook*) называется механизм перехвата особой функцией событий (таких как сообщения, ввод с мыши или клавиатуры) до того, как они дойдут до приложения. Эта функция может затем реагировать на события и, в некоторых случаях, изменять или отменять их. Функции, получающие уведомления о событиях, называются *фильтрующими функциями* и различаются по типам перехватываемых ими событий. Пример – фильтрующая функция для перехвата всех событий мыши или клавиатуры. Чтобы система смогла вызывать функцию-фильтр, эта функция должна быть установлена – то есть, прикреплена – к хуку (например, к клавиатурному хуку). Прикрепление одной или нескольких фильтрующих функций к какому-нибудь хуку называется *установкой* хука. Если к одному хуку прикреплено несколько фильтрующих функций, система реализует очередь функций, причем функция, прикрепленная последней, оказывается в начале очереди, а самая первая функция – в ее конце.

Когда к хуку прикреплена одна или более функций-фильтров и происходит событие, приводящее к срабатыванию хука, система вызывает первую функцию из очереди функций-фильтров. Это действие называется вызовом хука. Для установки и доступа к фильтрующим функциям приложения используют функции `SetWindowsHookEx` и `UnhookWindowsHookEx`.

Хуки предоставляют мощные возможности для приложений Windows. Приложения могут использовать хуки в следующих целях:

1. Обработать или изменить все сообщения, предназначенные для всех диалоговых окон (`dialog box`), информационных окон (`message box`), полос прокрутки (`scroll bar`), или меню одного приложения (`WH_MSGFILTER`).

2. Обработать или изменить все сообщения, предназначенные для всех диалоговых окон, информационных окон, полос прокрутки, или меню всей системы (`WH_SYSMSGFILTER`).

3. Обработать или изменить все сообщения в системе (все виды сообщений), получаемые функциями `GetMessage` или

PeekMessage.

4. Обработать или изменять все сообщения (любого типа), посылаемые вызовом функции SendMessage.

5. Записывать или проигрывать клавиатурные и мышечные события (WH\_JOURNALRECORD, WH\_JOURNALPLAYBACK).

6. Обработать, изменить или удалять клавиатурные события.

7. Обработать, изменить или отменять события мыши.

8. Реагировать на определенные действия системы, делая возможным разработку приложений компьютерного обучения – computer-based training (CBT).

9. Предотвратить вызов другой функции-фильтра.

Чтобы написать приложение для интерактивного обучения (CBT application), разработчик должен координировать его работу с работой приложения, для которого оно разрабатывается. Для достижения этой цели операционная система предоставляет разработчикам хук WH\_CBT. Система передает фильтрующей функции код хука, показывающий, какое произошло событие, и соответствующие этому событию данные.

Для каждого типа хука имеется свое определение функции-фильтра.

Функция-фильтр для перехвата сообщений хука типа WH\_CBT имеет следующее определение

```
LRESULT CALLBACK CBTProc(int nCode, WPARAM wParam, LPARAM lParam);
```

Фильтрующие функции принимают три параметра: *nCode* (код хука), *wParam* и *lParam*. Код хука – это целое значение, которое передает функции дополнительную информацию, определяет действие, из-за которого произошел вызов функции-фильтра:

HCBT\_ACTIVATE – система активизирует окно,

HCBT\_CLICKSKIPPED – система удалила сообщение от мыши из очереди сообщений. Получив этот код, приложение должно установить хук типа WH\_JOURNALPLAYBACK.

HCBT\_CREATEWND – система создает окно. Функция-фильтр вызывается перед посылкой сообщений WM\_CREATE или WM\_NCCREATE создаваемому окну. Если функция возвра-

тит ненулевое значение, система не создаст окно, в противном случае окно создается.

HCBT\_DESTROYWND – система удаляет окно.

HCBT\_KEYSKIPPED – система удалила сообщение от клавиатуры из очереди сообщений. Получив этот код, приложение должно установить хук типа WH\_JOURNALPLAYBACK.

HCBT\_MINMAX – система сворачивает или разворачивает окно.

HCBT\_MOVESIZE – система изменяет размеры окна или перемещает его.

HCBT\_SETFOCUS – окно получает фокус ввода.

HCBT\_SYSCOMMAND – выполняется системная команда, позволяет запретить переключение задач и обработку горячих клавиш.

*WParam, LParam* зависят от параметра *nCode*.

*Возвращаемое значение* определяет разрешить ли выполнение операций, из-за которых произошел вызов. 0 – разрешить выполнение, 1 – запретить.

Имя функции *CBTProc* может заменяться любым другим по желанию программиста.

Приложение устанавливает функцию-фильтр указывая тип хука (параметр *idHook*) WH\_CBT в функции *SetWindowsHookEx*:

```
HHOOK SetWindowsHookEx(int idHook, // WH_CBT
```

```
HOOKPROC lpfn, //указатель на функцию-фильтр
```

```
HINSTANCE hMod, //указатель на
```

```
DWORD dwThreadId //идентификатор потока, для которого устанавливается хук, 0 – для всех
```

```
);
```

Если установка хука успешна, функция возвращает идентификатор функции-фильтра, который передается функции *UnhookWindowsHookEx*, чтобы отключить хук. Это необходимо сделать до завершения приложения, чтобы освободить системные ресурсы.

Все вышеизложенное показывает, что ПК с Windows удобно использовать для создания учебных приложений.

Кроме того, ПК часто используются и для тестирования и позволяют реализовать все требования к тестам.

ПК дает дополнительные возможности при тестировании.

Например, построение адаптивных тестов, основанных на возможности изменения тестовых воздействий на основе быстрого анализа ответов; расчет косвенных критериев по тестам; проведение нестандартных видов тестов; выбор тестовых заданий из большой обновляющейся базы; защита базы тестовых заданий; сохранение результатов, создание отчетов по тестированию и т.п.

Совмещая в приложении одновременно мультимедийный учебный материал и тесты, можно получить довольно мощный обучающий комплекс.

На такое приложение можно возложить также ряд дополнительных функций, таких как автоматический подсчет результатов тестов, ведение статистики прогресса обучения, рекомендация повторов некоторых этапов обучения и окончательное тестирование. Для проведения обучения нет необходимости эмулировать изучаемое программное обеспечение, что является довольно сложной задачей, особенно для больших продуктов. Можно работать с реальным программным продуктом, отслеживая действия пользователя и при необходимости даже показывая как нужно поступить.

#### Литература

1. Microsoft Development Network Library, July – 2002 <http://msdn.microsoft.com/>
2. Усков В.Л., Ускова М. Информационные технологии в образовании // Информационные технологии. – 1999. – №1.
3. Кашицин В.П. и др. Состояние и развитие дистанционного образования в мире: Научно-аналитический доклад. – М.: Магистр, 1997.

## ДЕЯКІ АСПЕКТИ ВПРОВАДЖЕННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРИ ВИКЛАДАННІ КОМП'ЮТЕРНИХ ДИСЦИПЛІН У ВИЩІЙ ШКОЛІ

Л.Б. Ліщинська

м. Вінниця, Вінницький торговельно-економічний інститут Ки-  
ївського національного торговельно-економічного університету  
[L\\_Fil@rambler.ru](mailto:L_Fil@rambler.ru), [L\\_Fil1@mail.ru](mailto:L_Fil1@mail.ru)

Застосування інформаційних технологій в освіті пов'язано, з одного боку, з вирішенням задач використання технічних засобів, тобто, як забезпечити навчання цими засобами, як підвищити ефективність навчання з їх допомогою, з іншого боку, необхідно визначити, чому треба навчати, якою повинна бути система навчання, які завдання вона повинна вирішувати, яким має бути майбутній фахівець.

Крім цього, у наш час оволодіння сучасними знаннями у фаховій області не можливо без активізації самостійної роботи студентів, яка суттєво доповнює об'єм знань, отриманих традиційними методами, і є обов'язковою при виконанні інших видів навчальних робіт, особливо при підготовці курсових та дипломних робіт.

У діючих навчальних планах, зокрема для студентів-економістів, спостерігається тенденція до зменшення аудиторних занять та збільшення їх самостійної роботи, яка повинна бути постійно контролюватися з боку викладача.

З огляду на таку специфіку навчання, викладач не має можливості, та в загалом це і не є необхідним, приділяти велику кількість часу на аудиторних заняттях детальному розгляду окремих специфічних питань навчальної дисципліни.

Значне місце займає також наочність подання матеріалу, його візуальне супроводження, що сприяє усвідомленню та стійкості засвоєння знань. Наочність у навчанні підвищує зацікавленість, активізує пізнавальну діяльність студентів, відповідно сприяє реалізації і інших дидактичних принципів.

Особливе місце візуальне супроводження навчання для подання учбового матеріалу та його представлення займає при викладанні дисциплін комп'ютерного напрямку для студентів-



економістів. Наочність у навчанні розповсюджується на всі форми та методи одержання знань: лекції, практичні та лабораторні заняття, комп'ютерний практикум, організація та проведення ділових ігор, представлення та захист студентами курсових та дипломних робіт, тощо.

Сьогодні при інтенсивному розвитку комп'ютерних інформаційних технологій все більш широке розповсюдження знаходять: навчальні презентації, відеофільми з розглядом ситуаційних задач, тощо; що стало можливим завдяки використанню для цього в єдиному комплекті персонального комп'ютера (зручніше всього – портативного) та мультимедійного проектора.

Розглянемо використання візуального супроводження на основі застосування комп'ютерних інформаційних технологій для різних форм подання інформації як для навчання студентів, так і подальшого їх представлення результатів своєї самостійної роботи.

На лекції важливо змусити студентів уважно слухати і сприймати матеріал, що читається лектором. Для цього потрібно викликати підвищену зацікавленість до лекцій, що читаються, та при виборі методів викладання враховувати специфіку дисциплін, особливо комп'ютерно-орієнтованого напрямку.

Так, до дисциплін комп'ютерно-орієнтованого напрямку, що викладаються у Вінницькому торговельно-економічному інституті (ВТЕІ) КНТЕУ, відносяться “Інформатика та комп'ютерна техніка”, “Комп'ютерний практикум”, “Комп'ютерні мережі”, “Інтернет у бізнесі”, “Інформаційні технології та системи” та ряд дисциплін, пов'язаних з автоматизацією різних галузей економічної-фінансової діяльності.

Специфіка викладання комп'ютерних дисциплін полягає у тому, що без використання візуального супроводження просто неможливо на високому рівні навчити студентів вирішувати основні, а тим більше складні задачі за допомогою сучасних прикладних програм, тобто процес лекції зводиться до констатації фактів, в кращому випадку – запису алгоритмів, тощо, але студенти не можуть уявити, як саме виконуються ці послідовності команд для розв'язку конкретної задачі.

У ВТЕІ КНТЕУ інтенсивно впроваджується ситуаційний метод викладання, зокрема при вивченні дисциплін комп'ютерного

напрямку, детально не розглядається інтерфейси роботи з програмами, а студенти навчаються застосувати ті, чи інші можливості програми для вирішення конкретної фахової ситуації.

Тому, на лекціях з дисциплін комп'ютерного напрямку є доцільним використання мультимедійного проектора та портативного комп'ютера. Для того, щоб лекція була проведена на високому методичному рівні, вона потребує великої підготовчої роботи. Зрозуміло, що на перших порах використання допоміжних технічних засобів створює для лектора додаткові складності, але з часом процес читання лекцій значно полегшується завдяки використанню комп'ютерної техніки.

За допомогою мультимедійного проектора висвічуються на екрані: тема лекції, її план, список літератури, перелік основних термінів за темою лекції, що готуються заздалегідь за допомогою, наприклад, програм Microsoft Word (у вигляді текстового файлу), або Microsoft Power Point (у вигляді слайдів презентації).

Для розкриття основних питань лекції може бути застосовано декілька варіантів подання візуальної інформації:

- у вигляді слайдів, що містять необхідні структурні схеми, алгоритми, діаграми, тощо;
- викладач розкриваючи алгоритм, супроводжує його відповідною демонстрацією на комп'ютері, що відображається на великому екрані;
- використання електронних підручників, мультимедійних навчаючих курсів, тощо.

При візуальному супроводженні лекції необхідно дотримуватись таких вимог:

- його використання повинно бути методично доцільним, сприяти розкриттю основної ідеї лекції;
- необхідно чітка мета використання наочних засобів у структурі лекції (коли ввести, з якою метою, який висновок випливає);
- не перевантажувати лекцію наочністю – це знижує увагу, самостійність і активність студентів в осмисленні навчального матеріалу;
- потрібна візуальна ілюстрація для найбільш суттєвого матеріалу при розкритті теми;
- демонстраційний матеріал повинен бути допоміжним для

лектора, а не замінити його; тощо.

При проведенні лабораторних практикумів використання візуального супроводження дозволяє спочатку продемонструвати основні етапи виконання лабораторних робіт з наступним їх повторенням кожним студентом на своєму комп'ютері. Для розв'язку складних задач буде доцільним синхронне поетапне виконання завдання викладачем та студентами.

У Вінницькому торговельно-економічному інституті КНТЕУ при вивченні дисципліни “Комп'ютерний практикум” запроваджено проблемно-ситуаційний підхід, який полягає у тому, що центром є віртуальне (або реальне) підприємство, його проблеми, задачі, документи, контакти з клієнтами та постачальниками, тощо. Фактично, студенти імітують ситуацію зі своєї майбутньої професійної діяльності. Кожний студент в межах цієї дисципліни вирішує комплексну задачу по створенню інформаційної системи цього підприємства, застосовуючи знання та навички, придбані протягом вивчення дисципліни “Інформатика та комп'ютерна техніка” та відповідних фахових дисциплін, з наступною презентацією свого підприємства, тобто результатів своєї роботи, з використанням мультимедійних засобів.

У подальшому навчанні студенти застосовують засоби візуального супроводження при представленні та захисті курсових та дипломних робіт, після завершення вищого навчального закладу у своїй професійній діяльності.

Таким чином, підвищення ефективності навчання у сучасній вищій школі, усунення пробілів у знаннях, збільшення зацікавленості студентів у навчальному матеріалі можна досягти за рахунок впровадження в навчальний процес новітніх (особливо, інформаційних) технологій навчання, що стимулюють студентів до самоосвіти і змінюють роль викладача в цьому процесі.

## ІНТЕРНЕТ ЯК ЗАСІБ РОЗВИТКУ ПІЗНАВАЛЬНОЇ САМОСТІЙНОСТІ СТУДЕНТІВ

В.Г. Логвіненко

м. Суми, Сумський національний аграрний університет

Навчальна діяльність з інформаційних технологій у вузі передбачає вивчення форм людської діяльності, що зв'язані з процесами перетворення інформації за допомогою комп'ютера і телекомунікаційних засобів зв'язку та їх взаємодією із середовищем застосування. Однією з тем, що вивчається в рамках курсу, є “Основи Інтернет”.

Відомо, що з появою Інтернет можливості використання персонального комп'ютера значно розширилися, а користувачі отримали могутній засіб роботи із постійно зростаючим обсягом інформації. Зміни відбулися і в системі вищої освіти. Сьогодні кожен студент, що отримує вищу освіту, повинен:

– мати конкретні навички з використання технічних засобів (в т.ч. персонального комп'ютера і комп'ютерних мереж);

– вміти: 1) добувати інформацію з різних джерел: як із періодичної літератури, так і з електронних комунікацій; 2) переробляти отриману інформацію; 3) представляти її в зрозумілому вигляді; 4) ефективно її використовувати;

– знати особливості інформаційних потоків у сфері своєї діяльності.

І тому актуальним стає питання навчати студентів самостійно працювати, розвивати їх пізнавальну самостійність.

В психолого-педагогічній літературі ([1–4]) з проблеми пізнавальної самостійності наголошується на те, що пізнавальна самостійність – це одна з властивостей особистості, яка характеризується такими основними факторами: по-перше, сукупністю знань, умінь та навичок, якими володіє особистість; по-друге, відношенням особистості до процесу діяльності, до результатів і умов її виконання, а також зв'язками з іншими людьми, що виникають в процесі цієї діяльності.

Пізнавальна самостійність проявляється в самостійній пізнавальній діяльності, і від викладача залежить правильна організація такої діяльності. Існують різні підходи до практичного фор-

мування пізнавальної самостійності в навчанні. На нашу думку, при опануванні теми “Основи Інтернет” поєднання наступних форм самостійних робіт якнайбільше сприяє розвитку пізнавальної самостійності студентів:

1) самостійні роботи, які виконуються за інструкцією. Вони носять пізнавально-узагальнюючий характер, тому що відбувається формування нових понять та навичок роботи в Інтернет. В основі цих робіт – репродуктивний спосіб діяльності. Роль викладача – консультація студентів протягом виконання роботи, подолання утруднень, що виникають у студентів.

2) самостійні роботи індивідуального характеру. Вони носять систематизуючий й характер, тому що відбувається засвоєння основних положень Інтернет та основних технологій роботи. В основі цих робіт – продуктивний спосіб діяльності. Роль викладача – згідно індивідуальних або професійних інтересів студентів підготувати завдання.

Розглянемо розподіл самостійних робіт:

1) **Робота №1. Тема:** Робота з браузером Internet Explorer.

**Мета:** отримання практичних навичок роботи та використання головних можливостей браузерів.

**Основні поняття:** комп’ютерна мережа; глобальна комп’ютерна мережа Інтернет; HTML; URL; IP-адресація; доменна система імен; протокол передачі даних TCP/IP; web-браузер; web-сайт; гіперпосилання; закладка; доступ в Інтернет; FTP-сервери; послуги Інтернет.

**Основні навички:** запуск програми-браузера; використання панелі інструментів браузера; перехід до потрібного web-сайта; встановлення закладки; запам’ятовування поточної web-сторінки; робота з декількома вікнами.

**Робота №2. Тема:** Пошук інформації в Інтернет.

**Мета:** отримання практичних навичок роботи та використання різних пошукових систем і практичне застосування стратегій пошуку інформації в Інтернет.

**Основні поняття:** стратегія пошуку інформації в Інтернет; пошукова система; типи пошукових систем; технологія пошуку інформації.

**Основні навички:** технологія пошук за ключовим словом; технологія пошуку за декількома ключовими словами; техноло-

гія зберігання знайденої інформації для подальшого використання.

**Робота №3. Тема:** Робота з електронною поштою. Outlook Express.

**Мета:** отримання практичних навичок листування за допомогою електронної пошти.

**Основні поняття:** електронна адреса; основи листування за допомогою електронної пошти; Outlook Express; призначення папок в Outlook Express; створення повідомлення; відправлення повідомлення; створення власної поштової скриньки.

**Основні навички:** запуск Outlook Express; технологія створення повідомлення; технологія відправлення повідомлення; технологія створення власної поштової скриньки.

**Робота №4. Тема:** Створення власної web-сторінки. Мова HTML.

**Мета:** отримання практичних навичок створення web-сторінки на мові HTML.

**Основні поняття:** мова HTML; тег HTML; атрибути тега; структура документа HTML; призначення основних тегів; гіперпосилання HTML; розміщення малюнків в документ; технологія створення web-сторінки; технологія розміщення web-сторінки.

**Основні навички:** технологія створення найпростішої web-сторінки; технологія розміщення найпростішої web-сторінки.

**Робота №5. Тема:** Створення власної web-сторінки з використанням Word.

**Мета:** отримання практичних навичок створення web-сторінки з використанням Word.

**Основні поняття:** можливості Word по створенню web-сторінки; технологія створення web-сторінки.

**Основні навички:** технологія створення web-сторінки з використанням Word.

2) **Робота №6. Тема:** Пошук інформації (за індивідуальною темою).

**Мета:** використання стратегій пошуку інформації в Інтернет.

**Основні поняття:** закріплення понять роботи №1, №2, №3.

**Основні навички:** закріплення навичок роботи №1, №2, №3.

**Робота №7. Тема:** Створення web-сторінки (за індивідуальною темою).

**Мета:** використання знань та навичок створення web-сторінки.

**Основні поняття:** закріплення понять роботи №4, №5.

**Основні навички:** закріплення навичок роботи №4, №5.

При виконання робіт, особливо робіт №6-7, студент буде:

- залучати різні джерела інформації професійного напрямку, а також підручники та посібники по Інтернет;
- спілкуватися із своїми колегами-студентами, радитися з викладачем;
- складати план своєї роботи та ставити цілі своєї діяльності, тобто самостійно організовувати свою діяльність;
- прикладати свої вольові зусилля для виконання цієї роботи;
- оцінювати свою діяльність по ходу виконання.

Таким чином, представлена організація роботи студентів широко впливає на розвиток їх пізнавальної самостійності.

#### Література.

1. Педагогическая энциклопедия. Т. 3. Н–См. – М.: Советская энциклопедия, 1966. – 879 с.
2. Олейник Р.В. Развитие познавательной самостоятельности студентов. – Автореф. дис. ... к.п.н. – 13.00.01. – Х., 1991. – С. 19.
3. Кулагина Г.Н. Формирование у студентов вечернего отделения познавательной самостоятельности и активности (в процессе обучения на младших курсах). – Автореф. дисс. ... к.п.н. – 13.00.01. – М., 1980. – 16 с.]
4. Федорков Ч.М. Воспитание учебно-познавательной самостоятельности в процессе изучения естественно-математических наук – Автореф. дисс. ... к.п.н. – 13.00.01. – Минск, 1988. – 19 с.

## МІСЦЕ ОБ'ЄКТНО-ОРІЄНТОВАНОГО АНАЛІЗУ У ПРОЦЕСІ НАВЧАННЯ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ

І.М. Лукаш

м. Чернігів, Чернігівський державний педагогічний університет  
імені Т.Г. Шевченка  
Lukash@cg.ukrtel.net

Об'єктно-орієнтоване програмування (ООП) сьогодні вже зайняло провідне місце у курсах інформатики та програмування вищих навчальних закладів. Вивчення цього матеріалу не тільки сприяє підготовці високо кваліфікованих спеціалістів, здатних швидко адаптуватися до сучасних об'єктно-орієнтованих технологій, а й є базовим інструментом розвитку інтелектуальних здібностей людини, що вважається складовою її культурологічного розвитку.

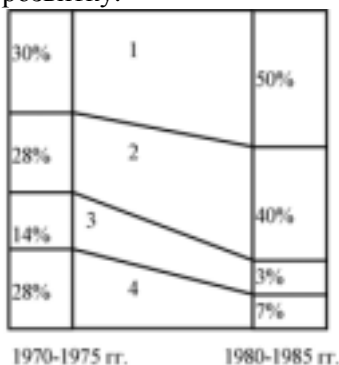


Рис. 1. Структура затрат праці на фазах реалізації програмного проекту:

- 1- супровід і підтримка;
- 2- аналіз предметної галузі і постановка задачі;
- 3- програмування та автономне налагодження;
- 4- комплексне налагодження та іспит.

В 70-х роках минулого століття вважалося [3], що у процесі розробки програмний проект повинен пройти через такий ланцюг спеціалістів:

- кінцевий користувач;
- системний аналітик;
- системний аналітик-програміст;
- прикладний програміст;
- системний програміст.

Як показано на рис. 1, вже у 80-х роках на аналіз предметної галузі і постановку задачі витрачалося 40% затрат праці і тільки 3% на програмування. Більша половина загальної чисельності світового корпусу програмістів –



системні аналітики не пишуть жодного рядка програмного коду, а займаються перетворюванням мети, призначення, технічних характеристик, умов правильності розв'язування у загальні формальні вимоги на програмний проект. На наступному етапі системний аналітик-програміст перетворює загальні вимоги у детальні специфікації на окремі програми, розроблює логічну структуру проекту.

Якщо навчання програмуванню проводиться на основі об'єктно-орієнтованої парадигми, то вище сказане обумовлює вивчення методів і відповідних засобів проведення об'єктно-орієнтованого аналізу (ООА) як важливого елемента технології ООП.

Під ООА розуміють методологію описання складних систем, при якій вимоги до системи сприймаються з точки зору класів і об'єктів, виявлених в предметній галузі [1]. В реальному житті більшість систем є складними, складно охопити одним поглядом всі її деталі. В залежності від типу системи вона може бути описана за допомогою кількох взаємопов'язаних уявлень про її архітектуру: вигляд з точки зору випадків використання, вигляд з точки зору проектування, вигляд з точки зору процесів, вигляд з точки зору реалізації, вигляд з точки зору розгортання [2, с. 48-49]. Кожне з цих явлень є однією з можливих проекцій структури системи та загострює увагу на деякому аспекті її функціонування.

Для візуалізації системи з різних точок зору, що допомагає вирішити проблему розбиття проекту реалізації її моделі на окремі структурні одиниці, створення інформаційної моделі системи, можна використовувати UML (Unified Modeling Language) методологію графічного моделювання на основі об'єктно-орієнтованого аналізу, офіційне створення якої почалося у 1994 році, а вже у 1997 році вона була прийнята за стандарт об'єктно-орієнтованого проектування.

Основу UML методології становлять діаграми. Діаграма – це одна з проекцій системи, що дає згорнуте представлення елементів, з яких складається система. Один і той же елемент може належати всім діаграмам, або тільки в кільком, або не належати жодній. Виділяються [2, с. 42-49] кілька типів діаграм: класів, об'єктів, прецедентів або випадків використання, функцій, по-

слідовностей, кооперацій, діяльності, станів, модулів тощо. При цьому не завжди всі можливі діаграми доцільно створювати при моделюванні системи.

Для автоматизації процесу ООА та проектування створюється спеціальне програмне забезпечення – CASE-системи. Існує кілька їх типів в залежності від функціональної орієнтації на ті чи інші процеси життєвого циклу програмного забезпечення (аналіз і проектування, розробка додатку (генерація програмного коду), документування, тестування, реінжиніринг (аналіз програмних кодів і схем баз даних, формування на їх основі різних моделей і проектних специфікацій) тощо).

Серед CASE-засобів зустрічаються такі, що реалізують тільки UML методологію (Rational Rose (Rational Software Corp.), ObjectiF (MicroTool), Visual UML (Visual Object Modelers)), а також такі, що дозволяють застосовують UML методологію в комбінації з іншими (Geode (Verilog), CASE/4/0 (MicroTool), Real (Ланит-Терком), ModelMaker (ModelMaker developers), WithClass99 (MicroGold Software)). Повніший комплекс діаграм UML методології надають CASE-системи: Real, Visual UML, Rational Rose, WithClass99.

Нажаль більшість випускників навіть технічних вузів мають тільки поверхнєве уявлення про зазначену методологію інформаційного моделювання. Надалі складність програмного забезпечення буде продовжувати зростати. Знання про методи і засоби проведення ООА та об'єктно-орієнтованого проектування дозволять майбутнім спеціалістам гідно розв'язувати поставлені задачі.

#### Література

1. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на С++, 2-е изд./ Пер. с англ. – М.: «Издательство Бинум», СПб.: «Невский диалект», 1999. – 560 с.
2. Буч Г., Рамбо Д., Джекобсон А. Язык UML. Руководство пользователя: Пер. с англ. – М.: ДМК, 2000. – 432 с.
3. Громов Г.Р. Очерки информационной технологии. – М.: Инфоарт, 1993.

## **ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ МУЛЬТИМЕДИЙНЫХ ТЕХНОЛОГИЙ В ПРЕПОДАВАНИИ ФУНДАМЕНТАЛЬНЫХ ДИСЦИПЛИН В ВЫСШЕЙ ШКОЛЕ**

Г.Ю. Маклаков

г. Севастополь, Севастопольский национальный технический  
университет

Традиционно основное внимание в преподавании фундаментальных дисциплин (особенно математики) уделяется формальной стороне предмета: знакомству с определениями основных понятий и логическому выводу их свойств, формальным правилам оперирования с понятиями и решению задач в рамках абстрактных моделей. В то же время психологи отмечают важность образной стороны мышления, поскольку многие результаты гораздо быстрее и легче усваиваются при оперировании образами и представлениями. Это относится и к достаточно абстрактным областям науки и техники, в том числе и к математике, и к конструкторской, и к проектной деятельности.

Рациональное использование компьютерных технологий в учебном процессе вуза создает условия для эффективного использования фундаментальных закономерностей мышления, оптимизации познавательного процесса. Связано это с тем, что становится возможным вносить в систему знаний существенно больше общих носителей информации, реализовать переработку информации параллельно на подсознательном и сознательном уровнях одновременно.

Одним из наиболее мощных факторов, позволяющим развить образную сторону мышления, является визуализация основных математических и физических понятий, процессов и явлений при помощи компьютера. Визуализация как элемент современных информационных технологий позволяет совершенно по-новому подойти к методике преподавания фундаментальных дисциплин в техническом вузе. Эффективное ее применение требует продуманной концепции, учитывающей специфику различных математических теорий, психологические закономерности формирования и развития пространственных представлений,

а также выбор адекватного языка визуализации и его программной реализации. Постигая учебный материал, студент осваивает информацию как явную, так и неявную, содержащуюся в тексте или в образах. Компьютерное построение фигуры, графика сопровождается подсознательным анализом этих рисунков, происходит активизация механизмов визуальной переработки информации. Полученные, с помощью компьютера: фигура, график, модель физического процесса – воспитывают не только логику, но и чувство красоты. Мультимедийные технологии целесообразно использовать в следующих направлениях: компьютерные демонстрации при изложении лекционного материала, при проведении практических занятий, при выполнении студентами домашних расчетно-графических заданий.

Акцент в использовании мультимедийных технологий целесообразно делать на когнитивную графику. Любая графика имеет два аспекта, или, точнее, имеется как бы две графики: иллюстративная и когнитивная. Отличительная особенность иллюстративной графики – фотореалистичность, узнаваемость. Когнитивная графика («cognition» – познание) изображает, визуализирует знание, активизирует процесс человеческого познания. Основоположителем идей и практических разработок компьютерной когнитивной графики (ККГ) является профессор А.А. Зенкин [5]. Еще в 80-х годах прошлого века он выдвинул идею информационной ККГ-технологии научного познания. Суть этой технологии можно хорошо передать словами Джима Кларка, одного из основателей компании Silicon Graphics, которые имеют прямое отношение именно к когнитивной графике. «Мы хотим, – говорил Кларк еще в 1981 году, – научиться обрабатывать образы и дать возможность человеку работать не на уровне цифр, букв и слов, а на уровне образов. Мы займемся задачами визуализации». Почему? Потому что, во-первых, отвечает Кларк, «97% образной информации, которую получает человек, – это визуальная информация», и, во-вторых, «без включения правого полушария мы не сможем разбудить интуицию, не сможем совершать новые открытия» [1]. По мнению Зенкина А.А., когнитивная графика изображает, визуализирует знание, которое еще никому (даже автору) не известно, и это новое знание нужно уметь увидеть, то есть открыть, понять и осмыслить. Идея ККГ –

превращения «незнаемого» в очевидное (очами видимое) знание. Суть этой идеи очень проста – нужно нарисовать суть интересующей вас предметной области, то есть как бы причину ее основных свойств. Тогда некоторые чисто графические фрагменты такой, как правило, абстрактной картины-орнамента начнут вам подсказывать такие следствия этой причины, о которых догадаться без когнитивной визуализации было просто невозможно.

Для практической реализации ККГ желательно использовать специально написанные для этой цели компьютерные программы, однако, имеющиеся пакеты математического проектирования (MathCAD, Maple, MatLab и др.) могут дать неплохие результаты. Как правило, все математические пакеты обладает мощными возможностями графической двух- и трехмерной визуализации вычислений и осуществлять анимацию полученных результатов решения. Особенно большими возможностями обладает пакет MatLab. Используя внутренний язык программирования, в полной мере удастся реализовать возможности ККГ-технологий.

Использование инструментальных математических сред создает принципиально новую основу для изучения фундаментальных дисциплин в ВУЗе [3]. Ключевой идеей нового подхода служит визуализация абстрактных понятий. В условиях учебного процесса появляется возможность работать с графическими объектами, исследование которых по своей сути близко к исследованию физических объектов и включает в себя такие этапы как наблюдение и накопление эмпирического материала, выдвижение и проверка гипотез, анализ и обоснование полученных выводов. ККГ-технологии позволяют интенсифицировать процесс обучения, но все же их главным достоинством является способность вызывать зрительные образы, ведущие к пониманию динамичного процесса или явления [2, 3, 5]. Такая возможность когнитивной графики особенно важна в настоящее время, когда в связи с ростом быстродействия компьютеров, складывается парадоксальная ситуация – исследователь может вычислить больше, чем понять. Трудности анализа огромного количества числовых данных, выдаваемых компьютерами, и стимулируют развитие ККГ-технологии, имеющую своей целью использовать уникальную человеческую способность видеть и понимать визу-

альные изображения для интерпретации больших объемов информации. ККГ является мощным инструментом для визуализации смысла математических абстракций.

В заключение хотелось бы подчеркнуть, что, несмотря на огромные возможности мультимедийных технологий, главную и определяющую роль в процессе обучения должен играть преподаватель, а использовать современные компьютерные методы активизации процесса обучения необходимо с учетом медико-психологических аспектов информационных технологий [4].

#### Литература

1. Карелов С. Храните информацию в правом полушарии. – «НГ-наука», 17 ноября 1999 г.
2. Маклаков Г.Ю. Когнитивное представление многомерных данных в мультимедийных обучающих системах. / В кн.: Теория и практика инженерного образования. Тезисы по материалам докладов VII Международной научно-методической конференции. 29 мая – 2 июня 2000 г.- Севастополь: Изд. СевГТУ, 2000. – С. 66-68.
3. Маклаков Г.Ю. Мультимедийная когнитивная система представления многомерных данных в системах компьютерного моделирования физических и технологических процессов. / В кн.: Комп'ютерне моделювання. Міждержавна науково-методична конференція. Тези доповідей. 29 червня – 1 липня 2000 р. Днепродзержинск: Изд-во ДДТУ, 2000. – С. 120-121.
4. Маклаков Г.Ю. Методологические принципы исследования влияния информационных технологий на психосоматические структуры человека. / В кн.: Труды Международных конференций «Искусственные интеллектуальные системы» (IEEE AIS-02) и «Интеллектуальные САПР» (CAD 2002). Научное издание. – М.: Изд-во физ-мат лит-ры, 2002. – С. 532-539.
5. Зенкин А.А. Когнитивная компьютерная графика. – М.: Наука, 1991. – 192 с.

## СОВРЕМЕННЫЕ КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ И ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ ЛИЧНОСТИ

Г.Ю. Маклаков

г. Севастополь, Севастопольский национальный технический  
университет

При внедрении современных компьютерных технологий в учебный процесс, как правило, рассматриваются только дидактические особенности обучающих систем. При этом опускается один очень важный аспект компьютерных технологий – медико-психологический. Речь идет, прежде всего, о влиянии современных информационных технологий на человека, точнее о воздействии информации на психические и соматические структуры человека.

В качестве примера рассмотрим медико-психологические аспекты дистанционного образования. Прежде всего, необходимо отметить, что сам принцип дистанционного образования предусматривает самое широкое использование глобальной сети Интернет. При этом возникает, по меньшей мере, две опасности. Одна из них – построение обучающих курсов дистанционного образования на основе современных компьютерных технологий (например, системы виртуальной реальности) способствует возникновению у человека информационных перегрузок, что, безусловно, сказывается на его психическом состоянии. Опасность во много раз возрастает при использовании технологий воздействия на подсознание (простейшая из них – «25<sup>ый</sup> кадр»). Вторая проблема, которая возникает при использовании дистанционного образования – использование Интернет. Следует отметить, что современное развитие информационных технологий дает возможность говорить о формировании в сети Интернет особого киберпространства (или «Интернет пространства»). Это пространство имеет четкую иерархию, для него свойственно наличие особого языка общения (жаргона, непонятным для непосвящённых), особого ритуала и символов, своеобразной письменности, искусства. В нем ведутся «войны» и совершаются преступления, имеются карательные и правоохранительные органы. Все это дает основания считать о существовании своеобразной суб-

культуры – «киберкультуры», которая формирует у «жителей» киберпространства характерные идеи, составляющие мировоззренческую позицию членов этого пространства, определяет этические нормы поведения. Любопытно отметить, что аналогичная субкультура возникла в среде криминального мира, в среде наркоманов. Под влиянием киберкультуры происходит изменение мировоззрения человек и в реальной жизни, могут измениться его духовно нравственные позиции и поведение в реальном обществе.

Критический анализ отечественных и зарубежных публикаций по проблеме информационной безопасности (включая обзор по сети Интернет Web-страниц ведущих зарубежных военных и гражданских научных исследовательских центров, лабораторий зарубежных высших учебных заведений), а так же опыт личных исследований по изучению влияния информационных технологий на психику и соматику человека позволяет сделать вывод о том, что современные информационных технологий могут представлять реальную угрозу для здоровья человека. Установлено, что интенсивное информационное воздействие вызывает измененные состояния сознания. Происходит изменение статуса личности (человек неадекватно оценивает себя и свои возможности) и статуса сознания. Измененное состояние сознания обуславливает трансформации восприятия (сдвиг порогов, синестезии), перемены эмоционального тона ощущений и структуры аффектов, памяти (спонтанное извлечение из памяти давно пережитых ситуаций, иногда переходящих в регрессию поведения), изменение восприятия течения времени (замедление, ускорение, раздробление). Тут важно подчеркнуть, что измененные состояния сознания стимулируют развитие регрессии поведения, которая трактуется психологами как специфическая форма ухода индивида от действительности, временный возврат его на более раннюю стадию развития, к более примитивным формам поведения и мышления.

При неконтролируемом использовании информационных технологий создаются весьма благоприятные условия для формирования особой психической зависимости. Эта зависимость по своему проявлению сходна с уже известными формами аддиктивного поведения (например, в результате употребления алко-



голя или наркотика). Такое одержимое поведение стало настоящей проблемой в некоторых студенческих городках развитых капиталистических стран, где персонал вынужден насильно отключать компьютеры у информационно зависимых студентов, использующих интернет-технологии более 18 часов в сутки (Sempsey, 1996). В этой связи примечательно высказывание профессора психологии Питсбургского университета Кимберли Янг: «Зависимость от Интернет можно сравнить с употреблением наркотиков. Это скорее всего напоминает патологический азарт – в этом состоянии человек теряет контроль над своими действиями». Показательно, что новый проект DSM-V (официальная классификация психических заболеваний США) включает главу «Кибернетические расстройства». Интересен и следующий факт. В зарубежной печати растет число сообщений относительно связанных с Интернетом смертей, например, в результате сердечного приступа из-за недостатка сна и нездорового образа жизни, сопутствующего интернет-зависимости; или в результате самоубийств, связанными с потрясениями испытанным в Сети (Almer, Dewitt, 1994; Rheingold, 1994). Преступления, совершаемые на почве наркотической зависимости от информационных технологий происходят в России, Украине и других государств СНГ.

Вышеприведенные факты ни в коем случае не свидетельствуют о необходимости запрета использования современных информационных технологий в образовании и ограничения развития дистанционного образования. Важно знать, что современные информационные технологии, безусловно, влияют на человека. Поэтому разработку систем дистанционного образования, в частности обучающих и контролирующих программ, важно вести не только с дидактических позиций, но и с учетом возможных психологических последствий информационных педагогических технологий. В настоящее время актуальным становится вопрос об ответственности создателей программного обеспечения для компьютеров за последствия их применения.

Сторона взаимодействия человека с компьютером – проблема психологических последствий информатизации заслуживает самого серьезного внимания.

## Литература

1. Контроль сознания и методы подавления личности: Хрестоматия / Сост. К.В.Сельченко. – Мн.: Харвест, М.: ООО «Издательство АСТ», 2001. – 624 с.
2. Маклаков Г.Ю. Метод об'єктивної оцінки вербального та музикального впливів на організм людини. // В кн.: S. Jarmus «On Our, Yet Not Our Own, Land...». Ukraine Through Canadian Ukrainian's Eyes. Our Culture and Science. «Tryzub». Winnipeg-Kyiv. 1999. P.42-46.
3. Маклаков Г.Ю. Метод оценки влияния информационных технологий на человека. // Збірник наукових праць за матеріалами 3 Міжнародної науково-методичної конференції «Освіта та віртуальність». 15-17 вересня 1999р. Міносвіти України, ХДТУР, СІАЕтаП. – Харків-Севастополь, 1999. – С. 56-59.
4. Маклаков Г.Ю. Метод исследования влияния информационных и коммуникативных технологий на человека. // «Научная сессия МИФИ-2001». VII Всероссийская научно-практическая конференция «Проблемы информационной безопасности в системе высшей школы». Сборник научн. тр. – М.: МИФИ, 2001. –С. 61-62.
5. Почепцов Г.Г. Информационные войны. – М: «Рефл-бук», К.: «Ваклер», 2000. – 576 с.
6. Прокофьев В.Ф. Тайное оружие информационной войны. – М.: СИНТЕГ, 1999. – 152 с.
7. Циганков В.Д., Лопатин В.Н. Психотронное оружие и безопасность России. Серия «Информатизация России на пороге XXI века». – М.:СИНТЕГ, 1999. – 152 с.

## РАЗРАБОТКА МУЛЬТИМЕДИЙНЫХ ПРИЛОЖЕНИЙ СРЕДСТВАМИ WEB-ТЕХНОЛОГИЙ

Г.Г. Маклакова

г. Севастополь, Севастопольский национальный технический  
университет

Разработку мультимедийных приложений с успехом можно осуществлять на основе электронных презентаций. При создании электронных презентаций чаще всего используются специальные программные комплексы (Freelance Graphics, Adobe Persuasion, Astound, PowerPoint и др.). Эти программные продукты хотя и обладают большими функциональными возможностями, но они ориентированы на представление презентаций в классическом виде. В них достаточно сложно, а порой просто невозможно использовать какие-либо специальные графические эффекты.

Вместе с тем разумное использование графических эффектов (например, морфинга) часто оживляет презентацию, а, следовательно, и мультимедийное приложение, делает ее привлекательной и убедительной. Следует отметить, что, как правило, для демонстрации презентации требуется установить в компьютере некоторую базовую программу. Существующие программные комплексы обладают широкими возможностями, но они, как правило, громоздки (занимают большой объем памяти) и достаточно дорогие.

Развитие сети Интернет способствовало возникновению и совершенствованию прикладных языков программирования, ориентированных на создание Web-узлов. Важно отметить, что большинство программ для создания страниц предусматривают не только использование гипертекста, но и широкое применение графики. В частности, в языке DHTML (Dynamic HyperText Markup Language) предусмотрены статические и динамические фильтры, которые с успехом можно использовать для преобразования изображений и создания разнообразных графических эффектов.

На основании проведенных исследований [1] статических и динамических фильтров, используемых в DHTML, были отобраны фильтры, которые можно использовать для создания мульти-

медийных приложений. Наиболее удобно использовать следующие статические фильтры: **blur** (размывает изображение на указанную глубину и в указанном направлении); **glow** (создает вокруг объектов светящийся ореол); **mask** (заливает все непрозрачные пиксели объекта белым цветом, а прозрачные объекты – заданным цветом фильтра); **wave** (добавляет «волновое» искажение к объекту). Гораздо больше возможностей для преобразования графических объектов дают динамические фильтры. Они позволяют задавать переход объекта из одного состояния в другое. Динамический фильтр **blendTrans** позволяет задать постепенное проявление или исчезновение объекта путем изменения его прозрачности. Параметры этого фильтра - *duration* (время в секундах, в течение которого переход осуществляется полностью) и *status* (этот параметр доступен только из сценариев и позволяет определить и изменить состояние перехода на текущий момент выполнения сценария). Возможные значения параметра *status*: 0 – переход остановлен (или еще не началось его выполнение), 1 – применение перехода завершено и 2 – переход в процессе применения. Значения параметров можно изменять и в процессе выполнения сценария с помощью методов перехода. Методов всего три: *Play* (начать выполнение перехода), *Apply* (временно приостановить выполнение перехода для переназначения параметров) и *Stop* (полностью остановить выполнение перехода). С помощью комбинаций методов можно создавать сложные динамические эффекты, полностью зависящие от конкретных действий пользователя. Особый интерес для создания графических эффектов представляет фильтр **revealTrans**. С его помощью можно менять один объект на другой, используя 24 варианта визуальных эффектов. Тип визуального эффекта задается параметром *transition*: замена изображения путем постепенно стягивания или растягивания геометрической фигуры (круг, прямоугольник); развертывание изображения в различных заданных направлениях; проявление в случайном или детерминированном (например, шахматном) порядке; эффект «вертикальных (или горизонтальных) жалюзи» и др.

Использовать фильтры для графических преобразований достаточно просто. В качестве примера можно привести программу (код сценария) постепенного преобразования одного

изображения (файл vas4.jpg) в другое (файл vas2.jpg):

```
<html>
<head><title>Переход revealTrans</title>
<script language="JavaScript">
    function transitionReveal() {
        img1.filters.revealTrans.Apply();
        img1.src = "vas2.jpg"
        img1.filters.revealTrans.Play();
    }
</script>
</head>
<body>
<DIV ALIGN="center">

<br><input type="button" value="Start"
onclick="transitionReveal(); ">
</body>
</html>
```

Использование фильтров DHTML позволяет достаточно просто и эффективно производить различные преобразования графических изображений и создавать эффектные мультимедийные приложения. Важно отметить, что при этом нет необходимости использовать дорогостоящее и громоздкое специальное программное обеспечение. В качестве недостатка данного подхода можно отметить, что рассмотренные фильтры поддерживаются только браузерами Internet Explorer 4.0 и выше. Однако, если учесть, что в настоящее время система Windows комплектуется браузерами Internet Explorer версии 6.0, то указанный недостаток вряд ли можно признать существенным.

#### Литература

1. Маклаков Г.Ю., Маклакова Г.Г. Исследование возможностей фильтров DHTML для создания специальных графических эффектов. В кн.: Графика XXI века. Сб. тез. докл. V Всеукр. студенч. научн. техн. конф. – Севастополь: Изд-во СевНТУ, 2002. С. 58 – 60.

## ПРОБЛЕМЫ ОБУЧЕНИЯ СОВРЕМЕННОМУ ПРОГРАММИРОВАНИЮ

В.Л. Малорян

г. Одесса, Южно-Украинский государственный педагогический университет им. К.Д. Ушинского  
mvadim@mail.od.ua

*Это будет личность, чувствующая себя в информационных потоках, как рыба в водных. Личность с новыми, небывалыми органами чувств – алгоритмической интуицией и бинарным мышлением, виртуальным видением и полифоническим восприятием. И предстоят ей великие свершения не на полях бранных или пахотных, а информационных и виртуальных, в мирах, доселе не знаемых.*

Многочисленные проблемы, с которыми сталкивается преподаватель программирования, особенно в вузе, имеют своей первопричиной сложность самого предмета изучения. Программирование относится к наиболее сложным видам человеческой деятельности [2, 12, 24]. Включая в себя все основные компоненты алгоритмической культуры [4–5], программирование тесно связывает их с *языком программирования*, что порождает дополнительный уровень сложности. С проблем языка мы и начнем.

Языки программирования – это формальные, искусственные системы, созданные для организации эффективного общения человека-программиста с компьютером-исполнителем. Интересно, что первоначально эти формальные языки не имели своего формального описания, доступного человеку. Фактически, язык определялся реализацией транслятора – переводчика этого языка в машинный код. Хотя с момента создания первого формально описанного языка Алгола-60 наличие описания становится скорее правилом, атавизмы влияния реализации сохранились до настоящего времени.

Язык программирования определяется своей *лексикой, синтаксисом, семантикой, прагматикой*. Все эти элементы языка подлежат изучению, в той или иной мере, в любом курсе программирования. Однако часто внимание преподавателя сосредото-

тацивается на первых двух элементах языка, более простых и более строго описанных. Конечно, владение лексикой и синтаксисом необходимо, но не в ущерб более сложным и важным элементам – *семантике*, придающей смысл предложениям формального языка [11], и *прагматике*, предлагающей оптимальные пути реализации типовых задач средствами данного языка. Трудность состоит в том, что эти 4 элемента нельзя полностью разделить. Поэтому серьезные курсы программирования дают необходимые сведения постепенно и параллельно. Чем в большей степени разделимы элементы языка, тем легче задача его изучения и преподавания. Однако при этом мы не должны упускать из виду, что установление связей между элементами жизненно необходимо для формирования *понимания* языка. Не только отдельных его слов и предложений, но также *духа* и *философии языка*.

Задержим внимание на одной из малозаметных проблем *лексики* языка. Должна ли она походить на лексику какого-либо естественного языка, в частности, родного языка программиста? Кажется очевидным положительный ответ на обе части этого вопроса. Однако не будем торопиться. Из практической лингвистики известен психологический эффект трудности изучения и использования близкородственного языка. Не ходя за далекими примерами, вспомним хотя бы многочисленные русизмы в речи ряда отечественных общественных деятелей. Это систематические, повторяющиеся ошибки. Их причина – в несоответствии всей семантической структуры близких, но различных языков, неадекватности смысловой нагрузки одинаковых или подобных лексем и выражений. В не меньшей степени это верно и для лексически близких формального и естественного языков. Логично поэтому думать, что лексика искусственного языка должна существенно отличаться от лексики естественных языков, по крайней мере, родного языка программиста. В то же время, действующая формальная лексика – «англоязычна». Не в этом ли одна из причин заметного превосходства в программировании представителей наций, привычных к кириллическому или иероглифическому письму?

Не знаю, как читателю, а автору уже давно хочется иметь под рукой язык программирования с иероглифической лексикой

(1 иероглиф = 1 лексема), по крайней мере, для стандартных лексем языка. Поэтому дважды нами ставилась задача построения подобного языка перед дипломниками, но с относительным успехом – задача оказалась достаточно трудной. Поэтому же хочется всячески поддержать опубликованные работы в близком направлении [13, 18].

Другой круг проблем связан с *развитием* каждого языка программирования в отдельности и всего семейства формальных языков в целом. Хотя, в силу теоремы Бома-Якопини, для реализации любого алгоритма достаточно трех управляющих структур (следования, выбора и повторения), в настоящее время насчитывается порядка 500 различных языков и диалектов программирования. Таким многообразием трудно оперировать, поэтому используются разнообразные принципы классификации по *уровню, парадигме, специализации*. Некоторые языки прошли несколько ступеней развития, меняя по пути названия и образуя языковые линии. Проблемой является само одновременное сосуществование большого количества разнотипных языков с различающимися целями, философиями, методологией применения. Это порождает и различные подходы к программированию и обучению, различные цели и методы обучения. Хотя опыт показывает, что труднее всего для изучения первый язык, это верно лишь для языков одной группы. За 50 лет интенсивного развития процесс программирования менялся от *написания* через *строительство* к *сборке* программ из готовых компонентов [2, 14, 17]. При этом ключевым мотивом развития является *повторная используемость*. Сегодня видны также тенденции дальнейшего развития в сторону *настройки, воспитания* и даже *выращивания* программ. Последние тенденции связаны, конечно, с макропрограммированием, нейронными сетями и генетическим программированием.

Это кардинальным образом меняло и меняет процесс обучения программированию. Трудность же преподавания современного программирования порождается необходимостью поддержания преемственности языков с подчас плохо совместимыми философиями.

Итак, проблема выбора языка, в частности, первого языка [15] для обучения программированию, весьма нетривиальна. И серьезное влияние на ее решение сегодня оказывают отнюдь не



методические аспекты, а вопросы практической применимости языка, его распространенности и т.п. В чем причина перехода большинства американских университетов в начале 90-х годов с Паскаля на C/C++? Да в этом же [8]. Все более сильное давление испытывают преподаватели программирования и в учебных заведениях Украины, даже в школах. И это происходит несмотря на очевидную сложность C++ [20]. Дело, по-видимому, в том, что линия Паскаль/Дельфи/Киликс сегодня не вполне отвечает запросам практики. Да и в теоретическом плане она является искривлением генеральной линии Вирта [3, 4] (Паскаль/Модула/Оберон).

Нельзя игнорировать и факт взаимозависимости языка программирования и операционной системы, подчеркиваемый наличием действующих международных стандартов программирования открытых систем. Мечта о всеобщей и полной переносимости языков и программ, все еще остается мечтой, несмотря на сверхусилия разработчиков многочисленных стандартов, в частности – авторов Джавы. Все же, для Юникс/Линукс платформ родным является язык C. Здесь он обеспечивает и портабельность, и эффективность. Все большее значение приобретают системы поддержки языков времени выполнения. Это особенность Джавы, хорошо известная и применявшаяся и до нее, и после нее – например, в языках платформы NET. Однако Microsoft, как всегда оригинальна: вместо переносимости между платформами она пытается все языки привести к одной, своей, платформе, где Бейсик, C++ и C# все более походят на один язык.

Наконец, следует отметить принципиальную важность возможности обобщенного программирования в C++ за счет использования шаблонов. Хорошая документированность и наличие качественных и разных учебников [8] также немаловажна.

*"Процесс составления программ для цифровой вычислительной машины особенно привлекателен тем, что он не только дает экономические и научные результаты, но и доставляет эстетические переживания, во многом близкие переживаниям, испытываемым при сочинении стихов или музыки".*  
Д. Кнут [12, с. 8]

Программирование – это не только собственно составление программ. Как вид человеческой деятельности, оно включает: *обучение, проектирование, составление (кодирование, написание), понимание (чтение), тестирование, отладку, документирование, модификацию (сопровождение)*. Эти виды деятельности имеют различные психологические особенности и нуждаются в специальном изучении. К сожалению, отечественная психологическая наука уделяет мало внимания *психологии программирования*, а выводы зарубежной [24] – недостаточно учитываются у нас на практике. Эффективность деятельности программиста сильно зависит от окружающей обстановки – среды, в которой она осуществляется. В малом, эта среда включает набор инструментальных средств, составляющих *систему программирования* и тесно взаимодействующих с операционной системой. В большом, – также и производственные условия, моральный микроклимат и т.п. Далее, на основании синтактико-семантической модели психологи считают, что в процессе обучения программированию в мозгу формируются многоуровневые *семантические мыслительные структуры*, не связанные с определенным языком [3, 24]. Именно поэтому обычно труднее изучить первый язык программирования: приходится сочетать изучение синтаксиса с формированием мыслительных структур. Без этого невозможно добиться понимания, хотя, как и в математике, необходим известный компромисс между пониманием и результатом. Уникальной возможностью для изучения семантики языка программирования является наличие двух абсолютно разнородных групп носителей языка: неформальных – программистов, и формальных – систем программирования. Одни вопросы программирования оказывается легче усвоить, общаясь с людьми, другие – с компьютером.

С точки зрения психологии, составление программ есть многоэтапный процесс, во время которого от формулировки задачи путем пошагового уточнения либо переформулирования программист приходит к решению задачи в виде внутренних семантических структур, а затем и к решению на заданном языке программирования. Процесс понимания программы есть ее перевод во внутреннюю семантическую структуру. Обучение программированию есть непрерывный процесс, внутренне присущий са-

мой профессии программиста.

Психология указывает также на некоторые характеристики личности, способствующие успешному занятию программированием. Среди них – настойчивость, самоконтроль, умеренная возбудимость, высокая мотивация, склонность к некоторому риску и работе в условиях неопределенности, точность, скромность, стрессоустойчивость [24]. Правда, число исследований в этой области – невелико, а их результаты почти не учитываются преподавателями, разве что на подсознательном уровне.

Важной задачей обучения программированию является формирование если не стиля мышления (хотя и его тоже), то *стиля* программирования. Хороший стиль программирования предусматривает наличие комментариев, осмысленные имена, оформление операторных структур сдвигами текста, модульность [4, 6]. Важность этих элементов стиля подтверждают внутренние стандарты организаций по программированию. Следует отметить, что как в любом деле, стилизуя программу, следует избегать чрезмерности. Так, комментарии должны быть эвристичными, обозначающими смысл каждого модуля и блока, но не чрезмерно подробными; сдвиги не слишком далеко уводящими от левого поля и т. п. Менее очевидны и известны результаты психологических экспериментов, из которых следует, что для понимаемости программ достаточно либо осмысленных имен, либо комментариев [24].

Не вполне соответствуют ожиданиям многих преподавателей и результаты экспериментов по влиянию предварительного составления блок-схем на качество получаемой программы [24]. Подробная блок-схема ожидаемого эффекта не оказывает. В крайнем случае, можно рекомендовать составление высокоуровневой общей блок-схемы. Подробная же блок-схема является просто эквивалентной записью на графическом языке того же уровня. Если она и обязательна при оформлении большинства проектов, то лишь поскольку более удобна для восприятия руководством, отвечая психологическим особенностям личности типичного руководителя. Типичный программист оформит блок-схему после того, как программа полностью отлажена. Сказанное касается традиционных плоских блок-схем. Появление ООП и UML существенно изменило роль проектирования и от-

крыло поле для более широкого использования графического унифицированного языка моделирования (UML) [22, 25]. Однако ему только начинают учить в отечественных вузах.

*"Занятие программированием отвечает глубокой внутренней потребности в творчестве и удовлетворяет чувственные потребности, которые есть у всех нас, доставляя пять видов радости:*

*-Радость, получаемая при создании чего-либо своими руками.*

*-Удовольствие создавать вещи, которые могут быть полезны другим людям.*

*-Очарование создания сложных головоломных объектов, состоящих из взаимодействующих движущихся частей.*

*-Радость, получаемая от неизменного узнавания нового, неповторяемости задачи.*

*-Удовольствие от работы со столь податливым материалом – чистой мыслью, который, тем не менее, существует, движется и работает так, как не могут словесные объекты".*

Ф. Брукс [2, с.212]

Обратимся теперь к мнению и выводам практиков и организаторов крупных программных проектов. Это удивительно и неожиданно слышать от них оценку программистского труда как творчества, притом творчества в рамках коллектива и строгой дисциплины. Ниже приведены еще несколько цитат из [2], которые представляются нам весьма правильными и полезными.

*"Дисциплина полезна искусству. Получение архитектуры извне усиливает, а не подавляет творческую активность группы исполнителей". (Сравните также с [7]).*

*"При изучении программирования труднее всего привыкнуть к требованию совершенства".*

*"...лучшие программисты-профессионалы в 10 раз продуктивнее слабых при равной подготовке..."*

*"...фактическая власть приобретается как следствие*

*успешного выполнения задач".*

*"Концептуальная целостность является наиболее важным соображением при проектировании систем".*

Современное программирование является уникальным примером широкомасштабного коллективного сотворчества программистов. Кроме того, это первый и пока единственный пример взаимодействия человеческого интеллекта с интеллектом иной, небологической, природы. Сколь поучительны могут быть уроки такого творческого взаимодействия. Практические рекомендации по оценке производительности труда и оценке качества программ могли бы быть использованы и для оценки учебных программ с необходимой поправкой в виде некоторого коэффициента. Рекомендации по организации бригад программистов могли бы быть полезны при выдаче курсовых и дипломных проектов практической направленности. Давно известны также программные метрики Боэма-Брауна-Лайпоу, Джилба, Холстеда [23, 24]. Жаль, что все это почти никем не используется при обучении программированию.

В данной статье отражены результаты осмысления автором некоторых необычных аспектов, обнаруженных им как в собственном 25-летнем опыте программирования и преподавания, так и, главным образом, во многих выдающихся работах, ставших классикой программирования во всех его ипостасях: искусства, науки и технологии [1-9, 11-12, 14, 19, 21, 24-25].

#### Литература:

1. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. Пер. с англ. – М.: Мир, 1979. – 536 с.
2. Брукс Ф. Мифический человеко-месяц или как создаются программные системы. – Пер. с англ. – СПб.: Символ-Плюс, 2001. – 304 с.
3. Вирт Н. Алгоритмы + Структуры данных = Программы. Пер. с англ. – М.: Мир, 1985. – 406 с.
4. Вирт Н. Систематическое программирование. Введение. Пер. с англ. – М.: Мир, 1977. – 184 с.

5. Грис Д. Наука программирования. – М.: Мир, 1984. – 416 с.
6. Дал У., Дейкстра Э., Хоор К. Структурное программирование. Пер. с англ. – М.: Мир, 1975. – 248 с.
7. Дейкстра Э. Дисциплина программирования. Пер. с англ. – М.: Мир, 1978. – 275 с.
8. Дейтел Х.М., Дейтел П.Дж. Как программировать на C++: Третье издание. Пер. с англ. – М.: ЗАО "Издательство БИНОМ", 2001. – 1152 с.
9. Ершов А.П. Введение в теоретическое программирование (беседы о методе). – М.: Наука, 1977. – 288 с.
10. Жалдак М.І. Проблеми інформатизації навчального процесу в школі і в вузі // Сучасна інформаційна технологія в навчальному процесі: Зб. наук. праць/ Редкол.: М.І. Шкіль (відп. ред.) та ін. – К.: КДПІ, 1991. – С. 3-16.
11. Калинин А.Г., Мацкевич И.В. Универсальные языки программирования. Семантический подход. – М.: Радио и связь, 1991. – 400 с.
12. Кнут Д. Искусство программирования для ЭВМ. – т.1. Основные алгоритмы. – М.: Мир, 1976. – 736 с.
13. Копаев О.В. Вплив сучасних інформаційних технологій на вивчення основ алгоритмізації в середній школі // Комп'ютер у школі та сім'ї. – 2000. – №2. – С. 24–27.
14. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. – М.: МЦНМО, 2000. – 960 с.
15. Малорян В.Л. К вопросу о выборе первого языка программирования // Теорія та методика навчання математики, фізики, інформатики: Зб. наук. праць. – Кривий Ріг: КДПУ, 2001. – т. 3. – С. 116-118.
16. 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.
17. Малорян В.Л., Варбанец С.В. Компонентно-ориентированный подход к изучению курса программирования в высших учебных заведениях // Зб. статей Другої всеукраїнської конференції молодих науковців «Комп'ютерне моделювання та

- інформаційні технології в природничих науках» м. Кривий Ріг, 2-5 травня 2000 р.
18. Миленький О.В. Система "Алгоритм" – крок від абстрактного поняття до потужного інструмента // Комп'ютер у школі та сім'ї. – 2001. – №2. – С. 12–15.
  19. Простое и сложное в программировании / Авт. предисл. Е. П. Велихов. – М.: Наука, 1988. – 176 с.
  20. Свердлов С. Арифметика синтаксиса // "PCWeek/RE". – 1998. – №42-43.
  21. Уззерелл Ч. Этюды для программистов. Пер. с англ. – М.: Мир, 1982. – 288 с.
  22. Фаулер М., Скотт К. UML в кратком изложении. Применение стандартного языка объектного моделирования: Пер.с англ. – М.: Мир, 1999.
  23. Холстед М.Х. Начала науки о программах. – М.: Финансы и статистика, 1981. – 128 с.
  24. Шнейдерман Б. Психология программирования: Человеческие факторы в вычислительных и информационных системах. Пер. с англ. – М.: Радио и связь, 1984. – 304 с.
  25. Booch G. Object solutions. Managing the Object-Oriented Project.- Menlo Park: Addison-Wesley, 1996. – 323 p.

## ОСОБЛИВОСТІ ВИКЛАДАННЯ КУРСУ КОМП'ЮТЕРНОЇ ГРАФІКИ

І.В. Настенко

м. Кривий Ріг, Криворізький державний педагогічний університет

Поширене використання комп'ютерної графіки в різних галузях науки і техніки, як-то побудова графіків функцій, діаграм, схем, карт, імітація тривимірної реальності (зображення сцен в комп'ютерних іграх, тренажерах, системах архітектурного проектування) тощо, вимагає від введеного курсу комп'ютерної графіки в вузі докладнішого і ширшого розглядання.

Складність розуміння студентами алгоритмів і методів комп'ютерної графіки залежить від наявного багажу знань з геометрії (афінні перетворення, види проєкцій, вектори, криві, поверхні), інформатики (вмінь і навичок програмування, знань про функціонування вхідних і вихідних графічних пристроїв) та фізики (відображення, переломлення світла).

Доцільно при викладанні даного курсу приділити більше уваги алгоритмам і методам побудови зображень, оскільки вони є в графічних бібліотеках.

Основні графічні пристрої є растровими. Вони подають зображення у вигляді прямокутної матриці пікселів, тобто растра. Сформувані растрові зображення можна по-різному. Наприклад, можна складати зображення об'єкту методом послідовної побудови окремих простих елементів, які підтримуються більшістю графічних пристроїв виведення. Будь-яке растрове зображення можна побудувати по пікселям, але цей метод складний і повільний. Потрібно мати більш складні елементи, в яких відображується одразу декілька пікселів. При вирішенні цієї задачі використовується велика кількість алгоритмів. Наприклад, при використанні інкрементних алгоритмів растеризації прямих і кривих ліній, досягається значне підвищення швидкодії для підрахунків кожного пікселя в порівнянні з прямим способом. Взагалі, вирішення питання про швидкодію при побудові графічних зображень є досить актуальним.

При використанні алгоритмів виведення фігур на екран по-



трібно знайти “найшвидший” алгоритм побудови фігур.

В залежності від поставленої задачі можна користуватись алгоритмами заповнення, які використовують математичний опис контуру (не передбачається попереднє створення пікселів контуру растра – контур взагалі може не виводитись) або алгоритмами зафарбовування від внутрішньої точки до границь довільного контуру.

Особливу увагу при викладанні дисципліни слід звернути на методи і алгоритми тривимірної графіки. Для опису форми поверхні можуть використовуватися різноманітні методи (аналітична модель, векторна полігональна модель, воксельна модель, рівномірна сітка, нерівномірна сітка). Необхідно підкреслити позитивні риси і недоліки кожної моделі. Наприклад, говорячи про векторну полігональну модель, слід відзначити серед її переваг:

- зручність масштабування об'єктів;
- невеликий обсяг даних для опису простих поверхонь, що адекватно апроксимуються плоскими гранями;
- необхідність обчислювати тільки координати вершин при перетвореннях систем координат чи переміщенні об'єктів;
- апаратна підтримка багатьох операцій у сучасних графічних відеосистемах.

Серед недоліків можна відзначити складність алгоритмів візуалізації для створення реалістичних зображень, складні алгоритми виконання топологічних операцій (наприклад, розрізів) [1].

При зафарбовуванні поверхонь важливими є такі поняття як відображення і переломлення світла, окрему увагу слід приділити моделям відображення світла. Для зафарбовування гладких криволінійних поверхонь, які описані у вигляді многогранників або полігональної сітки з плоскими гранями, призначені методи Гуро і Фонга. Цінність цих методів полягає в створенні ілюзії гладкості. Ці методи потребують чіткого і структурованого пояснення.

Розглядаючи вище згадані методи і алгоритми комп'ютерної графіки, спрощується розуміння функціонування графічних бібліотек. Тому потрібно сформулювати у студентів розуміння того, що знання алгоритмів і методів комп'ютерної графіки дозволять їм краще розуміти і використовувати можливості стандартних

графічних бібліотек, наприклад, OpenGL.

Роботу на практичних заняттях пропонуємо виконувати в такому порядку:

- поставити задачу;
- надати методичні вказівки до виконання завдання ;
- вказати, які алгоритми і методи використати в даному завданні;
- коротко подати основні теоретичні відомості;
- представити алгоритм вирішення даної задачі;
- продемонструвати програму і результат її виконання

(рис. 1);

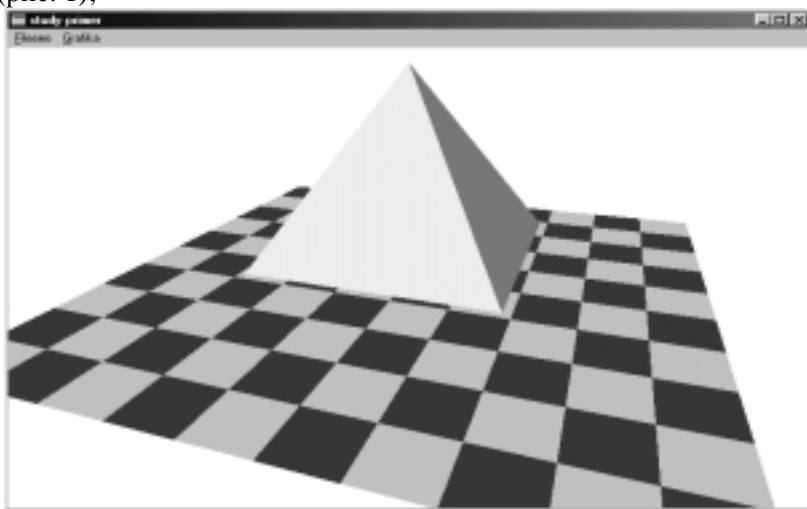


Рис. 1

– вказати, що не всі можливості OpenGL використані і запропонувати змінити програму, результат якої повинен бути схожим до попереднього, але зробити, наприклад, об'єкти освітленими.

Програма для побудови графічних зображень, як кінцевий продукт, з одного боку, повинна втілювати деякі узагальнені методи і алгоритми, а з іншого – узгоджувати ці методи і алгоритми з програмними та апаратними можливостями комп'ютера. При цьому важливим є знання особливостей і використання можливостей операційної системи. Це дозволить автоматично адаптувати програми до можливих змін системи.

В роботі передбачені комплект завдань та методичних рекомендацій на їх виконання, які забезпечують курс комп'ютерної графіки згідно робочої програми педагогічного університету.

Література:

1. Порев В.Н. Компьютерная графика. – СПб.: БХВ-Петербург, 2002. – 432 с.: ил.
2. Шикин Е.В., Боресков А.В. Компьютерная графика. Полигональные модели. – М.: ДИАЛОГ-МИФИ, 2001. – 464 с.
3. Роджерс Д. Алгоритмические основы машинной графики. – М.: Мир, 1989. – 512 с.
4. Эгрон Ж. Синтез изображений. Базовые алгоритмы. – М.: Радио и связь, 1993.

## **ФОРМУВАННЯ ДЕЯКИХ ПРОФЕСІЙНИХ УМІНЬ ФАХІВЦЯ З ВИЩОЮ ТЕХНІЧНОЮ ОСВІТОЮ НА ЗАНЯТТЯХ З ІНФОРМАТИКИ**

О.Г. Онишко

м. Хмельницький, Технологічний університет Поділля

Особливістю навчання інформатиці у технічному ВНЗ є наявність можливості у формуванні таких професійних якостей, як спрямованість на удосконалення творчих методів діяльності, неперервної самоосвіти в галузі своєї професії.

Щоб сформувати такі якості під час навчання інформатиці, необхідно певним чином підібрати зміст, методи навчання, систему педагогічних впливів.

Численними дослідженнями встановлено, що ефективність і результативність процесу самоосвіти особистості суттєво залежать від рівня розвитку спеціальних особистісних якостей, від засвоєння знань, умінь здійснювати самоосвіту. Тому в педагогіці поставлені і вирішуються питання ефективності навчання, цілеспрямованого формування готовності до самовиховання і самоосвіти (Ю.К. Бабанський, А.К. Громцева, В.С. Ільїн, Н.А. Половнікова, Л.І. Рувинський, А.В. Усова, Т.І. Шамова і ін.).

Розглядаючи готовність особистості до самоосвіти як синтез внутрішніх умов, які визначають принципову здійсненність процесу самоосвіти, можна виділити такі складові[1,2,3]:

- цілісний емоційно-особистісний апарат (внутрішня потреба у самовдосконаленні, особистісні цінності, емоційно-вольовий механізм, загальні розумові здібності і т.п.);
- система знань, умінь, навичок самоосвіти, засвоєна особистістю (повнота і глибина сформованості наукових понять, взаємозв'язків між ними, вміння співвідносити наукові поняття з об'єктивною реальністю, розуміння відносності знань і необхідності уточнення їх шляхом систематичного пізнання і т.п.);
- вміння і навички грамотно працювати з головними джерелами соціальної інформації, такими, як книги, бібліографічні системи, автоматизовані інформаційно-пошукові засоби тощо (вміння орієнтуватися у великих об'ємах інформації, вибирати головне, зафіксувати його і т.п.);

- система організаційно-управлінських умінь і навичок (визначати і виконувати завдання самоосвіти, планувати свою працю, вміло розподіляючи зусилля і час на різні обов'язки, створювати сприятливі умови для самодіяльності, здійснювати самоконтроль, самоаналіз результатів і характеру самодіяльності).

Приклад завдання, в процесі розв'язання яких викладач формує систему знань, умінь, навичок самоосвіти.

Завдання на атестацію (спец. МО)

1) Задано матрицю  $A$  ( $n \times n$ ), де  $n=3/$

- a) знайти визначник матриці;
- b) знайти обернену матрицю;
- c) знайти транспоновану матрицю;

2) Протабулювати функцію  $y=f(x)$  на інтервалі  $[a, b]$  з кроком  $h$ . Приклади даних наведено. Використовуючи знайдені значення, побудувати графік залежності  $y=f(x)$  за допомогою засобу

**Діаграми**

Номер варіанта	$f(x)$	$[a, b]$	$h$
1.	$y=\ln(2 +  \cos x )$	$[0; 1]$	0,1
2.	$y=\frac{x^2-1}{\sqrt[3]{x+1}}$	$[0; 1]$	0,1
3.	$y=\sin^2 x + \sqrt[3]{x}$	$[0; 1]$	0,1

3) Знайти корені заданого квадратного рівняння:

$x^2+N \cdot x-(N+1)=0$ , де  $N$  – номер варіанту.

7. Запустити програму «**Проводник**». Знайти свій файл і запустити його.

8. Мінімізувати Word. Привести Word в повноекранний режим.

9. Записати файл «Таня» під іменем «копія» в каталог Мельник. Закрити Word.

10. Відкрити у Word файл Таня. Запустити Word. У ньому відкрити файл «копія». Перевести обидві програми Word у режим вікон.

11. Змінити масштаб виведення обох файлів на «Страница целиком», при цьому має бути включена командою «**Разметка страницы**» в пункті «Вид».

Головною умовою, яка спонукає студентів до самоосвіти, творчою пошуку, є, на наш погляд, характер взаємодії викладача і студентів. Вважаючи, що така взаємодія повинна мати всі ознаки системи і максимально сприяти розвитку потреби особистості в самоосвіті. Педагогічний вплив на особистість студентів здійснюється через систему взаємовідносин викладача і студентів. Це дозволяє не тільки активізувати суб'єктну діяльність студентів, але й сприяє її вдосконаленню. Студенти отримують можливість через діяльність в межах системної взаємодії з викладачами виявляти самоврядування і покращувати якість. В основу планування і реалізації конкретних видів взаємодії між викладачами і студентами повинні бути покладені фактичний рівень готовності особистості до самоосвіти і завдання із “зони найближчого розвитку”, вирішення яких сприяє підвищенню цього рівня.

Особливості системи взаємовідносин викладача і студентів полягають в тому, що необхідно створювати умови, які максимально сприяють формуванню у студентів організаційно-управлінських умінь на основі заохочення їх суб'єктної діяльності. Для цього необхідно постійно ставити перед студентами нові завдання, вирішення яких вимагає від них нестандартних дій, самостійного пошуку раціональних навчальних процедур, умінь здійснювати самоуправління навчанням.

Виділимо основні види взаємодії між викладачами і студентами, які визначаються готовністю студента до самоосвіти і завданнями виховання, освіти, розвитку, професійної підготовки.

*Перший* вид характеризується інструктуванням викладачем студентів. Викладач повністю пояснює на міжособистісній, людино-машинній або іншій основі весь комплекс процедур, який повинен виконати кожний студент при вирішенні поставлених завдань. Діяльність останніх носить при цьому репродуктивний характер. Але як суб'єкти самоосвіти вони не пасивні. Від їх емоційно-вольових зусиль залежить старанність і ретельність при виконанні інструкцій викладача. Особливо помітна суб'єктна роль студентів при самостійній роботі. Здійснюється цей вид управлінської взаємодії шляхом використання рецептивно-репродуктивних методів виховання і навчання [4].

*Другий* вид характеризується тим, що викладач проводить альтернативне інструктування, а вибір найбільш раціональної в

конкретних умовах альтернативи здійснюється студентами самостійно. Це означає, що викладач ставить перед ними завдання, інструктує про можливі шляхи вирішення, а вихованці з власної ініціативи обирають один із вказаних шляхів вирішення поставлених завдань. Як суб'єкт навчання кожний студент повинен вибрати ту інструкцію, яку вважає вартою для якісного вирішення поставленого завдання. Його суб'єктна діяльність при цьому виді взаємодії все ще репродуктивна, але вже присутні елементи продуктивної діяльності, пов'язані з доцільним вибором раціональної альтернативи. Досягається цей вид взаємодії шляхом застосування рецептивно-репродуктивних і продуктивних методів виховання і навчання.

*Третій* вид характеризується тим, що викладач здійснює постановку різноманітних завдань, проводить оглядове інструктування можливих шляхів їх вирішення, враховуючи рівень готовності студентів до самоосвіти. Студенти ж як суб'єкти вибирають завдання із запропонованих викладачем, а серед відомих їм інструкцій вибирають ту, яка, на їх думку, найбільш сприятиме вирішенню поставленого завдання. При цьому не виключається можливість конструювання вихованцями за допомогою відомих їм інструкцій, нової, придатної для вирішення вибраного завдання. Цей вид управлінської взаємодії вимагає продуктивної діяльності з боку студентів. Здійснюється даний вид взаємодії застосуванням творчих, евристичних методів виховання і навчання.

*Четвертий* вид відрізняється тим, що викладач здійснює лише постановку сукупності завдань і консультує студентів. На перший план навмисно висунена суб'єктна функція студентів, яким належить самостійно визначити способи виконання отриманих завдань, питань організації процесу виконання і т.п. У викладача знаходяться всі засоби контролю і право спрямовувати суб'єктну діяльність студентів в доцільне русло. Здійснюється цей вид взаємодії шляхом застосування продуктивних, евристично-пошукових та інших активних методів [5].

Так, в процесі виконання наступного завдання студенти самостійно визначають способи виконання, питань організації процесу виконання.

### **Створення платіжної відомості.**

**Завдання.** Створити за допомогою програми Excel платіжну

відомість, що має таку структуру:

- Заголовок: Платіжна відомість 1
- Стовпець А: № п/п
- Стовпець В: Прізвище та ініціали
- Стовпець С: Посада
- Стовпець D: Рік прийому на роботу
- Стовпець Е: Оклад
- Стовпець F: Стаж роботи
- Стовпець G: Додаткові виплати
- Стовпець H: Сумарний дохід
- Стовпець I: Податок
- Стовпець J: Залишок
- Стовпець K: Аванс
- Стовпець L: На руки

Платіжна відомість повинна містити дані про 14 співробітників малого підприємства “Контакт” (табл.).

1. Користуючись клавишою на панелі Microsoft Excel Автосума обчислити суми стовпців E, G – L. Суми помістити у клітини E20, G20 – L20 відповідно.

2. Вирівняти назви стовпців і числові дані по центру стовпців.

3. Виділити “шапку” таблиці і стовпці шляхом окреслення їх границь.

4. Виділити суми у рядку <Всього> синім кольором.

5. Записати створену електронну таблицю на магнітний диск.

*П'ятий* вид характеризується переважно самостійною діяльністю студентів. Вони ставлять перед собою окремі задачі, вирішення яких пов'язане з виконанням в цілому завдання викладача, шукають засоби, форми і методи їх раціональних рішень. Викладач методами самостійної роботи спонукає студентів до самоосвіти, максимально сприяє цьому, наприклад, студенти самостійно здійснюють пошук функцій побудови графіку звіту вирішення попередньої задачі (рис. 1).

При цьому особистісно-діяльпісний підхід, заснований на технології індивідуальної підтримки, що здійснюється через ситуації успіху, створює умови для навчання студентів технологічній культурі праці.



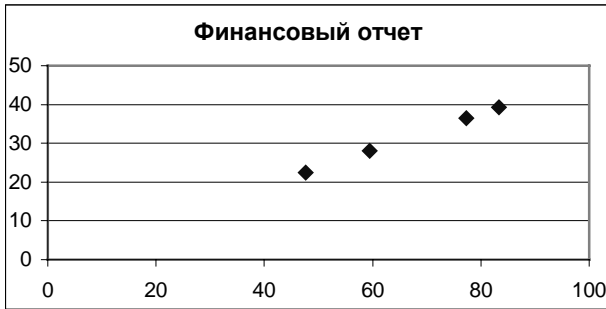


Рис. 1.

А діяльнісно-творчий компонент даного підходу сприяє формуванню і розвитку у студентів різноманітних способів діяльності, творчих здібностей, необхідних для самореалізації особистості. Особистісний компонент забезпечує самопізнання, розвиток рефлексивної здатності, оволодіння способами саморегуляції, самовдосконалення, морального самовизначення, формує життєву позицію.

#### Література:

1. Нечаев Н.Н. Психолого-педагогические аспекты подготовки специалистов в вузе. – М.: Изд-во МГУ, 1985. – 113 с.
2. Рубинштейн С.Л. Проблемы общей психологии / Отв. ред. Е.В. Шорохова. Изд. 2-е. – М.: Педагогика, 1976. – 416 с.
3. Давыдов В.В. Теория развивающего обучения. – М.: Интор, 1996. – 544 с.
4. Клочко В.І. Нові інформаційні технології навчання математики в технічній вищій школі. Автореф. дис...д-ра пед. наук. – К., 1998. – 36 с.
5. Основи нових інформаційних технологій навчання: Посібник для вчителів / Авт. кол.; За ред. Ю.І. Машбиця / Інститут психології ім. Г.С. Костюка АПН України. – К.: ІЗМН, 1997. – 264 с.

# РОЗВИТОК МИСЛЕННЯ СТУДЕНТІВ ПІД ЧАС НАВЧАННЯ ІНФОРМАТИЦІ У ВИЩОМУ ТЕХНІЧНОМУ НАВЧАЛЬНОМУ ЗАКЛАДІ ЯК КОМПОНЕНТ ЇХНЬОЇ ГОТОВНОСТІ ДО МАЙБУТНЬОЇ ПРОФЕСІЙНОЇ ДІЯЛЬНОСТІ

О.Г. Онишко

м. Хмельницький, Технологічний університет Поділля

Одним із важливих компонентів готовності фахівця до праці є його творчі, інтелектуальні можливості, які в підсумку визначають відношення до праці, його пізнавальні потреби, моральні установки, професійні орієнтири. Під час фахової діяльності зростає міра соціальної свободи для активної, творчої особистості і відроджуються орієнтації на ціннісно-значеннєве розкриття індивідуальності.

Для того, щоб процес навчання інформатиці виконував одночасно функцію впливу на розвиток мислення, творчих умінь, необхідних для майбутньої професійної діяльності, важливо прогнозувати мету впливу на особистість студента не лише під час навчальної роботи під керівництвом викладача, а й самостійної роботи. Формування творчих навичок засвоєння знань і застосування їх на практиці, що реалізується на заняттях, потребує подальшого розвитку і стабілізації у психічній сфері особистості спеціальної організації педагогічних впливів під час самостійного виконання студентами різноманітних завдань. Отже, самостійна творча діяльність студентів є основою підготовки сучасних фахівців.

Бажано зі студентами розглядати задачі які б не лише розвивати мислення, розширення сфери застосування теоретичних знань, а й виникненню професійних орієнтирів студентів, що змінює їхні погляди на інформатику і сприяють розвитку інтересу до фундаментальних знань в системі загальної інженерної освіти. Такі цілі реалізують задачі таких типів:

**Завдання 1.** На диску С: в кореновому каталозі створити каталог, ім'я якого співпадає з прізвищем студента. В створеному каталозі створити три підкаталоги: **Text, Pic, AnyDocs**.

Запустивши програму **Блокнот (NotePad)**, створити тексто-

вий документ наступного змісту:

В кінці документа написати своє прізвище, та поставити дату та час (за допомогою засобу програми Блокнот.

Вирізати речення “Комп’ютер являє собою універсальну машину для обробки інформації” з кінця тексту і вставити його перед реченням “Алгоритм – це ...”.

Зберегти документ в каталозі (папці) Text під ім’ям text1.

Закрити програму Блокнот.

**Завдання 2.** Запустити програму Paint. Створити довільний малюнок, використавши всі інструменти для малювання з панелі інструментів. Зберегти документ в каталозі (папці) Pic під іменем Mal.

**Завдання 3.** Запустити програму WordPad і скопіювати в документ WordPad текст збережений в файлі text1. Вставити після тексту малюнок з файла Mal. Записати документ в файл під ім’ям Doc1 в каталог.

**Завдання 4.** Використати калькулятор для обчислення значення виразів:

$$y = \frac{x}{x+1} \sin x + \arctg x ,$$
$$y=2^x,$$

де  $x$  – номер варіанту. Скопіювати одержані значення в документ Doc1. Зберегти зміни в документі.



**Завдання 4.** Скопіювати файли text1, Mal, Doc1 в каталог AnyDocs.

Перемістити папки Pic і Text всередину папки AnyDocs.

**Завдання 5.** Створити ярлик на робочому столі для завантаження документу Doc1.

Створити ярлик для папки AnyDocs.

Після перевірки виконання завдання видалити каталог AnyDocs і ярлики в корзину.

У навчальному процесі вузу можна виділити такі рівні готовності студентів до самоосвіти: початковий, середній і високий [1].

На *початковому* рівні мотиви самоосвіти студентів в основному стихійні. Вони поки що не пов'язують особисті потреби в самоосвіті з суспільними інтересами. Знання з навчальних предметів носять ізольований характер. Часто студенти не бачать навіть внутрішньо предметних зв'язків між різними науковими поняттями. Вміння працювати з джерелами інформації не систематизовані, деякі не використовуються зовсім. На цьому рівні студенти не вміють самостійно організувати самоосвіту, а можуть лише з певною мірою старанності і ретельності виконувати вказівки і рекомендації викладачів.

Для *середнього* рівня готовності студентів до самоосвіти притаманне прагнення навчитися самостійно ставити перед собою мету самоосвіти і якісно виконувати її. Вони розуміють необхідність пов'язувати самоосвіту з інтересами суспільства, але не завжди вміють чітко сформулювати відповідну мету. Знання з навчальних предметів систематизовані, але ще не завжди чітко усвідомлюються взаємозв'язки окремих дисциплін. Студенти вміють працювати з основними джерелами інформації, але не можуть раціонально застосовувати їх в самоосвітній діяльності в повному обсязі. Студенти вміють самостійно організувати процес самоосвіти, але не досить раціонально планують цілісний засіб самоосвіти.

Для *високого* рівня готовності студентів до самоосвіти характерні глибоке розуміння особистістю необхідності керуватися в самоосвіті соціальне вагомою метою, вміння чітко сформулювати таку мету, бажання досягти її оптимальним способом. Знання особистості мають цілісний характер. В їх основі лежить глибоке розуміння наявності внутрішньо- і міжпредметних зв'язків у науці. Студенти вміють раціонально застосовувати різні джерела інформації у своїй діяльності. Він уміє на науковій основі оптимально керувати процесом самоосвіти від планування до здійснення задумів і самоконтролю отриманих результатів.

Формування у студентів готовності до самоосвіти проходить у педагогічному процесі. Тому його спеціальна організація і подальша реалізація є головною умовою, яка спонукає студентів до самоосвіти [2, 3].

Необхідно відмітити, що рівень готовності студентів до самоосвіти обумовлений не одним яким-небудь мотивом. Опитувані називали по кілька мотивів.

Результати опитування студентів факультету інженерних та інформаційних технологій Технологічного університету Поділля щодо мотивації до творчої діяльності та до самоосвіти. Серед відповідей перше місце займає соціальний професійно-ціннісний мотив – бажання як можна краще підготуватись до роботи програмістом. До найважливіших мотивів, названих більшістю опитуваних слід віднести утилітарний мотив – бажання підвищити свій загальний рівень (50,3%).

Таблиця

№ п/п	Види мотивів	Відповіді студентів (у % від кількості опитуваних)
1	Бажання як можна краще підготуватися до професійної діяльності	51,5
2	Бажання підвищити свій загальний рівень з інформатики	50,3
3	Інтерес до предмету, теми, питання	45,7
4	Бажання набути навичок користувача	42,9
5	Повага до самого себе	25,8
6	Спілкування з викладачем	23,3
7	Прагнення бути не гірше за інших	19,3
8	Бажання мати престиж в групі	5,3
9	Задоволення від самого процесу творчості	5,2
10	Подобається виступати на конференціях	4,4

Із виділених мотивів, що мають найбільшу вагу серед інших, перші п'ять можна вважати внутрішніми, інші – зовнішніми мотивами та стимулами.

Таким чином, розвитку творчого потенціалу майбутніх спеціалістів в період навчання сприяють:

- глибокі теоретичні знання, знайомство з актуальними проблемами науки і станом практики застосування набутих знань;
- пошук можливостей розв'язання цих проблем з позицій власного досвіду діяльності;
- володіння методикою науково-дослідної роботи, надбання закріплення дослідницьких умінь;
- формування якостей інженера-дослідника;
- організація навчального процесу на принципах педагогічної взаємодії, співпраці, співтворчості;
- забезпечення умов для активної пізнавальної діяльності студентів.

#### Література

1. Психология и педагогика. Учебн. пособие / Под ред. К.А. Абульхановой, Н.В. Васиной, Л.Г. Лаптева, В.А. Слостенина. – М.: Совершенство, 1998. – 320 с.
2. Рубинштейн С.Л. Основы общей психологии. – СПб: Питер, 1999. – 424 с.
3. Нечаев Н.Н. Психолого-педагогические аспекты подготовки специалистов в вузе. – М.: Изд-во МГУ, 1985. – 113 с.
4. Володарская И.А., Арташкина Т.А. Использование профессиональных задач в процессе обучения / Новые методы и средства обучения. – М., Изд-во «Знание», 1989, № 4(8). – С. 69-90.

## МІСЦЕ ВІДКРИТОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В ОБРОБЦІ ГРАФІЧНОЇ ІНФОРМАЦІЇ

В.В. Осадчий

м. Мелітополь, Мелітопольський державний педагогічний університет

В умовах ринкової економіки та капіталістичного суспільства, до якого прагне дійти наша країна, за все треба платити, за інформацію та засоби її обробки також. Ліцензійні програмні засоби створення і обробки графічних даних (CorelDRAW, Adobe Photoshop, Adobe Illustrator тощо) коштують недешево. Не для кого не секрет, що фінансове становище середніх та вищих навчальних закладів не дозволяє купувати таке програмне забезпечення, також і у нашої держави є більш важливі проблеми, на які слід витратити кошти. Отже, вихід для багатьох навчальних закладів – у використанні відкритого програмного забезпечення в процесі викладання дисциплін базового курсу інформатики.

Під час викладання спецкурсу “Комп’ютерна графіка” студенти вивчають такі графічні редактори, що відповідають поняттю “вільне програмне забезпечення”: XPaint та OpenOffice.org DRAW (оглядово), GIMP, XFig. Всі ці редактори призначені для роботи в операційних системах Linux, Unix FreeBSD, в деяких випадках навіть Windows.

XPaint – редактор растрової графіки, схожий на Paint, що убудований в операційну систему Windows, але він має масу додаткових можливостей.

XPaint, незважаючи на свою простоту, являє собою набір необхідних засобів по створенню і редагуванню зображень. Редактор функціонує в операційному середовищі Linux і запускається командою `xpaint`. При запуску з’являється вікно з набором інструментів для малювання і панеллю управління (рис. 1), яка надає можливість роботи з файлами.

Робоче вікно містить: аркуш, на якому можна малювати, можливо, разом з його збільшеною копією і двома смугами прокручування (горизонтальною і вертикальною); ряд меню, що дозволяють виконувати різні дії з зображенням; область із двома палітрами і кнопками для роботи з ними.

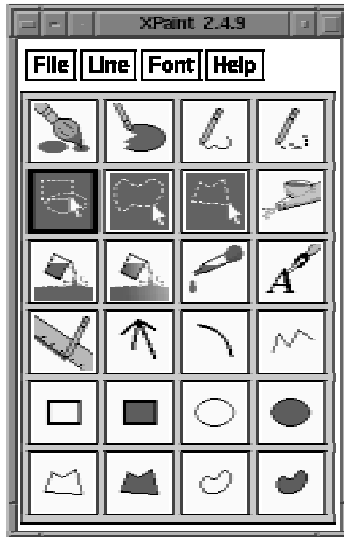


Рис. 1.

OpenOffice.org DRAW – багатофункціональний редактор створення різноманітної графічної продукції, працює і з простими геометричними фігурами та текстом, і з діаграмами та графіками, і з таблицями та формулами, а також зі слайдами тощо. Цей редактор з’явився не так давно і вже став поширеним серед користувачів.

Редактор має звичний інтерфейс для того, хто звик працювати програмними засобами, що працюють у середовищі Windows: панелі прокручування, стандартний рядок меню та кнопки створити, зберегти, відкрити, роздрукувати тощо. Він має 6 панелей: панель функцій, панель об’єктів, панель інструментів, панель гіперпосилання, панель опцій та панель кольорів.

OpenOffice.org DRAW працює у трьох режимах зображення: режим слайдів, режим тла, режим шарів. Загалом подібний до графічного редактора CorelDRAW, OpenOffice.org DRAW має таку відмінну рису, як плаваючі вікна, якими дуже зручно користуватися, коли обмаль простору для створення графічного зображення.

GIMP (*GNU Manipulation Image Program* – програма обробки зображень) – це могутній професійний графічний редактор, що має масу допоміжних програм, і володіє інструментами для



створення растрової графіки і засобами підготовки векторної графіки. Також у його склад входить унікальний редактор фрактальної графіки.

Перша версія GIMP була випущена у лютому 1996 і справила значний вплив як перша дійсно професійна і безкоштовна програма обробки зображень, що могла конкурувати з великими комерційними графічними редакторами. GIMP постійно розвивається, з'являються нові версії програми з новими функціональними можливостями.

Новачки можуть використовувати GIMP для простого малювання, проте професіонали напевно оцінять засоби, що надає GIMP для корекції і ретуші фотографій, інтерактивної пакетної обробки, створення зображень, конвертування графічних форматів тощо.

Розробники програми намагалися включити в неї всі традиційні засоби малювання і редагування, без яких не обходиться жоден графічний редактор, і найсучасніші засоби обробки зображень. Він дуже схожий на Photoshop, широко розповсюджений комерційний графічний пакет для ОС Windows. GIMP включає всі ті засоби, що є в програмі Photoshop, і, "розуміє" формат файлів, використовуваний у цій програмі, отже, користувачі Photoshop можуть легко перейти на використання GIMP.

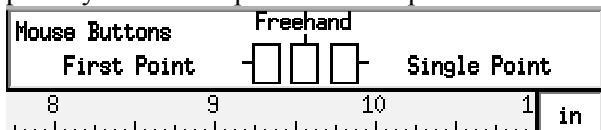


Рис. 2.

XFig – графічний редактор для роботи з векторними зображеннями. Основне вікно редактора містить чотири панелі керування. У редакторі існує покажчик функцій миші (рис. 2) призначений для виводу інформації про призначення кожної з трьох кнопок миші, у залежності від обраного інструмента чи стану редактора.

Знайомство студентів з комп'ютерною графікою починається із засвоєння закономірностей зорового сприйняття форми і простору, понять із образотворчого мистецтва (види та стилі мистецтва, види живопису, теорія кольору тощо) та естетики (художній образ, прекрасне, потворне, естетичний смак тощо), далі

відбувається знайомство з графічними редакторами шляхом порівняння та аналогії вільного програмного забезпечення з давно відомими комерційними програмами для створення та обробки графічних даних.

У процесі викладання курсу “Комп’ютерна графіка” ми прагнемо вносити корективи у його зміст та структуру з огляду на фах та спеціалізацію студентів, що повинні вивчати курс.

Для студентів МДПУ, що навчаються за спеціальністю “Вчитель хімії і основ інформатики та обчислювальної техніки”, ми надаємо простий користувальницький курс з опорою на шкільний курс викладання інформатики та з урахуванням специфіки майбутньої вчительської професії (унаочнення та візуалізація навчального матеріалу, естетичне виховання школярів тощо). Студентам МДПУ, що навчаються за спеціальністю “Вчитель математики і основ інформатики та обчислювальної техніки”, ми маємо на меті викладати також користувальницький курс, але з введенням математичних формул, наприклад, опису кольору (три закони Грассмана), вирахування потрібних лінійтур растру, кута повороту растру, дозволу оцифрування тощо. Для студентів Міжнародного університету розвитку людини “Україна”, що навчаються за спеціальністю “Комп’ютерний еколого-економічний моніторинг”, ми пропонуємо більш складний і розширений користувальницький курс, з елементами роботи з 3-D графікою, також вводяться поняття ергономіки, дизайну та реклами.

Проте відмінності, які вносяться з урахуванням специфіки спеціальності студентів, істотно не впливають на весь зміст курсу, метою якого є не тільки надати учням теоретичні знання з образотворчого мистецтва, естетики, комп’ютерної графіки, навчити працювати з різноманітними графічними редакторами, а й розвинути естетичну свідомість та творче мислення особистості студента, навчити відмежовувати прекрасне від потворного та жити і творити за законами краси.

Таким чином, викладання курсу “Комп’ютерна графіка” за допомогою відкритого програмного забезпечення дозволить не тільки уникнути фінансових та юридичних проблем, але і надасть змогу розширити кругозір студентів, краще підготувати їх як висококваліфікованих та професійних спеціалістів до вимог сучасного життя.

## СВОБОДНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ШКОЛЬНОЙ ИНФОРМАТИКИ

М.Е. Отставнов  
г. Москва, TechInvestLab.com  
maksim@otstavnov.com

На сегодня в школе, как и в некоторых других сегментах рынка, доминируют «альтернативные», нестандартные операционные системы и платформы (такие, как MS-DOS, Microsoft Windows, Microsoft Windows NT, MacOS версий до X). Пользование ими описывается в учебниках, на них ориентируются авторы отечественного (здесь и далее мы апеллируем к российскому опыту) учебного программного обеспечения (ПО).

В течение долгого времени «цена вопроса» просто не становилась поводом для сколько-либо серьезного анализа – ни для кого не секрет, что доля контрафактного ПО в школах очень высока, а общественное мнение склонно считать «легальность» программного обеспечения вопросом более отвлеченно-академическим, нежели насущно-практическим, а там, где дело касается зарубежных правообладателей, – относиться к нему не как к «пиратскому», а как к «трофейному».

Имеются веские причины, чтобы постепенно отказаться от этой недоброй традиции и вернуться в русло, более соответствующее магистральным линиям развития информационных и коммуникационных технологий:

- нестандартные системы ненадежны и небезопасны. С распространением в школах компьютеров и особенно сетей (включая доступ к региональным и глобальным сетям) ущерб от вирусов, несанкционированного доступа к информации и т.п. станет заметной статьей издержек информатизации школы;
- нестандартные системы дороги. Практически для всех стандартных технологий имеются свободные реализации (или, по крайней мере, конкурентный рынок реализаций), в то время, как ожидать появления свободной реализации интерфейсов, допустим, той же MS Windows в обозримом будущем не приходится, и, соответственно, цены будут оставаться моно-

- полно завышенными (в разы и десятки раз). Мы не сторонники «экономии на детях», но выделяемые на информатизацию школы деньги можно расходовать гораздо более разумно – от их вложения в перспективные разработки до повышения окладов учителей и обслуживающего персонала – не жели платить по сотне долларов за сторублевый диск;
- нестандартные системы ограничивают выбор оборудования и зачастую предъявляют весьма высокие требования к его параметрам;
  - поставщик нестандартных систем получает неосновательное преимущество в других сегментах рынка. Если систематически подменять обучение общим принципам и стандартным технологиям изучением конкретных программ, через некоторое время специфические навыки пользования ими распространятся в обществе настолько, что поставщикам конкурирующих технологий и решений пробиться на рынок будет совсем нелегко;
  - несвободное ПО отрезает учащихся и их наставников от самых прогрессивных технологических решений.

С технической точки зрения сегодня свободные реализации стандартных технологий способны качественно обеспечить учебный процесс в школе. Это подтверждается результатами, достигнутыми и на единственной сегодня в России масштабной экспериментальной площадке, где в учебный процесс систематически внедряется ОС GNU/Linux (ЦКТ МГИУ, <http://www.ctc.msiu.ru>), и рядом зарубежных проектов. Имеющийся пул свободных программ с избытком перекрывает потребности любого разумного учебного курса по информатике, причем большинство их способно работать на массовом и недорогом оборудовании, распространенном в школах (ПК архитектур IBM PC и Apple Macintosh, терминальные классы от SUN на UltraSPARC и т.п.), включая весьма «пожилые» модели. Некоторые программы требуют определенных усилий по локализации (переводу элементов интерфейса и документации), однако эти затраты на порядок меньше, чем стоимость «легализации» используемого сегодня несвободного софта.

С начала текущего (2002-03) академического года мы проводим небольшое техническое исследование, касающееся приме-

нимости существующих свободных программ в школе (в пределах «Обязательного минимума...» [1]), результаты которого публикуются в газете «Информатика» (г. Москва) и должны впоследствии выйти отдельной брошюрой. Полный цикл включает 14 сообщений, 10 из которых на момент подготовки настоящих тезисов уже опубликованы или готовятся к публикации (мы также передаем их тексты для распространения на диске на условиях свободной лицензии).

Упомянутый цикл публикаций и тем более эти краткие тезисы не призваны заменить методические руководства, учебники или техническую документацию к соответствующим программам; скорее, мы стремимся привлечь к свободным программам внимание широкой педагогической общественности. В настоящее время нам известен лишь один русский учебник, ориентированный на использование в школе преимущественно свободных программ [1], его текст также доступен для свободного использования.

## Право и экономика ПО

Поскольку в литературе и прессе существует изрядная путаница с терминологией, имеет смысл привести определения.

*Свободными* называются программы, автор (или иной обладатель имущественных авторских прав) которых обнародовал их в сопровождении так называемой «свободной лицензии», передающей приобретателю права: 0) пользоваться программой для любых целей и на неограниченном количестве компьютеров или мест в сети; 1) беспрепятственно получать доступ к ее исходным кодам; 2) изготовлять (производить) неограниченное количество дополнительных ее экземпляров, как для собственного пользования, так и для распространения или сдачи в прокат/аренду на тех же условиях, возмездно или безвозмездно; 3) модифицировать ее как для собственного пользования, так и для распространения на тех же условиях.

Конкретная «лицензия» может предоставлять приобретателю дополнительные правомочия, безусловно или на определенных условиях, и это не делает программу несвободной. *Несвободной* является программа, распространяемая на условиях, ог-

*раничивающих* вышеперечисленные права приобретателя.

Свободные программы (free software) не следует, как это часто делают, путать со «свободно распространяемыми» или «бесплатными».

Также следует иметь в виду, что, хотя термин «программы с открытыми исходниками» (open source software) часто используется как синоним «свободных программ», им иногда злоупотребляют.

Свободное ПО не следует путать и с «открытым» (open software): «открытость» относится к соблюдению стандартов на интерфейсы, и только, а свобода – к условиям лицензирования и модели разработки.

И наконец, не следует путать «коммерческое» с несвободным, а «некоммерческое» – со свободным: значительная часть (вероятно, большая) свободного кода разрабатывается в коммерческих рамках. В то же время, существует большое количество некоммерческого несвободного кода.

Очень часто (особенно, в популярной прессе) можно столкнуться с утверждением о якобы «бесплатности» свободных программ.

Экземпляр произведения (в том числе, программы), вне зависимости от формы его существования, является материальным (ограниченным) благом, и в этом качестве не может быть «бесплатным», сколь бы исчезающе малой не была стоимость его приобретения. «Бесплатное» материальное благо – это экономический перпетуум мобиле.

Произведение как таковое – идеальный объект, не ограниченный по своей природе в использовании, и говорить о его «цене» по меньшей мере некорректно, вне зависимости от того, каковы были затраты материальных ресурсов при его создании.

Узел терминологических тонкостей и концептуальных сложностей можно разубить, введя альтернативное определение: *свободные программы – это программы, все услуги по разработке, модификации, сопровождению и поддержке которых продаются на свободном рынке*. Если вы видите, что, например, свободный дистрибутив какой-либо версии GNU/Linux, включающий, помимо операционной системы и нескольких операционных сред, большое количество прикладных программ, стоит от

нескольких десятков до нескольких тысяч рублей, в то время, как «набрать» даже небольшую часть этой функциональности программами несвободными может обойтись и в десять, и в сто раз дороже, причина этому не в «бесплатности» чего-либо, а в конкурентности рынка.

Интересующихся экономикой ПО мы отсылаем к соответствующей литературе [3, 4], а здесь ограничимся лишь указанием на то, что соответствующий рынок успешно развивается уже в течении четверти века, есть примеры исполнения им задач, немислимых для отдельных корпораций (самый яркий из них – разворачивание в девяностых на основе свободных программ и систем всемирного сообщества сетей Интернет), а также на то, что на сегодня практически все лидеры компьютерной отрасли высказываются в поддержку свободного ПО.

### Свободные ОС

Операционной системой принято называть совокупность программ, управляющих оборудованием и предоставляющих его в качестве ресурсов прикладным программам.

Операционная система, наряду с оборудованием, сетью и графикой, является одной из основных вычислительных платформ, т.е. одним из столпов, на которых возводится здание информационных технологий.

С начала девяностых Всемирной организацией стандартизации (ISO) проводится стандартизация архитектуры ОС, т.е. совокупности системных вызовов (интерфейса прикладного программирования) и синтаксиса команд оболочки (пользовательского интерфейса). Текущая версия стандарта принята в 2002 г.

Относительно существующих стандартов все ОС можно разделить на ряд категорий: 1) стандартизированные, 2) альтернативные, 3) неполные, 4) специализированные.

К стандартизированным (открытым) – следующим или стремящимся следовать стандартам – на сегодня относятся ОС на основе ядер Unix, Linux и BSD (включая Darwin, лежащую в основе MacOS X). Называть все их «юниксами» некорректно, поскольку Unix является торговой маркой Open Group.

Самой распространенной альтернативной ОС, то есть, ОС,

стремящейся реализовать ту же функциональность, что и стандартные, но несовместимыми способами, является Microsoft Windows NT (сегодняшние версии ее называются MS Windows 2000 и XP), во многом продолжающая «доюниксовские» традиции ОС RSX-11 и VMS корпорации DEC. К альтернативным можно отнести и «постюниксовские» архитектуры ОС plan9 и Inferno.

«Неполными» можно назвать ОС с нереализованной базовой функциональностью, такой как многозадачность или многопользовательность (разделение прав). В основном, это ОС, унаследованные от досетевой эпохи «ПК», такие, как MS-DOS, «классическая» MacOS или Microsoft Windows на основе MS-DOS (3.x, 9x и Me), но также и ряд новых ОС, предназначенных для маломощных карманных компьютеров: PalmOS, Microsoft Windows CE и т.п. Присутствие неполных ОС в ИТ-решениях – основной фактор риска с точки зрения целостности и защищенности последних.

Специализированные ОС предназначены, в основном, для управления встроенными микрокомпьютерами, но также распространены и на суперкомпьютерах.

Основными тенденциями последнего десятилетия являются постепенное вытеснение стандартизованными системами прочих (включая их распространение в сферах, ранее монополизированных специализированными и альтернативными ОС) и постепенное вытеснение свободными реализациями стандартов (Linux, BSD) несвободных (Unix различных фирменных модификаций).

К числу тенденций уже самых последних лет можно отнести рост доли Linux-ОС в сравнении с BSD-ОС, ранее доминировавших в качестве платформы для Интернет-сервисов.

В настоящее время существует развитый рынок поставок – «дистрибутивов» свободных ОС с ядрами Linux и BSD. Появляются и базирующиеся в СНГ – с участием ключевых российских и украинских разработчиков – команды, выпускающие такие дистрибутивы, как ALT Linux и ASPLinux.

Как правило, дистрибутивы свободных ОС включают, помимо системных компонентов, широкий спектр прикладных программ разной направленности, число которых может варьировать от сотен для компактных однодисковых дистрибутивов



до тысяч для многодисковых. Описываемые ниже прикладные программы можно найти в составе практически каждого универсального дистрибутива. Нужно иметь в виду, что многие из них имеют версии также и для несвободных, в том числе, альтернативных ОС.

### Свободные прикладные платформы, программы и пакеты

*Коммуникации.* Типичные локально-сетевые приложения – это разделение файловых структур, услуги печати, централизованная аутентификация пользователей и т.п., предоставляемые в масштабе от одного офиса (комнаты) до целой группы зданий.

Типичные интернет-приложения – это электронная почта, WWW, обмен «быстрыми сообщениями» («интернет-пейджинг») и пр. системы, разворачиваемые обычно в глобальном масштабе.

Основной тенденцией последнего десятилетия стало постепенное вытеснение специализированных локально-сетевых низкоуровневых протоколов интернет-протоколами (TCP/IP). Даже протоколы, специально разработанные для персонально-компьютерных систем могут сегодня развертываться поверх TCP/IP.

Поддержка TCP/IP на сетевом и транспортном уровнях впервые реализована в ОС BSD и включена в современные ОС на уровне ядра. На уровне ядра обычно – из соображений эффективности – реализуются и такие, например, сервисы, как экранирование (firewalling) трафика. Протоколы более высокого уровня реализуются в виде системных или прикладных сервисов.

В открытых ОС локально-сетевая функциональность традиционно реализуется в рамках концепции Сетевой файловой системы NFS. NFS позволяет предоставлять каталоги для удаленного монтирования и монтировать их удаленно. Так же традиционно более сложная семантика сетевых взаимодействий, внедрялась с помощью баз данных совместного доступа и специальных протоколов доступа к ним (наряду с SQL), таких, как DAP (X.500) или его упрощенная и более популярная версия LDAP. LDAP входит в поставку практически всех дистрибутивов свободных ОС и широко используется в современных системах, но пока в основном для решения административных задач.

В персонально-компьютерных системах большее развитие получил альтернативный подход, опирающийся на концепцию обмена блочными сообщениями (SMB), на основе которого разработаны семейства локально-сетевых протоколов высокого уровня, к сожалению, не стандартизированные и переусложненные.

Существует современная авторитетная и эффективная свободная реализация SMB и (частично) его позднейших расширений, называемая SAMBA ([www.samba.org](http://www.samba.org)). SAMBA входит в поставку практически любого дистрибутива свободных ОС.

Серверная и инфраструктурная составляющая межсетевых цифровых коммуникаций – традиционное приложение для свободного ПО. Мы не будем погружаться в организацию локальных сетевых служб, WWW-серверов и т.п., хотя это важная и нужная в школе тема, она не имеет прямого отношения к предмету школьной информатики. Примером клиентского пакета, реализующего современную сетевую функциональность сессионного, презентативного и прикладного уровня, являются Mozilla (Web-браузер, программа работы с электронной почтой и новостями, IRC-чатами, простой редактор гипертекста). Mozilla реализует новейшие версии языков разметки, применяемых в WWW, а также язык разметки XUL, на котором написан пользовательский интерфейс. Реализация XUL делает Mozilla легко адаптируемым и расширяемым прикладным пакетом, практически полностью удовлетворяющим базовые потребности в Интернет-коммуникации.

*Графические операционные среды.* В отличие от сети, графическая среда не является компонентом ОС и представляет собой системно-независимую прикладную платформу.

Юридического стандарта графической среды на сегодня не принято, однако роль фактического отраслевого стандарта играют спецификации X Window System, разрабатываемые Open Group (<http://www.x.org>).

X Window System определяет протокол взаимодействия X-сервера (программы, управляющей терминальными устройствами) и X-клиентов (программ, нуждающихся в графическом интерфейсе), для этого протокола сеть прозрачна, и клиент с сервером могут исполняться на разных машинах, под разными ОС и

на разной аппаратной платформе.

Самой популярной реализацией X Window System является свободная XFree86, доступная для большинства ОС.

Поверх платформы X работает целый ряд различных свободных графических сред, начиная от «легких», представленных оконными менеджерами класса BlackBox, IceWM или WindowMaker и кончая интегрированными средами KDE и GNOME, широкий выбор которых позволяет организовать комфортную рабочую или учебную среду практически на любой топологии оборудования, от классов «тонких» терминалов до высокопроизводительных рабочих станций.

*«Офисные» приложения.* «Офисные» приложения, т.е. использование компьютера для автоматизации документооборота (в широком смысле) до недавнего времени оставались проблемными для свободного ПО из-за широкого распространения нестандартных форматов документов и высоких требований к производительности оборудования, предъявляемых стандартами.

Разработка на базе международного стандарта SGML расширяемого языка разметки XML (менее требовательного, чем полный SGML) и повысившаяся производительность средней рабочей станции вызвали к жизни целый ряд проектов, среди которых следует отметить офисные пакеты OpenOffice.org и KOffice, интегрированный со средой KDE.

Возможности входящих в их состав программ с избытком перекрывают требующиеся для знакомства учащихся с редактированием размеченного текста и работой с электронными таблицами, предусмотренного «Обязательным минимумом...». В состав OpenOffice.org, кроме того, входит развитый редактор векторной графики.

*Компьютерная графика.* В части редактирования графики растровой широкие возможности предоставляет редактор GIMP, являющийся одним из самых мощных в своем классе. GIMP снабжен встроенным интерпретатором языка Schema, что придает ему дополнительную методическую ценность в качестве средства обучения обработки графической информации.

\* \* \*

Ограниченный объем статьи не позволяет более подробно остановиться на перечисленных программах и рассказать о свободных разработках в области других приложений компьютера (например, мультимедиа), а также о свободных инструментальных (необходимых для практики в программировании) средствах.

Заинтересованному читателю мы рекомендуем знакомство с работами [2, 5–9] и с технической документацией, поставляемой в составе современных дистрибутивов свободных ОС.

### Литература

1. Министерство образования Российской Федерации. «Обязательный минимум содержания образования по информатике».
2. С.В. Сергеев, Н.А. Роганова. Практическая информатика: учебное пособие. Ч.1 – М.: МГИУ, 2001. (См. тж. <http://www.ctc.msiu.ru/materials/>.)
3. Эрик Рэймонд. «Собор и базар» // «Открытые системы», № 09-10, 1999 г.
4. Максим Отставнов (ред.). «Свободное программное обеспечение: бизнес-модели и корпоративные инициативы» – М.: ГУ-ВШЭ, 2001. Расширенную версию этого сборника можно найти на: <http://www.otstavnov.com/fsr>
5. Антон Ионов и др. OpenOffice.ru. Руководство пользователя. – М.: ALT Linux, 2002. – 115 с. (Тж. доступна на <http://docs.openoffice.ru>).
6. Дэвид Тейнсли. Linux и Unix: программирование в shell. – Киев: BHV, 2001.
7. Сергей Глушаков, Алексей Сурядный. Linux для дома и офиса: Учебный курс. – Харьков: «Фолио», 2002. – 389 с.
8. Karin Kylander & Olof S Kylander. GIMP: The Official Handbook. The Coriolis Group: 1999, ISBN 1-57610-520-2 (Тж. доступна на <http://manual.gimp.org>).
9. Carey Bunks. Grokking the GIMP New Riders Publishing, 2000; ISBN: 0735709246; 352 pp. (Тж. доступна на <http://gimp-savvy.com/BOOK/>).

## АВТОМАТИЗИРОВАННАЯ СИСТЕМА ПРОВЕРКИ ЗНАНИЙ TEST-2000

Е.А. Панина, Л.В. Гусева

г. Харьков, Академия пожарной безопасности Украины  
fd.apbu@list.ru

В большинстве развитых стран устные экзамены используются не только для оценки собственно знаний, но и для оценки способности человека внятно излагать свои мысли, общаться, вести разговор или выступать на публике. Что касается оценки человека с профессиональной точки зрения, то для этого есть огромное количество тестов, составленных для выявления у человека самых разных навыков, и важно, что тестирование используется не только в учебных заведениях, но и при получении профессиональных сертификатов, признаваемых работодателями.

В 2002 году на кафедре фундаментальных дисциплин в системе TEST-2000 были созданы базы для проведения контрольных работ и зачетов по дисциплине «Информатика и компьютерная техника». Совершенствование старых и разработка новых баз продолжают и в настоящее время.

Рассмотрим основные положения при работе с АСПЗ TEST-2000.

Система проверки знаний TEST-2000 состоит из трех частей:

**1-я часть: Ввод вопросов для тестирования** (собственно вопросы, варианты ответов, критерии оценки, время тестирования, количество вопросов для проведения тестов);

**2-я часть: Собственно тестирование.**

**3-я часть: Использование результатов тестирования** (просмотр и анализ).

В процессе функционирования системы в ее рамках формируется собственная база данных (БД), включающая:

- вопросы;
- варианты ответов;
- критерии оценки (необходимое количество правильных ответов для сдачи теста);
- критерии проведения тестирования (время тестирования,

количество вопросов, время на один вопрос, параметры сохранения тестов);

- результаты тестирования.

Вопросы для тестирования могут содержать не только тексты, но и рисунки. Количество вопросов может быть гораздо больше, чем предлагается одному тестируемому. Это позволяет системе при каждом следующем тестировании не только изменять порядок следования вопросов, но и выбирать разные вопросы из их списка. Такой подход исключает возможность фальсификации результатов тестирования.

Система позволяет сформировать произвольное количество ответов на каждый вопрос. Количество правильных ответов может быть несколько. Последовательность ответов на вопрос меняется при каждом следующем тестировании.

Критерием оценки служат баллы (в процентном выражении), набранные во время тестирования и соответствующие принятой системе дифференцирования знаний – «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

В процессе формирования БД вводится информация, содержащая нормативное время, отведенное на тестирование по конкретной теме. При превышении данного норматива тестирование прекращается. Создание БД сопровождается подсказкой, которую при необходимости можно отключить.

Удобный и простой графический интерфейс системы TEST-2000 позволяет без специальной подготовки использовать данную систему при создании баз для различных дисциплин.

Система также дает возможность созданные ранее базы по отдельным темам объединить в один модульный тест. В перспективе результаты тестирования могут использоваться при аттестации слушателей, а также при формировании зачетных или экзаменационных оценок.

Система TEST-2000 инсталлирована в учебном компьютерном классе кафедры фундаментальных дисциплин.

Пользователями системы являются:

1. преподаватели;
2. курсанты и студенты;

Каждый преподаватель имеет свой пароль. Пароль в основном регламентирует доступ к БД, содержащей вопросы и ответы,

а также к параметрам настройки тестов и самой системы тестирования.

Среди особенностей эксплуатации системы отметим следующие:

- Т.к. тестирование происходит в интерактивном режиме, количество одновременно тестируемых зависит только от количества персональных компьютеров в учебном классе.

- Необходимо задать ограничения, при которых тестируемый имеет доступ только к вариантам теста.

- Факт начала тестирования регистрируется внутримашинным таймером. При тестировании на заданную тему часы постоянно присутствуют на экране и стимулируют процесс тестирования.

- В процессе тестирования предусмотрена возможность возврата к предыдущим вопросам с целью изменения своего варианта ответа. При этом общее время тестирования остается неизменным. В файл БД, который предназначен для хранения результатов тестирования, в данной ситуации будет вноситься последний вариант ответа.

- Во время тестирования для каждого типа вопроса на экране предусмотрена компактная подсказка, суть которой заключается в том, как технологически правильно указать нужный ответ или ответы.

Применение новых технологий тестирования, при котором весь процесс сдачи теста проходит дистанционно, либо с использованием компьютерных программ, является шагом вперед по сравнению с традиционными методами тестирования, и существенно сокращает как время обработки результатов, так и вероятность ошибок.

## РАЗВИТИЕ АЛГОРИТМИЧЕСКОГО СТИЛЯ МЫШЛЕНИЯ МЛАДШИХ ШКОЛЬНИКОВ СРЕДСТВАМИ ЯЗЫКА ЛОГО

Н.В. Перегинец, И.А. Теплицкий  
г. Кривой Рог, Криворожский государственный педагогический  
университет

Выбор языка программирования для начинающих должен определяться психолого-педагогическими требованиями и методическими рекомендациями, позволяющими воспитывать у детей навыки хорошего стиля программирования. В общем виде можно сформулировать следующие требования к начальному языку программирования:

- логическая и синтаксическая простота;
- наглядность основных действий;
- исключение второстепенных понятий, заслоняющих основные принципы программирования;
- привлекательность в обучении;
- простота в общении с компьютером, дружественный интерфейс, возможность написания программ без предварительного длительного изучения языка.

В соответствии с этими требованиями в конце 60-х гг. известным американским педагогом С. Пейпертом, последователем идей Ж. Пиаже, была предпринята попытка разработать специальный язык для обучения детей основам программирования. В результате появился язык Лого, получивший международное признание и массовую популярность.

Основные характеристики языка:

- синтаксис, близкий к естественному языку;
- приспособленность к интерактивному режиму работы;
- ориентация на формирование самых общих представлений о программировании;
- необязательность глубоких знаний архитектуры ЭВМ;
- система графических примитивов.

Для того, чтобы этот язык был пригоден для обучения учащихся младших классов, в нем была реализована идея экранного исполнителя – так называемой черепашки. Сегодня черепашка –



это управляемое с помощью компьютера кибернетическое животное, существующее в языковой среде Лого. При движении черепашка может оставлять след, и эта ее способность используется для вычерчивания фигур на экране.

Неоценим оказался вклад Лого в методику обучения. Среда обучения Лого является примером новых технологий обучения, направленных на освоение средств, при помощи которых учащиеся могут самостоятельно добывать знания. Среда Лого позволяет так построить процесс обучения, что овладение языком программирования становится таким же естественным процессом, как изучение родного языка. Занятия в среде Лого превращают традиционную ситуацию компьютерного обучения. Здесь ребенок сам учит и программирует компьютер и, делая это, овладевает основами предмета.

Лого реализует новые подходы к обучению, нацеленные не на изучение правил, а на формирование процесса мышления. Лого является окном, позволяющим заглянуть в мыслительные процессы ребенка. На основе анализа данных о работе ребенка, накопленных в памяти компьютера, у педагога появляется возможность получить представление о том, как ребенок думает.

В этой среде ребенок сам управляет процессом обучения. Как и в реальной жизни, он сам ставит себе задачу, и сам находит пути ее решения. Из постоянно ждущего помощи от других он превращается в человека, самостоятельно ищущего и находящего решения. На собственном опыте ребенок учится делать выводы и обобщения.

В Лого изначально заложены принципы конструктивного обучения, в соответствии с которыми в процессе создания реального продукта (компьютерной программы) значительно повышается эффективность обучения. Это возможно только потому, что Лого – полноценный язык программирования, допускающий возможность создания настоящих, работоспособных программ.

Важную роль в современных условиях обучения играет отношение к ошибкам и их устранение. Отладка программы (поиск и исправление в ней ошибок), обычно являющаяся трудоемким процессом, в Лого значительно упрощается. Решение разбивается на множество мелких шагов, каждый из которых может быть проверен отдельно. Если допущена ошибка, то черепашка просто

не выполнит задания. Так как ученики могут самостоятельно найти ошибку и установить ее источник, они перестают бояться ошибок и чувствуют себя исследователями, не боящимися творить. Они обретают уверенность в своих силах и могут выразить свое знание предмета в форме, отличной от вербальной.

Лого привлекает еще и тем, что программирование из скучной дисциплины превращается в интересную игру, в процессе которой происходит быстрое и прочное усвоение основных понятий и навыков программирования.

В настоящее время Лого – одно из многих достижений в области методики преподавания и обучения. За годы своего существования Лого, многократно модернизируясь, произвел настоящую революцию в обучении.

Известно, что школьный курс должен отражать, основные направления развития информатики. К ним относится процедурное и объектное программирование. Обучение процедурному и объектному программированию при плавном переходе от одного к другому, без введения нового языка в изучение, сегодня возможно только при использовании Лого.

На первых этапах обучения Лого – классический процедурный язык, позволяющий легко усваивать принципы структурного программирования. При дальнейшем изучении языка (рекурсии, обработки списков и др.) в Лого становится видна его «объектная сторона».

Знакомство с системой Logo происходит после краткого обзора основных терминов и понятий информатики, и затем начинается естественное, подсознательное для ребенка формирование начал алгоритмического мышления через обучение черепашки. Вводится понятие команды, параметра команды, базовых команд, примеров их применения.

При введении новых понятий учитель может использовать методы проблемного обучения. Он состоит в том, что детям необходимо дать достаточное время на освоение той или иной группы команд, и всегда найдутся такие дети, которым надоест пользоваться несовершенными методами, и которые начнут задаваться вопросом: “А нельзя ли более просто и рационально сделать то же самое?”. Вот здесь и следует предложить им новые команды или понятия. Таким образом, процесс усвоения новых

понятий всегда будет востребован самими учащимися, что повышает эффективность процесса обучения.

При написании программ на Лого нередко приходится многократно повторять одни и те же команды. После того, как учащиеся заметят это, вводится определение циклического способа написания процедур. При подборе практических заданий в курсе особо выделяются задачи построения правильных многоугольников. В Лого имеется возможность определять процедуры с параметрами, т.е. работать с переменными величинами.

Понятие переменной величины – одно из фундаментальных понятий информатики и является достаточно сложным для понимания детьми. При формировании навыков работы с переменными мы обращаем внимание на особенности вызова значения переменной и на формальные и фактические параметры процедур. Используя наглядные примеры, знакомим учащихся с понятиями локальной и глобальной переменной, с образцами их рационального использования.

После того, как учащиеся овладеют навыками использования переменных, рассматривается арифметика Лого. Вводится понятие операции, как предписания, по которому вырабатывается результат, правила использования арифметических операций. Некоторые операции в Лого, которые выдают результат, называют датчиками.

После закрепления практических навыков с арифметическими операциями, основными из которых являются операции взятия случайного числа, остатка от деления, округления, отрабатываются навыки практического использования арифметических и неарифметических операций. Дети должны сами подойти к выводу о том, что каждая операция вырабатывает результат, который сразу же используется как параметр другой операции или команды. Это важнейший вывод: процедуры на Лого составляются не простыми перечислением предписаний языка, а их комбинациями, которые представляют собой в полном смысле конструирование сложных действий из более простых.

При рассмотрении практических упражнений решаются математические задачи: нахождение степени числа, среднего арифметического и т.д. После достаточно большой практики создания линейных алгоритмов производится знакомство с алго-

ритмической конструкцией “ветвление” и условной командой Лого. Рассматриваются истинность или ложность выражений и утверждений, формализуется понятие операций отношения, знакомые из курса математики, начальное знакомство с логическими операциями. Понятие алгоритма с ветвлением вводится на простых примерах с использованием жизненного опыта учащихся. После знакомства с записью условной команды Лого большое внимание уделяется формированию навыков написания процедур с использованием условий.

Особое место в программировании на Лого занимает знакомство и овладение навыками работы с рекурсией – мощным средством программирования, позволяющим эффективно писать программы. Умение пользоваться им составляет одно из важных направлений алгоритмической культуры. Известно, что использование механизма рекурсии, строго говоря, не является обязательным, т.к. любой рекурсивный алгоритм может быть заменен эквивалентным ему циклическим алгоритмом. Экономия мышления – главный аргумент использования рекурсии. С помощью рекурсии можно компактно записать сложный алгоритм. Теория рекурсивных функций – это подход к формализации теории алгоритмов – сущности информатики. Рекурсивные функции заложены в языке Лисп, а Лого написан именно на Лиспе и наследует от него единственный, но универсальный тип данных – список. Практическое знакомство с рекурсией происходит при написании процедур, рисующих окружности, правильные многоугольники.

Далее в учебном курсе происходит формализация понятий и отработка практических навыков работы по использованию декартовых координат при составлении алгоритмов и программ на основе знаний по математике (вычерчивание отрезков, треугольников, окружностей и других фигур с помощью циркуля и линейки). Благодаря методу координат сегодня к решению задач по геометрии привлекаются компьютеры. Акцент делается на использовании компьютера как инструмента для геометрических построений.

Поскольку Лого является весьма мощным средством в области обработки текстов, в его арсенале есть очень эффективные средства для работы с текстом прямо на экране.

Одной из основных задач преподавания информатики в школе является интегрирование ее с другими школьными предметами. При изучении списков целесообразно рассмотреть программы, которые можно использовать на уроках языка.

Лого объединяет в себе черты многих языков программирования. Изучение Лого как начального языка значительно облегчает дальнейшее изучение профессиональных языков программирования. Лого является интерпретатором, обеспечивающим диалоговый характер общения с пользователем. Структурное построение программы роднит Лого и Паскаль. Лого, как и Форт, – не только структурные, но и саморасширяющиеся языки, допускающие введение новых слов и расширение словарного запаса языка.

В заключение приведем аргументы в пользу введения Лого в школьный курс информатики:

- Лого удовлетворяет требованиям начального языка программирования;

- философия Лого – истинное достижение в методике обучения;

- Лого дает широкий взгляд на программирование;

- обучение Лого является пропедевтикой изучения профессиональных языков программирования;

- Лого пригоден для изучения основ машинной графики.

Опыт показывает, что изучение курса “Основы информатики” на базе системы Лого в 6–7 классах является эффективным средством формирования основ информационной культуры. Лого активизирует поисковую и творческую деятельность, создает условия для развивающего обучения, повышая уровень познавательной активности школьников.

Именно поэтому за рубежом лучшим стартовым языком обучения детей основам программирования и машинной графики признан язык Лого. Ему посвящены десятки книг, издаются специальные журналы. В Болгарии, Японии, США и Франции уроки информатики с использованием Языка Лого начинаются с V класса, и уже обсуждается вопрос о введении его изучения с III класса. К сожалению, в Украине Лого пока мало известен, практически отсутствуют методические пособия и учебники.

#### Литература:

1. Пейперт С. Переворот в сознании: Дети, компьютеры и плодотворные идеи. – М., 1989.
2. Машбиц Е.И., Каптелинин В.Н., Маргулис Е.Д. Введение в язык Лого: Учеб.пособие. Под общ. ред. А.А. Стогния, Е.А. Ющенко, Е.И. Машбица. – К.: Вища школа, 1989. – 208 с.
3. Малярчук С.Н. Среда Лого как способ реализации деятельного подхода в обучении. // Применение новых компьютерных технологий в образовании: Тезисы докладов международной конференции. – Самара, 1993. – С. 14.
4. Дьяконов В.П. Язык программирования Лого. – М.: Радио и связь, 1991. – 144 с.
5. Машбиц Е.И. Психологические основы управления учебной деятельностью. – К.: Вища школа, 1987. – 224 с.

## **ВИКОРИСТАННЯ КОМП'ЮТЕРНО-ОРІЄНТОВАНИХ МЕТОДИК ПРИ ВИКЛАДАННІ ФУНДАМЕНТАЛЬНИХ ДИСЦИПЛІН**

Л.І. Побережна

м. Івано-Франківськ, Навчально-методичний кабінет професійно-технічної освіти управління освіти і науки Івано-Франківської обласної державної адміністрації  
[MetodkabinetPTO@ukr.net](mailto:MetodkabinetPTO@ukr.net)

Сучасний світ переходить до інноваційної економіки, яка потребує модернізації освіти. Це період, коли зміна ідей, технологій, знань відбувається швидше, ніж зміна поколінь, що зумовлює нові завдання у розвитку освіти, де навчальний процес повинен виконувати дві функції: підготовку студента до самостійного оволодіння знаннями та трансформування набутих знань у життєву компетентність.

Нові завдання освіти вимагають застосування 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.

Основою впровадження їх при викладанні фундаментальних дисциплін є використання навчальних та контролюючих програм. Навчальні програми з допомогою машинної графіки, використовуючи імітаційне моделювання, дозволяють ознайомити учнів з такими явищами та процесами, продемонструвати які іншим способом дуже складно, або практично неможливо. При цьому у студентів з'являється можливість не лише пасивно спостерігати, але й безпосередньо впливати на їх перебіг шляхом зміни тих чи інших параметрів.

При викладанні курсу хімії це, наприклад, може бути імітація протікання реакції; фізики – броунівського руху та дифузійних процесів, математики – побудова графіків складних функцій з можливістю їх аналітичного дослідження тощо.

Таким чином, застосування сучасних комп'ютерно-орієнтованих методик навчання стимулюватиме динаміку та ефективність навчального процесу та допоможе педагогам у фо-

рмуванні наукового мислення, розвитку пізнавальної активності та реалізації творчих здібностей студентів.



## **О СОСТАВЛЕНИИ И РЕАЛИЗАЦИИ УЧЕБНЫХ ПЛАНОВ ПО КИБЕРНЕТИЧЕСКИМ ДИСЦИПЛИНАМ В ВЫСШЕЙ ШКОЛЕ**

А.П. Полищук, С.А. Семериков

г. Кривой Рог, Криворожский государственный педагогический  
университет  
cc@kpi.dp.ua

Любой технологический процесс выполняется “в прямом направлении” – от сырья к конечному продукту. Однако любая технология разрабатывается (проектируется) в обратном порядке – от требований к готовому продукту к характеристикам промежуточных до входных сырьевых включительно и к средствам обеспечения этих характеристик на всех стадиях переработки. Учебный процесс не исключение – при составлении учебных планов вначале должны быть определены требования к знаниям и навыкам выпускников и сформированы учебные планы заключительных предметных курсов, определен уровень подготовки слушателей, необходимый для восприятия заключительного и, очевидно, наиболее сложного материала – в том числе по поддерживающим дисциплинам, не относящимся непосредственно к выпускающим кафедрам. Затем должны формироваться планы более ранних этапов обучения и т.д. вплоть до требований к уровню подготовки абитуриентов по профильным дисциплинам. Сказанное достаточно тривиально, но учебные планы по кибернетическим дисциплинам для связанных с ними специальностей (или специализаций) во многих вузах характеризуются эклектизмом как по номенклатуре, так и последовательности преподавания.

В последнее время предпринимаются попытки устранить непоследовательности в учебных планах различного рода организационными мерами. Так, преподаватели всех вузов сейчас обязаны предоставить в учебную часть рабочие (плюс рабочие учебные) программы читаемых курсов (что в принципе не вызывает возражений); первые пункты этих программ должны содержать цель и задачи курса, а также перечень знаний и умений студента после его освоения. Эти пункты явно не по адресу –

квалифицированный специалист может прекрасно прочитать курс, не зная цели, с которой он был включен в учебный план. Например, нелегко обосновать наличие в учебных планах для специальности “Информатика и экономика” (готовит специалистов по математическим методам решения экономических задач) предмета “Микропроцессорные устройства и системы”, читаемого вдобавок без предварительной базовой подготовки студентов по промышленной электронике. Даже очень хорошо поставленный курс, уложенный на неподготовленную почву, не даст ожидаемого результата.

Но алогичность просматривается уже в последовательности действий: сначала верстается учебный план как перечень названий без хотя бы краткой расшифровки предполагаемого содержания, а затем по каждому его пункту составляется рабочая программа. Обе работы выполняют специалисты различной квалификации, учебный план – работником административного профиля, программа – исполнителем и они зачастую по-разному понимают скрытое под нерасшифрованным названием содержание. Нормально было бы вначале рассмотреть подробную рабочую программу курса (авторскую преподавателя или взятую в качестве образца уже поставленного ранее в текущем или другом вузе) и по ее содержанию принимать решение о включении или не включении курса в учебный план.

В требованиях к содержанию учебных программ нет перечня знаний и умений, необходимых для восприятия студентами содержания курса – это означает отсутствие требований к поддерживающим дисциплинам, которые должны быть прочитаны на предшествующих этапах обучения и порождает одну из причин упомянутого эклектизма учебных планов.

Конечно, ключевая проблема организации и обеспечения качества обучения – кадровая. Острая нехватка квалифицированных в области кибернетических наук преподавательских кадров была всегда, конъюнктура лишь усилила спрос, не обеспечив его удовлетворение. Квалифицированный программист или электронщик из промышленности, академической или прикладной науки (особенно без ученой степени и звания) не кандидат в преподаватели хотя бы по чисто материальным соображениям. В результате кибернетические предметы (как, к слову, и экономичес-

кие) зачастую преподают люди, не имеющие ни малейшего практического опыта работы в этой области, с дипломами самых разных специальностей. А специфика кибернетических дисциплин требует наличия опыта: непрограммирующий преподаватель программирования – нонсенс, быстро найти ошибку в программе студента (в компьютерном классе их от 12 до 17), помочь в решении даже чисто технической проблемы он не сможет. Вузам остается оснащаться кадрами из внутренних ресурсов – использовать специалистов собственного приготовления и здесь круг замыкается.

В наиболее трудном положении оказываются периферийные вузы “непрестижного” профиля, в частности “трудные” физико-математические факультеты педагогических университетов, потребляющие абитуриентов (выпускников обычных городских и сельских школ) по остаточному принципу – с низким в среднем уровнем школьной подготовки по физике и математике, с неразвитым логическим (алгоритмическим) мышлением, ориентированные на добросовестное заучивание и повторение шаблонных действий при решении задач. Неудовлетворительное качество сырья предопределяет низкое качество выпускаемого продукта, который поступает в школу и готовит на свое место новую волну слабо подготовленных абитуриентов.

Одно из забавных (но не безобидных) следствий кадровой проблемы – терминологические неточности. Введенный в свое время в обиход термин “информатика”, который является всего лишь другим названием кибернетики (науки об оптимальном управлении и математических методах обоснования управленческих решений), породил немало путаницы – еще недавно можно было встретить в университетах параллельно действующие кафедры, например, экономической информатики и экономической кибернетики с полностью совпадающими направлениями работы. К информатике сегодня могут отнести любые компьютерные приложения – от компьютеризированной пишущей машинки (текстовый редактор + принтер) до различных интерактивных оболочек уровня первичной компьютерной грамотности и даже компьютерных игр, – а за подменой понятий следует подмена содержания преподаваемых курсов и соответствующее снижение уровня подготовки специалистов [2].

Возможно, когда-нибудь ключевая проблема преподавательских кадров для кибернетических специальностей будет решена, если преподавать будут параллельно с основной работой активно действующие работники промышленности, академической и прикладной науки и “выработавшие ресурс” специалисты с большим накопленным в практической работе багажом знаний и навыков. Пока же остаются попытки хотя бы маленьких организационных усовершенствований для пусть незначительного повышения результативности преподавательской работы.

Первое предложение в этом направлении нами уже изложено: каждый курс в рабочей программе должен содержать подлежащие выполнению в учебном плане требования к уровню подготовки на предыдущих этапах обучения.

Рассмотрим это предложение на примере условной кибернетической специальности, заключительные курсы которой предполагают изучение методов формирования оптимальных управляющих воздействий и методов идентификации (математического моделирования) управляемых систем (так называемые дуальные кибернетические задачи). Решение обеих задач предполагает (для проверки адекватности модели и оптимальности рассчитанного управления) многократное решение третьей основной кибернетической задачи – задачи анализа, сводящейся, в свою очередь к решению систем дифференциальных (интегро-дифференциальных) уравнений.

Если считать, что десятый заключительный семестр отводится под подготовку выпускной (дипломной) работы, в девятом изучается сопровождаемый лабораторными работами курс теории оптимального управления, в седьмом и восьмом семестрах – методы идентификации дискретных и непрерывных управляемых систем соответственно, то не позднее шестого семестра должны быть завершены следующие поддерживающие курсы:

- методы машинных вычислений, включая методы аппроксимации функций, решения дифференциальных уравнений, линейная алгебра с методами решения проблемы собственных значений (5-й и 6-й семестры);
- теория вероятностей и статистические методы обработки результатов экспериментов (4-й семестр);
- теория планирования экспериментов (5-й семестр);

– общая теория систем автоматического управления и регулирования (6-й семестр);

– математическое программирование, включая теорию массового обслуживания для поддержки методов идентификации дискретных систем (5-й семестр);

В соответствии с этим планом не позднее 4-го семестра должны быть завершены изучением курсы, поддерживающие методы машинных вычислений:

– общий курс математического анализа;

– объектно-ориентированного программирования;

– архитектура компьютеров и операционные системы (2-й семестр);

– фундаментальные алгоритмы и структуры данных (3-й семестр);

– алгоритмические языки и техника программирования (1, 2 семестры);

Мы ограничились планированием общих для всех компьютерных специальностей курсов, не затрагивая многие специальные курсы, зависящие от специализации, например, компьютерные сети, цифровая схемотехника, системное программирование и пр. Не рассматривали также проблемы, связанные с первичной компьютерной грамотностью и офисные пакеты – для них в любом случае не остается ресурса времени, они должны быть решены в довузовской подготовке.

Дефицит времени для определяющих профессиональную подготовку курсов порожден “наступлением” с двух сторон – сокращением общего числа учебных часов в неделю до 30 для создания “щадящего” режима для студентов и перегруженностью учебных планов дисциплинами гуманитарного и общеобразовательного направления, в педагогических университетах дополнительно – психолого-педагогическим комплексом и курсами различных методик преподавания. Все эти дополнения из благих намерений, но все они – за счет снижения уровня профессиональной, предметной подготовки. Трудно оценить эффективность курсов по культуре семейных отношений, валеологии, социологии, делового украинского языка, безопасности жизнедеятельности и многих других, читаемых, естественно, за счет урезания времени на чисто профессиональную подготовку и вполне до-

ступных для самостоятельного освоения всеми желающими. Все же высокоморальное и гуманное общество может быть построено только на основе материального благополучия, а материальные ценности создаются специалистами, в том числе в области высоких технологий.

Один из источников временного дефицита в подготовке специалистов кибернетического профиля объясняется огромной инерционностью, свойственной системе образования. Наиболее ярко он выражен в преподавании компьютерного программирования. Около 15 лет в школе и вузах начала программирования преподаются на учебном языке Паскаль. Как правило, под это отводится пара семестров, затем приходит очередь объектно-ориентированного и системного программирования и оказывается, что Паскаль для этого слишком неудобен, в профессиональном прикладном и системном программировании не используется вообще, что базовым языком для всех операционных систем является язык Си – более простой и наиболее эффективный [1], лежащий в основе всех современных языков сетевого и интернет-программирования, а в прикладном программировании преобладает объектно-ориентированный язык C++. Не меньше половины отведенного для системного и прикладного объектно-ориентированного программирования времени (тоже не более 2-х семестров в сумме) приходится отводить под изучение С и C++, после чего для изучения собственно предметной области времени не остается – результат легко предсказуем. Мы не против Паскаля как средства любительского программирования “для себя”, но использовать его в профессиональной подготовке специалистов в области информатики (кибернетики) с неизбежным последующим отбрасыванием как ненужного – в нынешних условиях слишком расточительно.

Как хороший филолог должен владеть несколькими языками, так и специалист в области кибернетики должен владеть несколькими языками программирования. Дав в качестве базового простой язык Си, не стоит жестко определять, какие еще языки должен знать будущий специалист – выбор инструмента (языка) определяется особенностями предметной областью, а не нормативными актами (учебными планом). Так, в курсе архитектуры компьютеров и операционных систем целесообразно дать один

из языков ассемблера, курс объектно-ориентированного программирования проиллюстрировать примерами из Смоллток, при изучении систем искусственного интеллекта куда естественней дать Лисп и Пролог, чем использовать язык общего назначения, а в курсе сетевого программирования можно познакомиться с Явой.

#### Литература:

1. Полищук А.П., Семериков С.А., Грищенко Н.В. О выборе языка программирования для начального обучения. / Збірник наукових праць «Комп'ютерне моделювання та інформаційні технології в природничих науках». – Кривий Ріг, 2000.

2. Семеріков С.О. Методичні основи вивчення теми “Операційні системи” у підготовці майбутнього вчителя // Рідна школа. – 2003. – №1. – С. 44-45.

# ПРОГРАММА АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ И КОНТРОЛЯ УЧЕБНОЙ ДЕЯТЕЛЬНОСТИ

А.Л. Поляничко

г. Киев, Киевский национальный университет технологий и дизайна

Контроль знаний эмпирическими методами, как известно, обладает рядом недостатков, главным из которых можно считать субъективность оценок. Это приводит к тому, что у нас отсутствует возможность реалистически подойти к объективности оценок. При использовании дидактических процессов нет возможности их совершенствования и путей улучшения.

Это, конечно, не означает, что эмпирические методы контроля знаний обучающегося следует отбросить, как непригодные для использования. Устные методы контроля можно применять для работы со студентами на семинарах, лабораторных и практических занятиях, при проведении бесед, конференций.

Для проверки результатов теоретического и практического усвоения студентами учебного материала на кафедре информатики Киевского национального университета технологий и дизайна автором разработана система компьютерного контроля и диагностики качества занятий. В рамках этой системы тест представляет собой набор из  $N$  заданий приблизительно одинаковой сложности. Результат выполнения каждого задания оценивается по бинарной шкале. Если студент успешно выполнил  $i$ -ое задание, то оно оценивается в один балл ( $p_i=1$ ) и в противном случае – ноль ( $p_i=0$ ). Появление  $i$ -го задания является случайным и не зависит от номера и числа заданий. Поскольку тест имеет ограниченную длину ( $N$  заданий), то в результате тестирования получается выборочная оценка истинного уровня обучения  $p_iN$ .

С целью получения объективности оценки, при степени точности  $\varepsilon=0,05$ , надежности  $d=0,9$  и истинном уровне обученности  $p=0,7$  теоретическая длина теста, выбрана по данным таблицы, полученным расчетным путём:



$d/p$	0,5	0,6	0,7	0,8	0,9
0,8	169	162	142	108	61
0,9	272	261	<b>228</b>	179	98

Автор принял длину теста  $N=200$  заданий, разбив его на два термина, и использовался для проверки знаний студентов первого курса по дисциплине «Информатика».

При разработке контрольных заданий в тестах использовались такие педагогические приёмы, как

- установление недостающих в данном задании группы слов;
- вставка в определенные места многоточия недостающего слова;
- выбор правильных (неправильных) ответов из предложенных;
- возможность правильного выбора утверждений;
- конструирование и ввод с клавиатуры правильного словесного ответа, состоящего из ключевых слов или формулы.

Шкала оценок была взята в пределах 100 условных единиц и разбита на четыре интервала: оценка «отлично» – 88–100 единиц; «хорошо» – 69–87; «удовлетворительно» – 50–68 и «неудовлетворительно» – 0–49. Представив изменение уровня формальных знаний по приведенной 100-бальной шкале в виде линейной шкалы, получим четыре оценочных интервала. Как видим, области не соприкасаются и оценки принадлежат каждой из них из-за своей целочисленности.



В средних точках оценочного интервала получили дискретный уровень формальных оценок в четырехбальной системе.

При этом заштрихованные участки соответствуют отклонению формальной оценки от фактического уровня знаний на каждом оценочном интервале. Особенно критичной является оценка на границах интервалов. Кроме этой узкой зоны, имеется более широкая зона – «неполной уверенности в оценке». В пределах этой зоны имеет место завышение или занижение оценки вследствие нивелирования. Откуда видна проблема «спорных» оценок при недоборе нескольких баллов студентом при тестировании.

Приведенные Беспалько В.П. в работе «Методы и критерии оценок знаний и умений при программированном опросе» наблюдения привели к заключению, что вероятности «уверенных» оценок в использованном оценочном интервале могут быть выражены нормальным законом распределения, а вероятность случайных величин могут быть определена по уровню 0,85 от оценочного интервала. Например, в оценочном диапазоне «удовлетворительно» интервал составит 18 условных единиц, а полужона «неполной уверенности» определяется в  $\pm 8$  условных единиц ( $18/2 \times 0,85$ ). По 100-бальной шкале это интервал ( $+66 \Leftrightarrow 73$ ) условных единиц и в четырёхбальной системе соответствует  $3 \times 0,85 = 3,4$  балла. Возникает проблема, какую оценку выставить: «удовлетворительно» или «хорошо»?

В разработанной программе тестирования знаний студентов в зоне «неполной уверенности» производится уточнение с помощью корригирующих дополнительных вопросов, которые генерирует программа. Так, получив за тестирование 66 баллов (зона «неполной уверенности»), студенту для оценки «хорошо» программа предлагает 8 дополнительных заданий, которые он должен полностью выполнить. Количество и качественные характеристики вопросов зависят от разброса.

Тестовая программа написана на языке VBA MS Excel. Тестовые задания размещены в соответствующих ячейках таблицы (текст, схемы, рисунки). В столбце А размещены все тестовые задания. В блоке столбцов (В:F) – варианты ответов на эти задания (на некоторые задания предлагается несколько правильных ответов), или варианты недостающих слов, или набор утверждений, или набор ключевых слов и т.п. В столбце G – коды правильных вариантов решения задания. В столбце H – признак характера педагогического приёма, связанного с отображением на

экране соответствующей формы. Форма содержит текстовые окна для отображения задания, окна для ввода ответа с клавиатуры, окна с вариантами выбора правильного ответа, причем правильных ответов может быть несколько. При ошибочном (неправильном) ответе выводится стандартное диалоговое окно с правильным ответом и ошибочным ответом студента. В форме также помещены таймер, показывающий время, оставшееся до завершения тестирования, и индикатор количества правильно выполненных заданий.

Статистические данные, полученные при тестировании знаний студентов, позволяют определить, какие разделы дисциплины требуют корректировки при их последующем изложении. На основании данных дидактико-статистического анализа определяются те разделы курса, на которые правильно ответило только 30-40% студентов. Эти разделы требуют дополнительной переработки при изложении учебного материала либо изъятия этих заданий из теста.

# АКТИВІЗАЦІЯ ПІЗНАВАЛЬНОЇ ДІЯЛЬНОСТІ СТУДЕНТІВ ПІД ЧАС ВИВЧЕННЯ ІНФОРМАТИКИ ЗАСОБАМИ ПІДРУЧНИКА

Н.І. Праворська

м. Хмельницький, Хмельницький інститут економіки та підприємництва

Необхідною умовою повноцінного засвоєння студентами знань є використання прийомів активізації пізнавальної діяльності. Психологічний аналіз їх показав, що всі прийоми тією чи іншою мірою впливають на характер змістовної обробки навчального матеріалу. Використання цих прийомів забезпечує міцне і ефективне запам'ятовування матеріалу. Наприклад, одним із прийомів є використання у підручнику ілюстрацій. Ефективність наочності пов'язана з особливостями мислення людини, яке базується на зорових образах. При вивченні матеріалу увага студента своєчасно переключається із одержаної про об'єкт інформації на засоби наочності і навпаки, що суттєво зменшує непродуктивну діяльність студента.

Предметна галузь інформатики, на відміну від інших дисциплін, змінюється дуже динамічно. Необхідно часто змінювати навчальні плани, програми, навчально-методичне забезпечення. Тому до важливих принципів, на яких ґрунтується розробка навчальних посібників з інформатики для вищих навчальних закладів, слід віднести:

- принцип єдиної методичної основи як при вивченні інформатики, так і інших дисциплін на базі інформатики (теоретична інформатика, Internet, CorelDraw, математичні системи типу Maple і інші);

- розділи підручника, які мають теоретичний та методологічний характер, закінчуються контрольними запитаннями, а розділи, які мають практичний характер, закінчуються вправами та завданнями для типових розрахунків, курсових робіт, дослідницької роботи тощо.

Формування змісту навчального посібника з інформатики повинно ґрунтуватись на принципі інтеграції навчальних програм інформатики та інших дисциплін, які входять до навчально-

го плану. Тобто, необхідним є такий методичний підхід. Конкретна дисципліна надає систему “задача – методи”, а інформатика забезпечує систему “засоби – прийоми” [1]. Використання між-предметних зв’язків дає можливість один навчальний предмет наповнити елементами та фактами іншого. Наявність у підручнику такої інформації відіграє важливу роль у формуванні цілісної уяви про предмет та системи наукових фахових знань. Прикладом може бути експериментальний навчальний посібник “Інформатика – 7” [2].

Цих принципів було дотримувалася автор під час підготовки навчального посібника “Інформатика та комп’ютерна техніка” [3].

Важливими засобами є також формування діяльності аналізу структури об’єктів завдання. Якщо в навчальному посібнику наводяться лише алгоритми розв’язування різних класів задач, то студент не завжди розуміє сутність способів одержання результату. А тому доцільно поряд з алгоритмами надати студентові засоби орієнтації у знаковому середовищі та засоби переробки знакової інформації, яка притаманна таким дисциплінам, як математика і інформатика [4]. Система завдань повинна супроводжуватись поясненнями методологічного характеру, в яких розкриваються загальні принципи застосування сучасних інформаційних технологій у всіх сферах людської активності.

Засоби підручник дозволяють поєднувати в собі навчальний матеріал для студентів з різним рівнем знань і розвитку. Вибір навчального матеріалу пропонується здійснюватись у відповідності до принципу провідної ролі теоретичних знань з навчального предмета.

Психологічною основою створення підручника є організація навчально-пізнавальної діяльності студентів, яка передбачає:

- порівняння за взірцем, оволодіння репродуктивним методом розв’язування типових задач;
- компіляцію типових розв’язків, пошук нових варіантів методів розв’язування задач;
- набуття навичок творчої діяльності при розв’язуванні нових і нетипових задач.

Тому у навчальному посібнику [3] розроблена система завдань, за допомогою якої студенти оволодівають знаннями під

час виконання лабораторних робіт та самостійному вивченні окремих розділів курсу. Після виконання всіх запропонованих завдань студенти набувають відповідних навичок, які необхідні спеціалістам економічних спеціальностей, а саме:

- вміння працювати з операційною системою Windows – створювати файли та папки, проводити зміни в створених файлах та папках; знищувати та відновлювати файли, копіювати об'єкти системи; користуватись пошуковою і довідковою системою; проводити настройки системи і т. ін.;
- працювати з текстовим редактором Microsoft Word – створювати файли документів, проводити необхідні редагування та форматування готових файлів; створювати будь-які таблиці; використовувати вставку формул, малюнків; створювати різноманітні графічні ефекти, тощо;
- вміти використовувати електронну таблицю Microsoft Excel – створювати та вносити дані в робочі книги; проводити необхідні розрахунки (як прості, так і з використанням майстрів); будувати будь-які діаграми по розрахованим даним, редагувати вже готові діаграми; використовувати електронні таблиці як бази даних і т. ін.
- працювати з системою управління базою даних Microsoft Access – створювати таблиці в базі даних; редагувати їх; зв'язувати декілька таблиць та створювати запити і вибірки по запитам; будувати форми та звіти по створеним таблицям, тощо;
- мати навички роботи з глобальною міжнародною мережею Internet – налагоджувати персональний комп'ютер для роботи в Internet; виконувати пошук інформації по заданій адресі; працювати з пошуковими системами та різними службами мережі; працювати з електронною поштою і т. ін.

Тобто до посібника включено основні програмні продукти, які необхідно знати всім висококваліфікованим спеціалістам з різних профілів. Навички здобуті при вивченні дисципліни “Інформатика та комп'ютерна техніка” слугуватимуть не тільки під час вивчення інших дисциплін, таких як “Економіко-математичні методи та моделі технологічних процесів”, “Інформаційні системи підприємств” та ін., а й у подальшій роботі.

Системний підхід до формування змісту навчального посібника передбачає також використання структури і змісту предметних, наукових знань, методології пізнання і практики. Тому прийнято три рівні побудови навчального посібника: на вищому рівні розглядаються основи інформатики, фундаментальні твердження наукових знань, які передаються в поняттях, фактах, висновках, наслідках; на рівні безпосереднього спостереження та на рівні перевірки і використання знань.

При вивченні матеріалу підручника однією з найважчих проблем є сформованість достатньої пізнавальної мотивації, тобто інтересів, стимулів навчання. Стійкий і тривалий інтерес до предмету автор підручника створює за допомогою проблемних ситуацій. Тому ми вважаємо, що навчальний текст необхідно структурувати так, щоб у ньому була достатня кількість рівномірно розподілених характерних проблемних ситуацій і рекомендацій щодо їх розв'язання. Пізнавальну мотивацію стимулює наявність матеріалу, який, принаймні, досить стисло розкриває нові, ще не розв'язані проблеми.

Рівень пізнавальної мотивації суттєво залежить і від складності навчального тексту. Тому у навчальному посібнику оптимальною прийнято таку структуру тексту, коли його головне змістовне ядро знаходиться на початку тексту.

Такий підхід дозволяє залучати студента до прочитання, розбирання, структурування навчального матеріалу. Головною метою навчальної діяльності при цьому є набуття умінь розуміти текст, виділити і усвідомити постановку типових проблем і раціональних зразків діяльності. При самостійній роботі над текстом студент не завжди сприймає ідеї, методи, схеми доведень, логічні ланцюги міркувань, алгоритми, загальні підходи тощо. Самостійна робота над текстом дозволяє підвищити ефективність оволодіння матеріалом за допомогою техніки правильного читання, яка полягає у попередній орієнтації в навчальному матеріалі, швидкому перегляді тексту, поглибленому читанні з змістовним аналізом і конспектуванням.

Аналіз методичної літератури щодо змісту вправ і задач висвітлює такі недоліки:

- недостатня розробленість структури системи задач і методики її формування;

- майже всі дослідження виконано на матеріалі курсів математики і фізики загальноосвітньої школи.

А практика навчання показує, що в діючих підручниках з інформатики переважає традиційна організація підбору задач – прагнення розв’язати їх зі студентами якнайбільше без врахування навчаючих якостей; використання вправ переважно для безпосереднього закріплення знань чи їх повторення; не використовуються завдання, які дозволяють формувати в студентів навички аналізу, синтезу, узагальнення, абстрагування, моделювання.

Характерний недолік структури системи задач в багатьох навчальних посібниках – ізольованість їх одна від одної, відсутність інформаційної спільності між ними, порядок розв’язування майже довільний і відноситься до компетенції викладача або відсутній. Набір таких задач не забезпечує міцність і свідомість засвоєння знань. Варто відмітити, що при систематизації вправ в більшості випадків враховується лише збільшення кількості задач без якісних змін їх структури. Таким чином, актуальною на сьогодні є розробка особистісно-орієнтованої системи задач, побудованої на основі диференціації навчання, яка враховувала б психолого-методичні закономірності формування знань, з інформатики особливості навчальної діяльності студентів, різнорівневі вимоги до їхньої підготовки.

Специфіка задач з інформатики проявляється перш за все у вузькій їх спрямованості. Значну їх частину становлять задачі на вивчення операцій, відношень між ними і їх властивостей. Однак, з дидактичної точки зору в процесі розв’язування задач з інформатики закладені більші можливості розвитку мислення, просторової уяви студентів. Характерна особливість курсу інформатики – широкий спектр конфігурацій, що використовуються в задачах.

Правильне вирішення питання про поняття системи задач, їх послідовність, різноманітність, типи і вимоги, методику їх розв’язання є однією з важливих умов корінного вдосконалення теорії і практики навчання. Систематизація в значній мірі визначає якість навчання інформатиці. На жаль, поки що немає чітко встановлених принципів, які б дозволили судити про те, що саме повинно бути досягнуто з допомогою задач і якої складності во-



ни повинні бути, в якому порядку розміщуватись у підручнику.

Наведемо приклад завдання, яке входить до складу розробленої в посібнику системи завдань [3].

З метою визначення рівня підготовки студентів на перших заняттях практикуються нульові контрольні роботи.

### **Нульова контрольна робота.**

Нульова контрольна робота проходить в три етапи:

**I етап** – тестовий – студент повинен відповісти на запитання тесту, щоб виявити рівень теоретичних знань по питанням з тем “Операційна система MS DOS”, “Операційна оболонка NC”, “Операційна система Windows 95”

**II етап** – виконати елементарні дії:

створити каталог (папку); в каталозі (папці) створити файл, відредагувати його зміст; перейменувати файл; знищити файл, відновити його і т. ін. в операційній системі MS DOS, операційній оболонці NC, операційній системі Windows.

**III етап** – 1) виконати в операційній системі MS DOS дії: передивитись вміст файлу, який треба знайти по вказаному викладачем шляху; скопіювати вказаний файл на диск A під іншим ім'ям; знищити файл, враховуючи що він знаходиться в каталозі.

2) виконати дії в операційній оболонці NC: файл, який знаходиться на диску A, заархівувати та помістити у місце по вказаному викладачем шляху; файл на новому місці розархівувати та передивитися його вміст; змінити вміст файлу та скопіювати його на диск A з новим ім'ям, яке буде вказано викладачем.

3) виконати дії в операційній системі Windows: скопіювати файл, який знаходиться на диску A, по вказаному викладачем шляху, вказаним способом (використовуючи рядок меню або панель інструментів, контекстне меню або метод перетягування, або використовуючи комбінований спосіб); знайти довідкову інформацію про вказане викладачем питання; знайти за допомогою пошукової системи, вказані викладачем файли; провести вказані викладачем налагодження операційної системи.

Досвід показує, що розумінню навчального матеріалу перешкоджає низькій рівень попередньої підготовки студента. Йому не завжди був зрозумілим зміст тих понять та міркувань, які вивчались раніше. Виділимо складові частини процесу розуміння навчального тексту. Перш за все студент повинен розуміти зміст

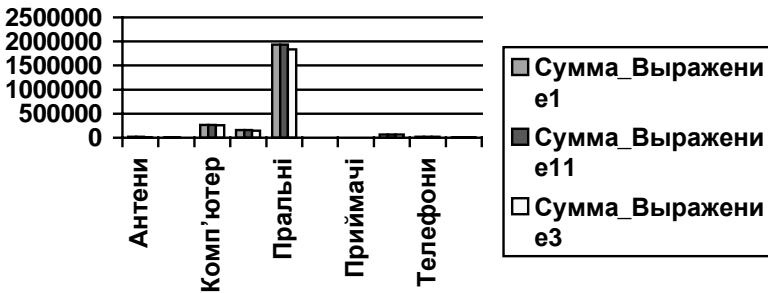
окремих слів та словосполучень. Тому у навчальному тексті і, частково, у навчальній програмі повинен бути реалізований принцип надання максимально можливих пояснень до понять, які використовуються.

*Системне завдання MS Access*

1. Набрати таблицю в базі даних MS Access за зразком.
2. Провести розрахунки , використовуючи *Построитель вы-  
ражений* при створенні запитів: провести обчислення суми продажу продукції за кожний період; обчислити відрахування – якщо сума продажу перевищує 60000 грн., то відрахування складають 5% від всієї суми за період, інакше 3%; розрахувати всю суму продажу з урахуванням відрахувань.
3. Створити діаграму 3 згідно зразка.
4. Провести аналіз чим відрізняється процес обчислення в програмах MS Word та MS Excel від MS Access. Вказати недоліки та переваги MS Excel.
5. Провести аналіз процесу побудови діаграм в програмах MS Word та MS Excel від MS Access. Вказати недоліки та переваги MS Excel.

№п/п	Назва товару	Сума за березень, грн.	Сума за квітень, грн.	Сума за травень, грн.	Сума за період, грн.	Відрахування	Всього
1.	Телевізори	10356	51256	10251			
2.	Телефони	2512	10123	4251			
3.	Праски	1362	523	2531			
4.	Годинники	1223	2556	1254			
5.	Комп'ютери	125236	80145	65236			
6.	Приймачі	1345	234	675			
7.	Магнітофони	24567	56567	78767			
8.	Антени	1577	11123	2313			
9.	Пральні машини	567678	909876	456789			
10.	Гостер	3456	4567	1234			

## Сист\_завд Запрос



Такі якості знань, як глибина і повнота, системність, узагальненість, міцність і інші, засобами підручника можуть бути сформовані за допомогою пояснювально-ілюстративного подання матеріалу. Проте такий підхід не дозволяє на достатньому рівні сформувані такі характеристики знань, як усвідомленість, оперативність та інші.

Із загальнопсихологічної точки зору, вивчати напам'ять необхідно базові елементи всіх рівнів системи семантичної пам'яті, які складають її структурну основу переробки образної і семантичної інформації. Без вивчання напам'ять цих елементів неможливо сформувати повноцінну систему довготермінової семантичної пам'яті. До базових елементів відносяться поняття і закони, які зв'язують суттєві ознаки цих понять. З кожного розділу, теми вказують такі базові елементи (алфавіт), а також орієнтувальну основу дій (її ще називають функціонально-інформаційною системою).

В теперішніх умовах розв'язання проблеми забезпечення навчальними та методичними матеріалами для проведення аудиторних занять, організації самостійної роботи з інформатики вимагає від кафедр розробки та видання таких дидактично вивірених матеріалів. Оскільки у сучасних економічних умовах відсутні можливості придбання навчальної літератури, яка до того ж не завжди є якісною, що відзначалося вище. Крім того, література з інформатики старіє на протязі короткого часу, що пов'язано із

швидкими темпами розвитку комп'ютерних технологій, систематичним розширенням сфери використання комп'ютерів, систематичним впровадженням обчислювальних засобів у навчальний процес, і, відповідно до цього, із змінами робочих навчальних програм з інформатики.

Навчальні посібники з інформатики мають носити концептуально-методологічний характер, розглядати вузлові, проблемні питання, визначати напрямки, методи та засоби самостійної роботи студентів.

Формування змісту навчального посібника з інформатики повинно ґрунтуватись на принципі інтеграції навчальних програм інформатики та інших дисциплін, тобто, на основі такого методичного підходу. Конкретна дисципліна надає систему “задача – методи”, а інформатика забезпечує систему “засоби – прийоми”. Використання міжпредметних зв'язків дає можливість один навчальний предмет наповнити елементами та фактами іншого, що дозволяє суттєво підвищити рівень вивчення як інформатики, так і інших навчальних дисциплін, активізувати навчально-пізнавальну діяльність студентів.

Підсилення практичної спрямованості навчання інформатиці визначається як швидким розвитком обчислювальних засобів, так впровадженням в різні галузі науки, виробництва, а тому збільшується кількість користувачів, які не є фахівцями у питаннях розробки та експлуатації обчислювальних засобів. Розширення можливостей комп'ютерів призводить до нових, складніших технологій їх практичного використання та зменшення необхідності вивчення структури комп'ютера, що надає можливість викладачеві перерозподілити термін навчання інформатиці студентів технічних економічних і інших спеціальностей з акцентом на питання практичного застосування обчислювальних засобів. А створення та широке впровадження перспективних інформаційних технологій, які ґрунтуються на ідеях штучного інтелекту, призведе до подальшого підсилення практичної спрямованості курсу з інформатики.

Знання та вміле застосування основних специфічних закономірностей навчання інформатиці сприяє методично правильній побудові процесу навчання, правильно відібрати зміст навчального матеріалу для проведення аудиторних занять і організації

самостійної роботи студентів, оперативно вносити зміни у зміст навчання в міру розвитку засобів інформаційних технологій.

#### Література

1. Информатика. Базовый курс. Учебник для вузов / Симонович С.В. и др. – СПб: Питер, 2001. – 640 с.
2. Жалдак М.І., Морзе Н.В. Информатика – 7. Експериментальний навч. посібн. для учн. 7 кл. ЗОШ. – К.: ДіаСофт, 2000. – 208 с.
3. Праворська Н.І. Информатика та комп'ютерна техніка: Навчально-методичний посібник для студентів вищих навчальних закладів. – Хмельницький, 2002. – 312 с.
4. Формирование учебной деятельности студентов/ Под ред. В.Я. Ляудис. – М.: Изд-во Моск. ун-та, 1989. – 240 с.

## **ЭКСПЕРТНО-ОБУЧАЮЩИЕ СИСТЕМЫ РЕЙТИНГА ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ НА ОСНОВЕ КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ ОБУЧЕНИЯ**

Ю.Ф. Рева, М.А. Кислова

г. Кривой Рог, Технологический факультет Института экономики  
и новых технологий

Как известно, компьютерные технологии обучения позволяют осуществить разработку экспертно-обучающих систем оценки знаний, умений и навыков, в основу таких экспертных систем должны быть положены принципы теории поэтапного формирования умственных действий и умений. Среди этих принципов выделяют следующие:

- 1) переход к планированию учебного процесса в соответствии с уровнем усвоения знаний;
- 2) введение в учебный процесс количественного измерителя степени законченности процесса обучения в виде коэффициента усвоения;
- 3) экспертно-обучающая система оценки знаний, умений, навыков должна создаваться с учетом двух выше названных принципов.

Создание экспертно-обучающих, экспертных систем по оценке качества усвоения знаний и завершенности процесса обучения предполагает прежде всего учет основополагающих закономерностей:

- 1) изменение роли и функции преподавателя, превращение его в специалиста-консультанта, что добавляет новую обязанность в его преподавательской деятельности;
- 2) отказ от поточного метода обучения и перехода к индивидуальной подготовке специалиста;
- 3) перенос центра тяжести учебного процесса на самостоятельную работу студентов;
- 4) подготовка учебно-методического комплекса на основе учета особенностей компьютерной технологии обучения. Каждый студент обеспечивается полностью пособиями и многовариантными заданиями по дисциплине;
- 5) отказ от традиционных форм контроля и внедрение индиви-

дуального кумулятивного индекса, в котором резко возрастает роль текущего, рубежного и итогового контроля знаний, умений и навыков.

Если вышеназванные принципы строго выполняются, то можно говорить о наличии возможностей разработки и использования в учебном процессе экспертно-обучающих систем и системы экспертной оценки усвоения знаний, умений и навыков.

Экспертно-обучающая система состоит обычно из двух независимых частей:

- 1) универсальная программа-оболочка, поддерживающая интерфейс общения со студентами, содержащая подсистему логического вывода и не зависящая от содержания конкретной дисциплины;
- 2) базы знаний конкретных дисциплин, содержащих описание основных объектов, используемых в данной дисциплине, логические правила классификации задач, решаемых в данной дисциплине, описание конкретных методов и примеров решения задач, определения и примеры, помогающие студенту в правильном выборе конкретной задачи.

В свою очередь, экспертная система оценки уровня усвоения знаний и умений должна быть адаптирована к экспертно-обучающей системе, базе знаний, но программа-оболочка должна содержать также квалиметрические параметры оценки усвояемости (шкалу оценок), нормативные коэффициенты, определяющие степень законченности обучения в соответствии со специальностью, и программное обеспечение, обеспечивающее выдачу протокола результатов общения ЭВМ и студента с определением индивидуального коэффициента усвоения знаний.

По мнению ведущих специалистов, тестирование должно быть измерением качества усвоения знаний, умений и навыков. Сравнение правил выполнения задания (задачи), предложенного в тексте, с эталоном ответа позволяет определить коэффициент усвоения знаний (**Kus**). Следует заметить что  $Kus = A/P$ , где **A** – число правильных ответов, а **P** – число заданий в предлагаемых тестах.

Определение **Kus** является операцией измерения качества усвоения знаний. **Kus** поддается нормировке ( $0 < Kus < 1$ ), процедура же контроля усвоения легко автоматизируется. По коэффи-

циенту судят о завершенности процесса обучения, если  $K_{us} > 0.7$ , то процесс обучения можно считать завершенным. При усвоении знаний с  $K_{us} = 0.7$  студент в профессиональной деятельности систематически совершает ошибки и неспособен к их исправлению из-за неумения их находить. Нижнюю допустимую границу окончания процесса обучения повышают до величины, необходимой с точки зрения безопасности деятельности. Это относится к работе водителя, механизатора, повара и т.д.

Вместе с тем компьютерное обучение требует серьезного психоэмоционального обеспечения. Имеются случаи конфликтов, эмоциональной напряженности, монотонии, утомления. В связи с этим обязательно необходима разработка рекомендаций по психологической готовности к работе с компьютерной техникой, особенно в диалоговых системах. Сейчас известно, что эффективность компьютерного обучения немыслима без учета индивидуальных психодинамических особенностей обучаемого.

В работах по исследованию проблемы применения ЭВМ в процессе обучения даются результаты исследований из области компьютерных технологий обучения. Например, групповое дифференцированное обучение экстравертов и интровертов выявило своеобразные стили компьютерной деятельности и высокую эффективность по сравнению со смешанным обучением. При разработке сценариев к диалогам учитывалась интеллектуальная нагрузка учебных программ и в зависимости от этого составлялись эмоциональные сценарии каждого занятия. В эти сценарии включались элементы логических или моторно-динамических игр, что способствовало поддержанию положительного эмоционального тонуса и высокой работоспособности.

При работе с компьютером оптимально включены все каналы восприятия, произвольное внимание, достаточно высок уровень оптимального возбуждения, комфортность процесса познания максимальна, процесс обычно подпитывается энергией обучаемого. В связи с этим предлагается учитывать основные личностные характеристики обучаемых: темперамент, особенности эмоционального реагирования, тип межличностного взаимодействия, особенности протекания познавательных психических процессов, интеллектуальный потенциал и т.д. Учет личностных характеристик также необходим для сглаживания или предупре-



ждения отрицательных сторон компьютерного обучения. Сюда следует отнести вопросы отчуждения, неравные условия обучения, снижение роли письменной речи, ослабление творческого мышления, потеря чувства реальности и т.д. се это говорит о том, что компьютеризация обучения одна из сложнейших задач, решение которой должно осуществляется на принципах системного подхода. В качестве первоочередных должны разрабатываться экспертные системы оценки усвоения знаний.

Экспертные системы усвоения знаний требуют специальных программных оболочек, которые, будучи приспособленными для того, чтобы их мог заполнять преподаватель, не являющийся специалистом в области программирования, делает его автором автоматизированного учебного курса и втягивают их в работу со средствами компьютерной технологии обучения.

Для определения степени обучаемости студентов по каждой учебной дисциплине выделяют объем знаний, который необходим для усвоения согласно учебной программе, что составляет базовый объем знаний. Базовые знания представляют минимум государственного образовательного стандарта. Но и среди базовых знаний выделяют те, которые должны оставаться в памяти по любой дисциплине. Выделяют несколько звеньев знаний: базовые знания, программные знания, сверхпрограммные знания. Педагогические тесты – единственный инструмент, позволяющий не только измерить обученность, но и умение использовать знания. Если говорить только об умениях, то на всех уровнях усвоения знаний можно выделить четыре вида умений:

- 1) умение узнавать объекты, понятие, факты, законы, модели;
- 2) умение действовать по образцу, по известному алгоритму, правилу;
- 3) умение проводить анализ ситуации, вычленять главное и строить из освоенных операций процедуры, позволяющие получить решение тестового задания;
- 4) умение и способность находить оригинальные решения.

Эти умения не противоречат теории поэтапного формирования умственных действий, в основу которой и положен метод разработки автоматизированного тестирования с целью оценки усвоения знаний, приобретения умений и навыков. Это позволяет создать не только экспертные системы оценок степени обу-

чаемости студентов, но и построить гибкую динамичную рейтинговую систему контроля знаний.

Одной из особенностей компьютерной технологии обучения является возможность управлять процессом усвоения знаний на основе четкой систематизации и структуризации курса. Этот подход позволяет заложить в каждую составную часть учебной программы весовой коэффициент и на этом построить системный подход к оценке знаний.

Структурно-логический подход к содержанию обучения, а затем и систематизация и структуризация предмета, по мнению специалистов, способствует следующему:

- формированию у студентов системных знаний;
- повышению объективности самооценки и оценки знаний;
- возможности более объективного и глубокого анализа степени усвоения отдельных фрагментов учебной программы.

Попытки индивидуализировать процесс обучения в традиционной методике приводят только к интенсификации труда преподавателя. При структурировании и логическом анализе содержания обучения, выделении учебных элементов, постановке дидактических целей обучения с ориентировкой на конкретные учебные элементы индивидуализации обучения становится реальной. Путем реализации идеи партнерства студента и преподавателя, во время индивидуальных консультаций создаются ситуации, способствующие развитию творческих и индивидуальных способностей студентов.

В соответствии с выделенными учебными элементами, заранее установленными дидактическими целями, можно автоматизировать процесс объективной и непрерывной оценки знаний. Оценка результатов обучения играет определенную роль в корректировке и направлении результата обучения в соответствии с поставленными целями. В этом случае оценка знаний становится эффективным инструментом повышения учебно-познавательной активности студента. Появляется возможность самоконтроля знаний и разработки экспертно-обучающей, а затем и рейтинговой систем контроля знаний. Создание экспертно-обучающих и рейтинговых систем контроля знаний в какой-то мере помогает выбрать направления в решении еще одной из важнейших проблем – в выработке единого подхода к оценке профессионализма

выпускника учебного заведения. В настоящее время единственной объективной оценкой качества подготовки специалиста является их оценка предприятиями и организациями. Этот метод неприемлем для использования в процессе подготовки специалиста, поэтому стали разрабатываться целые учебно-методические комплексы управления качеством подготовки, включающие в себя следующие задачи:

- формирование эталонов качества подготовки специалистов;
- разработка средств контроля на базе эталонов качества;
- разработка, проведение процедуры сравнения достигнутого уровня подготовки с эталоном качества.

Выработка системы управляющих воздействий условиями и факторами, определяющими достигнутое качество, с целью минимизации обнаруженных отклонений.

Введение компьютерных технологий обучения привлекли педагогов к поискам объективных измерителей оценки уровня усвоения знаний умений и навыков. В качестве педагогических новаций предлагаются тесты как инструмент проверки соответствия требований к подготовке выпускников заданным стандартам знаний и выявлению пробелов в знаниях.

Тесты в сочетании с компьютерными технологиями обучения помогают перейти к созданию экспертных систем оценки знаний, умений и навыков.

Контроль выполняет свою функцию только тогда, когда он основан на непредвзятом подходе, объективности. Если контроль осуществляется человеком, то он всегда несет в себе влияние этого человека и его отношение к проверяемому. Использование рейтинг-контроля на базе применения ЭВМ позволит устранить эти негативные факторы и проверить знания студентов вне зависимости от “человеческого фактора”.

Для четкого функционирования рейтинговой системы требуется:

- разработка логических структур содержания обучения по всем темам;
- разработка перечня учебных элементов;
- разработка тестов в соответствии с перечнем учебных элементов и поставленными дидактическими целями;
- разработка отдельных фрагментов сценариев по работе с тес-

- тами разных уровней усвоения знаний;
- написание сценариев с учетом психофизиологических особенностей обучаемых;
  - разработка макета протокола занятия с подведением полученных результатов;
  - выбор программы-оболочки, позволяющей реализовать программы-сценарии.

Реально сейчас ни один пункт не обеспечен методической литературой и учебными пособиями, поэтому говорить о полном внедрении данной системы контроля знаний пока рано.

Использование экспертно-обучающих систем рейтинга знаний позволяет сократить в большинстве случаев время на выяснение подготовленности студентов к занятиям; заинтересованность студентов в максимально возможной для них рейтинговой оценке настраивает их на добросовестную работу в процессе подготовки к занятию.

Подготовленность же к занятиям тех студентов, которые смирились с тем, что не получают оценку по рейтингу, можно проверять в индивидуальном порядке, не сокращая для большей части студентов время, выделяемое на самостоятельную работу. Это способствует, с одной стороны, отходу от традиционных “школярских” методов работы, а с другой – позволяет при непрерывном контроле оказывать большее доверие к студенту, не подвергая изначально сомнению факт его подготовки к занятию.

Рассматриваемая система позволяет получать достаточно объективную информацию о степени успешности обучения студентов относительно друг друга. Уже по истечении двух-трех месяцев можно выделить лучших и худших студентов группы. Это дает администрации мощный рычаг, позволяющий поощрять лучших и наказывать худших.

Помимо этого, уже на раннем этапе формируются массивы студентов по прогностическому показателю: претендентов на “отлично”, “хорошо”, “удовлетворительно” и тех студентов, которые отстают от учебного плана и могут остаться не аттестованными. Ранний прогноз позволяет внести корректировку в дальнейшее обучение.

На первый взгляд может показаться, что студенты, набравшие определенную сумму баллов, обеспечивающую подходя-

шую оценку, могут перестать заниматься. Но, в основном, происходит срабатывание механизма соревновательности в обучении. Студент, занявший определенное место в групповом табель-рейтинге, не хочет перемещаться вниз, так как это воспринимается как его личная неудача.

Введение такой системы контроля знаний в значительной степени устраняет негативные стороны уравнилельной системы обучения. В результате исчезают усредненные группы отличников, хорошистов и т.д. Вместо них появляются “первый”, “пятый”, ”сотый”. Использование рейтинга позволяет также снижать возможность получения незаслуженной (случайной) оценки по изучаемой теме, поскольку результирующая оценка учитывает работу студента в течение полугодия. Что же касается баллов, выставляемых за реферат, участие в олимпиаде и т.д., то они определяются только коллегиально с учетом мнения как можно большего числа преподавателей.

Использование экспертно-обучающих систем рейтинга знаний приводящих к состязательности в процессе обучения, существенно повышает стремление студентов к приобретению знаний, что приводит к повышению качества подготовки специалистов.

Предварительные итоги использования рейтинговой системы оценки знаний студентов показывают, что студенты стали работать систематически.

Опыта работы с экспертно-обучающими системами рейтинга знаний еще практически нет, поэтому они непрерывно изменяется и дорабатывается. Делается это с учетом анкетирования студентов.

Существенное различие экспертно-обучающая система рейтинга знаний может получить за счет разработки более дифференцированных по уровню сложности заданий, как теоретического, так и практического плана. Очевидно, что это возможно только при высоком уровне учебно-методической работы преподавательского коллектива.

В условиях рыночных отношений итоговый рейтинг студента-выпускника может быть критерием для заказчиков при подборе кадров и заключении трудовых отношений.

# МЕТОДИЧНІ ОСНОВИ НАВЧАННЯ ІНФОРМАЦІЙНОГО ПОШУКУ ТА ПРОБЛЕМА ІНДИВІДУАЛІЗАЦІЇ ЗНАНЬ СТУДЕНТІВ

С.І. Романенко

м. Біла Церква, Білоцерківський державний аграрний університет

Отримання Україною статусу незалежної держави обумовило потребу докорінних соціально – економічних змін. Серед них важливим є створення засад для формування в Україні інформаційного суспільства. Такі зміни потребують нових підходів у підготовці фахівців, що здобувають освіту у вищих навчальних закладах.

Вступ суспільства в ХХІ ст., вік інформаційних технологій, вимагає нового осмислення ступеню готовності членів суспільства до ефективного формування і засвоєння інформаційних ресурсів. Вищі заклади освіти повинні забезпечувати своїм випускникам провідні позиції в інтелектуально-культурному авангарді суспільства. У вищих навчальних закладах закладаються глибокі та вузькоспеціальні знання, які допомагають стати людині спеціалістом в тій чи іншій галузі.

Успіхи в науково-технічному розвитку людства, постійне вдосконалення інформаційного забезпечення всіх сфер суспільного виробництва вносять в систему освіти значні позитивні зміни і разом з тим породжують закономірні проблеми. Нові інформаційні технології навчання нині посідають належне місце в роботі навчальних закладів усіх рівнів, необхідність їх використання у гармонійному поєднанні з традиційними навчальними методами усвідомлена переважною більшістю як теоретиків, так і практиків освітянської сфери. “Дедалі більша кількість молодих людей стали пов’язувати перспективи отримання роботи, й хорошої роботи зокрема, з високим рівнем освіти і кваліфікації за сучасною професією” [2]. В ХХІ ст. конкурентноздатність підприємців та фізичних осіб буде визначатись обсягом та якістю інформаційних ресурсів, якими вони володіють. Чим повніша інформація, яка потрібна людині і якою вона може оволодіти, і чим своєчасніше вона цю інформацію отримає, тим у вигіднішо-

му становищі в порівнянні зі своїми колегами по роботі чи навчанні вона опиниться. Член інформаційного суспільства повинен мати можливість користування засобами інформаційного обслуговування, знати про існування загальнодоступних джерел інформації і вміти ними користуватись.

“Вважається, що найважливішим наслідком впровадження комп’ютерів є сучасна миттєва можливість спілкування людей через комп’ютерні мережі, що перетворює весь світ у єдине інформаційне середовище. Сьогодні у світі налічується близько 250 мільйонів користувачів так званої “повної” глобальної мережі Інтернет (з них усього лише близько 3 тис. в Україні)” [4]. Вивчення Інтернет-технологій дозволить повніше розкрити науковий та теоретичний зміст курсу інформатики, а також ефектніше використати широкий аспект зв’язків предмету основ інформатики та комп’ютерної техніки з іншими навчальними дисциплінами. “Застосування комп’ютерних мереж скорочує час навчання майже втричі, а рівень запам’ятовування зростає на 35-40%” [4].

Ми погоджуємось із думкою Т. Ростовцевої про те, що пошук в новому інформаційному середовищі можна розбити на 3 етапи:

*Етап 1. Формулювання і уточнення інформаційного запиту.*

Крок 1. Формулювання запиту на звичайній мові. Мета цього кроку – формування уявлень про шукану інформацію. Для того щоб отримати це уявлення, користувачеві необхідно якомога точніше подати її на звичайній мові.

Крок 2. Визначення типу інформаційного запиту (межі інформаційної потреби чітко визначені; межі визначені нечітко; межі можуть змінюватися з часом).

Крок 3. Визначення мети пошуку інформації (дозволяє виявити ступінь повноти потрібної інформації: попереднє знайомство з проблемою чи ґрунтовне її вивчення ).

Крок 4. Визначення необхідних обмежень пошуку. Основними традиційними обмеженнями є: обмеження за видами видань, мовами публікацій, за географічними та хронологічними межами пошуку. Тому необхідно чітко з’ясувати інформація з яких джерел цікавить користувача (статті, монографії, матеріали конференцій), на якій мові, які географічні межі пошуку, які

хронологічні межі.

Крок 5. Кінцеве формулювання інформаційного запиту (для кінцевого формулювання інформаційного запиту необхідно провести всебічний лексичний аналіз інформації, яка підлягає пошуку. Необхідно сформулювати максимально широкий набір ключових слів у вигляді окремих термінів, словосполучень, професійної лексики, кліше, за необхідності на декількох мовах).

*Етап 2. Планування пошукової процедури.*

Крок 6. Вибір та оцінка якості пошукових систем (під час вибору пошукових систем необхідно звертати увагу на такі параметри: об'єм індексу пошукової системи, контроль якості інформації, що міститься в ній, можливість мовного запиту).

Крок 7. Розробка загальної програми пошуку. Складовими процесу розробки загальної програми пошуку є визначення об'єктів, методів і можливих напрямків пошуку на основі виявлених обмежень пошуку (тематичних, хронологічних, мовних, географічних), а також ступеня повноти пошуку.

*Етап 3. Реалізація пошуку.*

Крок 8. Відбір сукупності інформаційних ресурсів з теми пошуку (на цьому етапі необхідно чітко визначити, що відомо про відшукувану інформацію і що невідомо, але необхідно з'ясувати). Даний крок дозволяє створити персональну інформаційну систему, спрямовану на вирішення конкретного завдання пошуку.

Крок 9. Проведення пошуку в інформаційних ресурсах (при проведенні пошуку документів необхідно мати можливість як звужити, так і розширити запит, використовуючи дві базові структури запиту: галузевий і предметний).

Крок 10. Оцінка повноти інформації. Виявивши частину документів шляхом проведення пошуку в первинному колі інформаційних ресурсів, доречно здійснити перегляд та оцінку повноти отриманої інформації. Далеко не завжди користувачеві необхідна вся інформація із заявленої теми. Іноді йому достатньо лише декілька авторитетних джерел. На цьому етапі формується список документів і здійснюється перевірка на дублетність. Однак необхідно здійснити додатковий пошук інформаційних ресурсів, коли повнота отриманої інформації є недостатньою.

Крок 11. Розширення кола інформаційних ресурсів (необхід-



но провести пошук у зарубіжних мегапошукових системах з метою виявлення баз даних, присвячених конкретно тематиці пошуку, провести пошук за гіпертекстовим посиланням, рухаючись вже від знайдених документів) [5].

На правильно сформульоване питання в Інтернеті практично завжди можна отримати відповідь. Але при цьому не слід забувати, що задасте питання не людині, а машині, яка аналізує інформацію в залежності від закладених в неї алгоритмів. Програми, які обробляють запит, – це інформаційно-пошукові системи [3]. В україномовному Інтернеті прикладами словникових ПС (інформаційно-пошукових систем) є сервери: [www.ukrop.com](http://www.ukrop.com), [www.uaportal.com](http://www.uaportal.com), [www.meta-ukraine.com](http://www.meta-ukraine.com), в російськомовному – [yandex.ru](http://yandex.ru), [www.aport.ru](http://www.aport.ru), [www.rambler.ru](http://www.rambler.ru).

Для оптимальної і швидкої роботи з пошуковими системами такого типу існують певні правила написання запитів. Незалежно від того, яка форма слова стоїть в запиті, пошук містить усі словоформи за правилами української чи російської мови. Якщо в запиті слово набране з великої літери, то всі ПС будуть враховувати регістр, в якому набрані літери. Якщо запит являє собою одне або декілька слів, то по ньому будуть знайдені документи, в яких зустрічаються всі слова запиту. Зірочка в кінці слова символізує довільну кількість будь-яких літер. Лапки вказують пошуковій системі, що дані в запиті слова є ключовою фразою, яку потрібно шукати в тексті повністю.

Для ефективності інформаційного пошуку слід використовувати логічні оператори: И – синоніми AND, & ,+ (швидкий пошук), ИЛИ – синоніми: OR, |, “кома” (дозволяють відшукувати документи, які містять хоча б одне із вказаних слів), НЕ – синоніми : NOT, “мінус”, ~ обмежують пошук документами, що не містять слово, вказане після даного оператора [1, 3].

Ми зупинились на загальних підходах до навчання студентів інформаційного пошуку. Ними можуть скористатись усі, хто прагне оптимізувати процес пошуку необхідної інформації в Інтернеті. Однак студент вищого навчального закладу з позицій сучасної парадигми освіти в Україні розглядається насамперед як особистість, як неповторна індивідуальність, яку необхідно поглиблювати. Реалізації цього завдання сприяє індивідуалізація навчання студентів. Під індивідуалізацією навчання сьогодні ро-

зуміють вивчення його змісту, вибір методів і форм відповідно до природних та особистісних якостей, досвіду й досягнень студентів. Засадовим положенням індивідуалізації є врахування індивідуальних особливостей майбутніх фахівців. У процесі навчання інформаційного пошуку викладач має можливість використовувати індивідуальні завдання, що відповідають інтересам студентів, тісно пов'язані з тими видами діяльності, до яких вони мають здібності (тобто здатні їх успішно виконувати). Наприклад, студентам пропонується знайти інформацію про творчу діяльність улюбленого співака, художника, композитора; про досягнення відомих спортсменів, культурних, політичних діячів; інформацію про необхідне літературне джерело та бібліотеку, де його можна відшукати; відвідати віртуальний музей. Така організація занять сприяє формуванню в студентів таких якостей, як самостійність, ініціативність, творчість, захопленість, дослідницький стиль діяльності, культура пошуку і праці.

Методичні рекомендації щодо навчання студентів інформаційного пошуку дозволять значно розширити можливості студентів в освоєнні комплексу спеціальних та професійних дисциплін, ефективно використовувати сучасні засоби комп'ютерної техніки на лабораторних і практичних заняттях, у процесі підготовки курсових та дипломних робіт.

### Література

1. Бочкин А.И. Методика обучения информационному поиску // Информатика и образование. – 2000. – № 3. – С. 7–10.
2. Грішнова О. Розвиток вищої освіти в Україні: тенденції, проблеми та шляхи їх вирішення // Вища школа. – 2001. – № 2–3. – С. 22–33.
3. Исаева О.В. Поиск информации в Интернете // Информатика и образование. – 2000. – №6 – С. 76–88.
4. Петренко А. Віртуальні університети і лабораторії (Інформаційні технології в освіті) // Вища освіта України. – 2002. – №1. – С. 21–25.
5. Ростовцева Т. Организация и использование информационных ресурсов // Информационные ресурсы России. – 2001. – №8. – С. 11–15.

## МЕТОД ОЦЕНКИ ЭФФЕКТИВНОСТИ ТЕСТОВ ДИАГНОСТИРОВАНИЯ МИКРОСХЕМ ПАМЯТИ

В.Г. Рябцев

г. Черкассы, Черкасский государственный технологический университет  
cheti@cheti.cherkassy.ua

Многие фирмы-изготовители микросхем памяти выполняют их тестовое диагностирование без научного обоснования эффективности применяемых тестов. Свойства применяемых тестов изучены недостаточно, из-за значительного объема памяти ЭВМ, необходимой для хранения программ имитационных моделей, и большой продолжительности процесса моделирования.

Впервые модель оперативного запоминающего устройства предложена Хейсом С.П. (Hayes S.P.) [1] и представлена в виде цифрового автомата, содержащего множество запоминающих ячеек  $C_i \in C$ , которые подвергаются воздействию операторов:

$$O_i = \{W_i, V_i, R_i\},$$

где  $W_i$  – оператор записи единицы в ячейку  $C_i$ ;  $V_i$  – оператор записи 0 в ячейку  $C_i$ ;  $R_i$  – оператор считывания состояния ячейки  $C_i$ . Обычно емкость памяти в такой модели ограничивают двумя-тремя ячейками, что существенно сокращает продолжительность исследований.

Количество операций, при помощи которых можно изменить состояния запоминающих ячеек, определяется по формуле:  $O_{sm} = n * 2^n$ , где  $n$  – емкость микросхемы памяти. Для трех запоминающих ячеек число возможных операций, изменяющих состояния ячеек памяти равно 24.

Для оценки диагностических свойств тестов предлагается модифицировать модель памяти Хейса и получать модели, которые описывают функционирование микросхем, содержащих неисправности. Рассмотрим константные неисправности микросхемы памяти – константный ноль и константную единицу запоминающих ячеек. Данные неисправности можно представить в виде графа состояний, в котором одна или несколько ячеек не изменяют свои состояния.

На рис. 1 представлена модель запоминающего устройства,

содержащего неисправность константа ноль ячейки, код адреса которой равен нулю. Такая ячейка не может переключиться в единичное состояние, поэтому недопустимые состояния показаны на рис. 1 пунктиром.

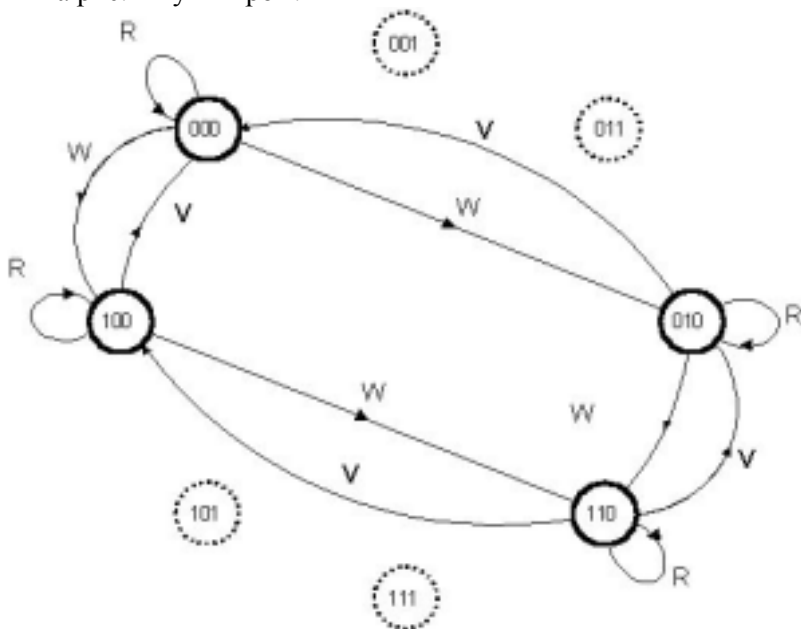


Рис. 1. Модель запоминающего устройства с неисправностью константа ноль

Данную диаграмму состояний микросхемы можно преобразовать в матрицы соединений, элементы которых содержат операции, необходимые для перехода микросхемы из одного состояния в другое [2].

Элемент матрицы на пересечении  $i$ -той строки и  $j$ -того столбца содержат значения операций при наличии переходов из состояния  $S_i$  в состояние  $S_j$ . Если возможны несколько переходов из  $S_i$  в  $S_j$ , то  $M_{i,j}$  содержит дизъюнкцию выполняемых операций. Если переход не выполняется, то такой элемент матрицы будет пустым.

Матрица соединений диаграммы состояний  $M_{k1}$  данной микросхемы при наличии неисправности константа единица в нуле-

вой ячейке имеет вид:

	$R(A)$		$W(A_1)$		$W(A_2)$		
$M_{k1} =$	$V(A_1)$		$R(A)$				$W(A_2)$
	$V(A_2)$				$R(A)$		$W(A_1)$
			$V(A_2)$		$V(A_1)$		$R(A)$

Для оценки возможности выявления неисправностей при помощи теста марш рассмотрим состояния ячеек памяти, которые возникают при выполнении данного теста [3, 4]. При выполнении данного теста выполняется  $O_{st} = 4n$  операций обращения к микросхеме памяти, которые обеспечивают получение состояний ячеек, приведенных на рис. 2.

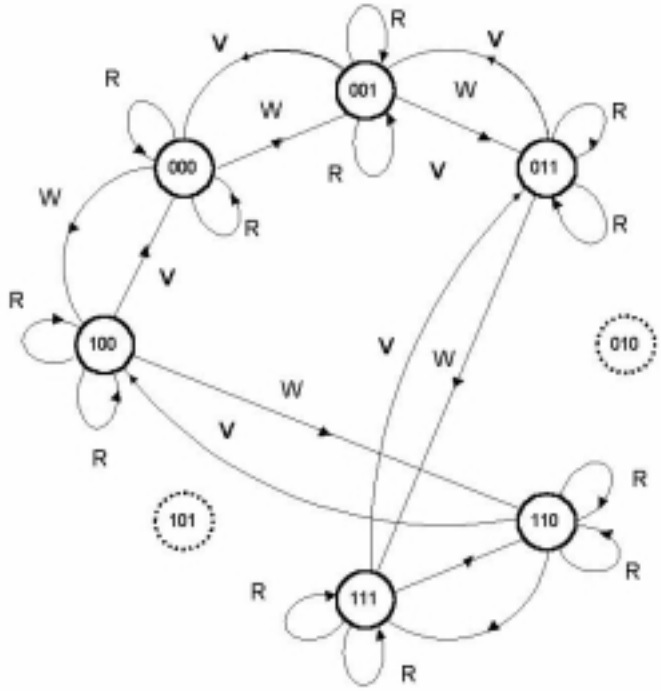


Рис. 2. Последовательность смены состояний трех ячеек памяти при выполнении теста марш

Матрица соединений диаграммы состояний  $M_m$  автомата, реализующего тест марш, приведена ниже.

$$M_m =$$

$R(A)$	$W(A_0)$			$W(A_2)$			
$V(A_0)$	$R(A)$		$W(A_1)$				
	$V(A_1)$		$R(A)$				$W(A_2)$
$V(A_2)$				$R(A)$		$W(A_1)$	
				$V(A_1)$		$R(A)$	$W(A_0)$
			$V(A_2)$			$V(A_0)$	$R(A)$

Всего в микросхеме памяти может быть  $S_m = 2^n$  возможных состояний, однако при выполнении теста марш не все они формируются.

Для оценки возможности обнаружения неисправности константа единица тестом марш определим разности между элементами, имеющими одинаковые индексы, матрицы состояний теста марш  $M_m$  и матрицы соединений микросхемы, имеющей неисправность:

$$(m_{ij})^{no} = (m_{ij})^{tm} \cdot (m_{ij})^{k1}.$$

Операцию определения разности выполняем также как для элементов множеств, поэтому во вновь образованной матрице отсутствуют отрицательные элементы. Матрица невыполнимых операций для неисправности константа единица имеет вид:

$$M_{no} =$$

$R(A)$	$W(A_0)$			$W(A_2)$			
$V(A_0)$							
$V(A_2)$				$R(A)$		$W(A_1)$	
				$V(A_1)$		$R(A)$	$W(A_0)$
						$V(A_0)$	

При выполнении теста марш для трех ячеек памяти обеспечивается проверка возможности формирования следующих состояний 000, 001, 011, 111, 110 и 100. В модели с неисправностью константа единица нулевой ячейки отсутствуют состояния 000, 110 и 100, а в модели с неисправностью константа ноль отсутствуют состояния 001, 011, 111, которые формируются тес-

том марш, следовательно, константные неисправности запоминающих ячеек тестом марш выявляются.

Однако при диагностике перемежающихся отказов диагностические свойства данного теста снижаются, что не обеспечивает эффективную проверку наличия данных видов отказов в микросхемах памяти большой емкости. Для выполнения эффективных процедур диагностирования микросхем памяти рекомендуется применять другие тесты, например, приведенные в [3, 4].

### **Литература**

1. Hayes S.P. Detection of pattern-sensitive fault in random-access memories // IEEE Trans. Comp. – 1975, № 2. – P. 150–157.
2. Мелихов А.Н. Ориентированные графы и конечные автоматы. – М.: Наука, 1971. – 416 с.
3. Мельников А.В., Рябцев В.Г. Контроль модулей памяти компьютеров. – К.: “Корнійчук”, 2001. – 172 с.
4. 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.

# ПСИХОЛОГО-ПЕДАГОГІЧНІ АСПЕКТИ ВЗАЄМОЗВ'ЯЗКУ ПІЗНАВАЛЬНОЇ ДІЯЛЬНОСТІ ТА ВИКОРИСТАННЯ КОМП'ЮТЕРНОЇ ТЕХНІКИ ПРИ ВИВЧЕННІ ДИСЦИПЛІН ІНФОРМАЦІЙНОГО ЦИКЛУ

Л.О. Савчук

м. Хмельницький, Хмельницький інститут бізнесу

Успішне вирішення багатоаспектних проблем комп'ютеризації в сфері освіти можливе лише при виконанні ряду психолого-педагогічних вимог та умов, які передбачають ефективність навчально-виховної, управлінської й науково-дослідницької педагогічної діяльності. Актуальність психолого-педагогічної проблематики обумовлена насамперед тим, що вона охоплює практично всі напрямки використання комп'ютера. Зокрема, до зазначених проблем відносяться такі:

- різнорівневі умови розвитку студентів до початку навчання у вузі;
- подолання психологічного бар'єру, який виникає у потенційних користувачів по відношенню до інформаційних технологій і пов'язаною з цим потребою оволодівати теоретичним матеріалом, який необхідний для роботи на цій техніці;
- стимулювання психологічних механізмів організації уваги, тренування пам'яті, розвиток мислення при використанні комп'ютерної техніки;
- підвищення рівня інформаційної насиченості заняття, як засобу активізації пізнавальної діяльності.

Психолого-педагогічне обґрунтування, безумовно, необхідне і при вирішенні питань, пов'язаних з використанням інформаційних технологій в навчальному процесі, що передбачає органічну триєдність мети: навчання, виховання й розвиток студентів.

Та найбільш актуальною, на сьогоднішній день, є проблема психолого-педагогічного забезпечення використання комп'ютерної техніки в якості ефективного засобу навчання. Тут виникає безліч питань, так чи інакше пов'язаних з розробкою цілісної психолого-педагогічної концепції навчання інформаційним технологіям. Мається на увазі весь комплекс питань,



пов'язаних з вирішенням таких проблем, як формування мотивації та пізнавального інтересу в навчанні; встановлення раціонального, педагогічно виправданого діалогового спілкування студентів з комп'ютером на всіх етапах ознайомлення і засвоєння відповідної навчальної інформації; поєднання індивідуальних, групових і колективних форм комп'ютерного навчання; активізація навчально-пізнавальної діяльності студентів, розвиток їх самостійності; організація оперативного контролю і самоконтролю результатів навчально-пізнавальної і творчої діяльності; виявлення найбільш ефективних шляхів формування творчих здібностей студентів; організація продуктивної взаємодії викладача і студентів, а також студентів один з одним в умовах комп'ютерного навчання.

Потрібно зазначити, що більшість з вказаних вище проблем ще не досліджені повністю. З'являються публікації в яких визначаються раціональні шляхи використання комп'ютерної техніки в навчальному процесі з певних психолого-педагогічних позицій [1; 2]. Обґрунтуванню психолого-педагогічних проблем комп'ютеризації присвячені також праці багатьох зарубіжних вчених. Американські вчені Л. Резнік, Л. Клопфер, розкриваючи сутність пізнавальних цілей, вказують на необхідність створення нового ядра теорії навчання, заснованого на конструктивістських, саморегулюючих положеннях про природу вчення. Серед основних елементів подібного навчання називаються: способи презентації і визначення послідовності наданої інформації; організація зворотного зв'язку; мотивація; зв'язок лабораторних занять з іншими формами навчання; оцінка діяльності [3, с. 4].

Одним з головних засобів вирішення зазначених психолого-педагогічних проблем щодо ефективного використання комп'ютера в навчальному процесі є залучення студентів до активної пізнавальної діяльності. Існують різні підходи до визначення даного поняття. М. Хант пізнавальну діяльність розглядає через афективну сферу [4]. А Н.Ф.Тализіна, наприклад, досліджує пізнавальну діяльність з точки зору оволодіння знаннями, вміннями і навичками [5]. І дійсно, про активну пізнавальну діяльність студента можна говорити лише тоді, коли він самостійно може вирішувати навчально-пізнавальні завдання, шукати нові шляхи їх розв'язку проявляючи ініціативу, творчість. Та при

цьому зазначимо, що структура діяльності людини істотно змінюється у зв'язку з використанням сучасних інформаційних технологій. Комп'ютер є потужним інструментом, який розвиває інтелектуальну і мотиваційну сферу особистості людини, вносячи свої корективи. Тому використання комп'ютера у процесі навчання відіграє важливу роль в становленні особистості студентів за умови педагогічного осмислення впровадження комп'ютера у навчальний процес.

Психолого-педагогічний аналіз основних етапів повного циклу комп'ютерного навчання дозволяє виокремити пізнавальну діяльність, як основний елемент ефективності учіння при викладанні дисциплін інформаційного циклу (рис. 1).



Рис. 1. Психолого-педагогічний аналіз взаємозв'язку пізнавальної діяльності та процесу учіння при вивченні дисциплін інформаційного циклу

Психологічний бар'єр, який виникає в потенційних користувачів, знімається на етапі формулювання цілей і мотивації навчальної діяльності. Мета обирається відповідно до можливостей та інтересів студента. Важливим є те, що один з найбільш ефек-

тивних прийомів стимулювання розумової діяльності студентів, формування мотивів навчання пов'язаний зі створенням проблемних ситуацій. При цьому повинні бути необхідні умови для того, щоб сформувати у студента такі прийоми розумової і практичної діяльності, які в найбільшій мірі відповідають змісту і характеру навчальних завдань, забезпечуючи одночасно можливість активізації навчально-пізнавальної діяльності. Це спонукає студента до самостійної діяльності, творчого пошуку рішень того чи іншого навчального завдання, засвоєння програмного матеріалу та здобуття необхідних вмінь і навичок, їх практичного застосування і корекції для постановки більш віддаленої мети. Подача нового матеріалу опирається не лише на пам'ять студента, а й на включення і розвиток його мислення. Закріплення матеріалу базується не лише на його відтворенні, але і на творчому переосмисленні. Основна частина нових знань повинна засвоюватися не в готовому вигляді, а в процесі самостійного творчого пошуку за рахунок комп'ютерного моделювання різних ситуацій, використання різноманітних ділових, ролевих і сюжетних ігор.

Стимуляція викладених психічних процесів позитивно впливає на вирішення зазначених питань і призводить до підвищення інтелектуального рівня студентів та активізації їх пізнавальної діяльності.

#### Література

1. Winne Ph. H. Steps toward promoting cognitive achievements // Making our schools more effective: Proceedings of Three State Conferences. – USA, 1989. – P. 166-187. – ED2499588
2. Слєпкань З.І. Наукові засади педагогічного процесу у вищій школі. Конспект лекцій. – К: НПУ ім. М.П. Драгоманова, 1999. – 150 с.
3. Resnik L.B., Klopfer L.E. Toward the thinking curriculum: Current cognitive research. – Alexandria, VA, 1989. – p. 1-18.
4. Hant W.C. Teaching and learning in the affective domain. A review of the literature – Olimpia: Information Analysis (070), 1987, – 100 p. – ED288871.
5. Галызина Н.Ф., Гальперин П.Я. Формирование познавательной деятельности младших школьников. – М.: Просвещение, 1988. – 175 с.

## ОРГАНІЗАЦІЯ ПРОЦЕДУР ДОСТУПУ ДО ІНФОРМАЦІЙНИХ РЕСУРСІВ ТРІАС

Н.І. Соколянська

м. Київ, Інститут проблем реєстрації інформації НАН України

У відповідності з сучасним підходом, виділяють наступні аспекти інформаційної взаємодії у територіально-розподіленій інформаційно-аналітичній системі (ТІАС): доступність інформації (можливість за прийнятний час отримати необхідну інформаційну послугу); цілісність інформації (актуальність і непротиворічність інформації, її захищеність від руйнування і несанкціонованої зміни); конфіденційність інформації (захист від несанкціонованого ознайомлення).

Задачу забезпечення доступу до інформації необхідно вирішувати для сучасних конфігурацій, побудованих в технології клієнт/сервер. Це означає, що захисту потребує весь ланцюжок від користувачів (можливо, віддалених) до серверів.

Відповідно до підходу клієнт/сервер, доступ до ресурсів в інформаційній системі можна представити у вигляді наступної моделі інформаційної взаємодії (див. рис. 1).

Основною задачею контролю і керування доступом в інформаційних системах є встановлення множини операцій, дозволених для користувачів. Існує два основних механізми керування доступом – дискреційний (довільний) і мандатний (нормативний).

Доступність системи в загальному випадку досягається за рахунок застосування трьох груп мір, направлених на її підвищення:

1. Безвідмовності (під цим розуміється мінімізація імовірності виникнення будь-якої відмови);
2. Живучості (під цим розуміється збереження доступу до системи незважаючи на відмову будь-яких компонентів);
3. Обслуговуємості (під цим розуміється мінімізація часу непрацездатності компонентів, що відмовили, а також мінімізація негативного впливу ремонтних робіт на ефективність інформаційних сервісів).

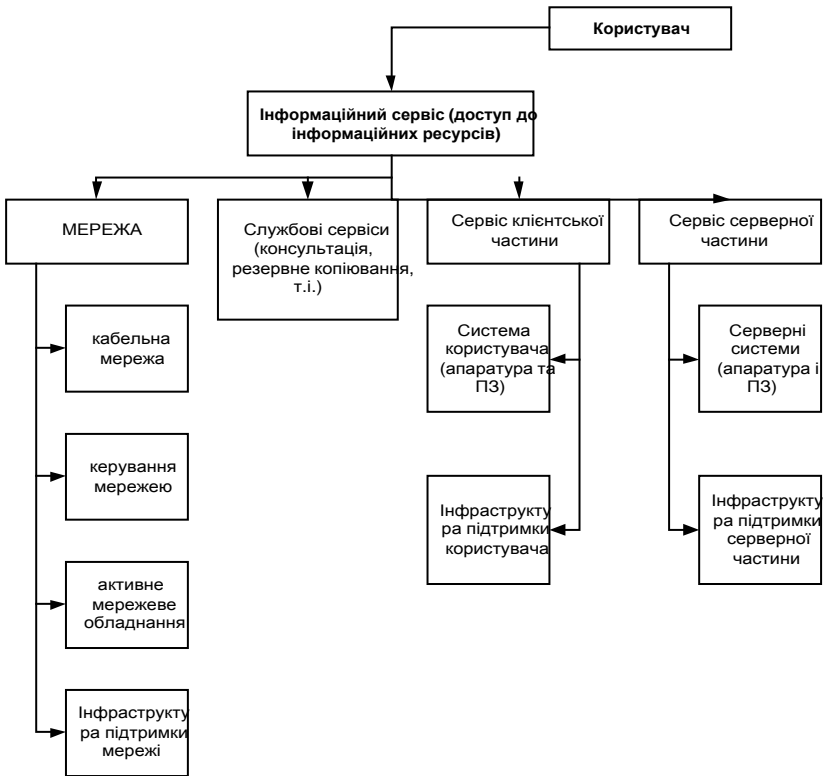


Рис. 1. Модель сервісів доступу до ресурсів ТРІАС

Очевидно, щоб система не містила поодиноких точок відмови, тобто залишалася живучою при виникненні будь-якої загрози, нейтралізацію відмов потрібно виконувати всередині системи, невидимим для користувачів чином, за рахунок розміщення достатньої кількості надлишкових ресурсів.

### Міри забезпечення доступу до ресурсів

Формування режиму інформаційної безпеки взагалі і забезпечення високої доступності зокрема – проблема комплексна. Заходи щодо її вирішення, які можуть бути вжиті в рамках ТРІАС, можна поділити на три рівні:

- адміністративний (дії загального характеру, що вживаються керівництвом організації, такі як вироблення політики

безпеки стратегії доступу);

- процедурний (конкретні міри, які орієнтовані на людей);
- програмно-технічний (конкретні програмно-технічні засоби).

### **Механізми контролю за здійсненням доступу**

Задача засобів контролю за здійсненням доступу – забезпечити виконання правил безпеки. Традиційно контроль доступу здійснюється на двох ступенях – апаратному (кільця захисту та ізоляція адресних просторів) і загальносистемному (контроль доступу до об'єктів файлової системи на рівні системних викликів ядра). Переваги технологій, застосованих при побудові мікроядерних ОС, дозволяють здійснювати багатоступінчастий контроль за здійсненням доступу на декількох рівнях взаємодії. Для мікроядерних ОС в ієрархії засобів контролю доступом можна виділити чотири рівня: апаратний контроль, контроль на рівні мікроядра, загальносистемний і прикладний.

### **Основні положення щодо доступу до інформаційних ресурсів**

Основна задача політики стратегії доступу – сформувані сукупність норм і правил, що регламентують порядок опрацювання інформації в системі, із метою запобігання несанкціонованому доступу до інформації, що веде до її розголошення або модифікації. Керування доступом в інформаційних системах базується на традиційній концепції контролю доступу активних компонентів системи (суб'єктів) до пасивних (об'єктів). Основним механізмом контролю доступу є монітор взаємодій, в обхід якого здійснити доступ неможливо.

При розробці політики стратегії доступу до інформаційних ресурсів рекомендується керуватися наступними стратегічними принципами:

- апробованість усіх процесів і складових інформаційної системи;
- уніфікація процесів і складових;
- керованість процесів, контроль стану частин;
- автоматизація процесів;
- модульність архітектури;
- орієнтація на простоту рішень.

Політика стратегії доступу повинна передбачати визначення

загального порядку робіт, виділення відповідальних за розробку документів, які конкретизують програму, та за періодичний перегляд і уточнення документів, розроблених раніше. Головне, що повинна забезпечити подібна програма, – це повнота і систематичність при проведенні робіт. Сфера чинності програми повинна розповсюджуватися на всі інформаційні ланцюжки – від інфраструктури до користувачів. Кожний процес, кожний складник ТРІАС повинні мати відповідального на адміністративному рівні, що володіє достатніми повноваженнями для виділення відповідних ресурсів і для впливу на виконавців. В цьому зв'язку представляється доцільним передбачити укладання карти ТРІАС (в друкованому і комп'ютерному представленнях), в якій фігурували б усі об'єкти ТРІАС, їхній стан, зв'язки між ними, процеси, що асоціюються з об'єктами і зв'язками. За допомогою подібної карти зручно формулювати міри, що намічаються, контролювати їхнє виконання, аналізувати стан ТРІАС. Необхідно розробити регламент, що гарантує підтримання карти в актуальному стані. В програмі повинні знайти відбиток всі три основні напрямки забезпечення доступу до інформації: безвідмовність, живучість, обслуговуємість.

### **Висновки**

Щоб міри по забезпеченню доступу до інформаційних ресурсів були ефективними, вони повинні охоплювати всі рівні - адміністративний, процедурний, програмно-технічний; впродовж усього життєвого циклу інформаційної системи, всі її компоненти – інфраструктуру, апаратуру, програми, дані, персонал, користувачів. Необхідно піклуватися про всі аспекти доступу – безвідмовність, живучість, обслуговуємість. Ці загальні положення при практичному використанні повинні бути доповнені розробкою конкретних адміністративних, процедурних і програмно-технічних мір.

## ОРГАНИЗАЦИОННЫЕ И МЕТОДОЛОГИЧЕСКИЕ ПРОБЛЕМЫ ПРЕПОДАВАНИЯ ДИСЦИПЛИНЫ «ИНФОРМАТИКА И КОМПЬЮТЕРНАЯ ТЕХНИКА» В ВЫСШЕЙ ШКОЛЕ

В.А. Стороженко, А.Ю. Вакула  
г. Одесса, Одесский государственный экономический университет  
Alice\_v@ukr.net, victory@te.net.ua

Коренное отличие информатики от других дисциплин, изучаемых в высшей школе, состоит в том, что её предмет изучения меняется ускоренными темпами. Один раз в полтора года в среднем удваиваются основные технические параметры аппаратных средств, один раз в два-три года меняются поколения программного обеспечения. Таким образом, кардинальным отличием информатики от других дисциплин является тот факт, что её предметная область изменяется чрезвычайно динамично.

Все, кто причастен к преподаванию информатики в высшей школе, хорошо знают, как часто приходится менять содержимое учебных планов, рабочих программ, учебно-методической документации. Кроме того, возникают проблемы соответствия материально-технической базы учебного процесса текущему состоянию предметной области.

Поэтому для преподавания курса «Информатика и компьютерная техника» необходимо расширенное взаимодействие между учебными программами по информатике и другими фундаментальными дисциплинами. Основные принципы, вытекающие из такого подхода, включают *непрерывность* и *системность* образования, а также – *раннюю профессиональную ориентацию*.

Непрерывность образования обеспечивается тем, что практические приемы работы со средствами вычислительной техники закрепляются не только в рамках дисциплины «Информатика и компьютерная техника», но и в течение всего периода обучения при проведении занятий по самым разным дисциплинам.

Системность образования предполагает перекрестное взаимодействие изучаемых дисциплин. Конкретная дисциплина представляет комплекс *задача – методы*, а информатика обеспечива-



ет комплекс *средства – приемы*. Таким образом, обеспечивается единый методологический подход, основанный на системе *задача – средство – методы – приемы*.

Ранняя профессиональная ориентация обеспечивается тем, что информатика является одной из немногих общетехнических дисциплин, развивающая такие практические навыки, которые могут быть востребованы сразу же после включения молодого специалиста в профессиональную деятельность.

Из нашего опыта преподавания курса «Информатика и компьютерная техника» в ОГЭУ хотелось бы отметить ряд особенностей, которые являются предпосылками создания частной методики преподавания этой дисциплины:

1) не все студенты имеют одинаковую предварительную подготовку по данному предмету;

2) многие студенты могут «работать» на ПК, но, в основном, их знания и навыки сводятся к умению использовать лишь некоторые возможности ПК на уровне самых простых операций и поиском информации в Internet;

3) для улучшения качества усвоения учебного материала необходимо больше времени выделять практической и самостоятельной работе. При этом очень важно правильно определить, использовать и сочетать все виды практической работы студента: практика, лабораторная работа, индивидуальные задания, самостоятельная работа. Например, каждый из перечисленных видов должен выполняться студентом самостоятельно и индивидуально, каждый вид работы должен оцениваться по отдельным критериям;

4) для обеспечения самостоятельной работы должен быть создан банк заданий, представляющих интерес для студентов и имеющих практическое применение в их специальности;

5) необходимо учитывать различные виды контроля знаний в соответствии с используемой в вузе рейтинговой системой оценки знаний;

6) для удобства работы с теоретической частью курса, необходимо оформлять этот материал в специальном виде удобном для использования;

7) довольно сложно принимать у студентов отчеты по выполненной работе только на основании созданных электронных

документов, что вызывает необходимость создания специальных форм отчетности (рабочих тетрадей, дневников), которые должны содержать задания и отчеты по их выполнению.

Опыт преподавания дисциплины и анализ качества обучения по различным моделям показал необходимость разработки методики, охватывающей все виды учебной работы и предлагающей возможности наиболее объективной оценки знаний каждого студента.

Современный уровень технической оснащенности в нашем ВУЗе позволяет при изложении теоретического материала пользоваться мультимедийным проектором. Однако, как показывает опыт, через 40-50 минут снижается концентрация внимания студентов, они не всегда фиксируют в конспекте все необходимое, отвлекаясь на внешние эффекты лекционного материала. Все это вызвало необходимость оформить теоретический материал курса в виде опорных конспектов лекций, в которые по ходу проведения лекции студенты могли бы вносить дополнения. А практическую работу по каждой теме курса предлагается разделить по направлениям:

- *практическое занятие* – проводится традиционно с обсуждением в аудитории основных вопросов (они приводятся в методических указаниях, ответы студент готовит в процессе подготовки к занятию). В методических указаниях приводятся также основные сведения по рассматриваемой теме и задания для выполнения.
- *лабораторная работа* – включает постановку задания с указанием общей справочной информации. Результатом работы студента является выполненное на компьютере задание и отчет, оформленный в соответствии с требованиями, приведенными в методических указаниях.
- *индивидуальная работа* – комплексное задание (отдельно для каждого студента), включающее задания, основывающиеся на знаниях и навыках, приобретенных студентами при выполнении предшествующих практических и лабораторных заданий. Работа выполняется во внеурочное время. На основании лабораторных работ и индивидуального задания определяется рейтинг студента по теме курса.
- *самостоятельная работа* – включает задания (необязатель-

ные) по каждой теме, предназначенные для работы студентов во внеурочное время для лучшего усвоения темы. Задания приводятся по разным степеням сложности. Возникающие проблемы можно решать с преподавателем во время, предназначенное для консультаций. Эти же задания могут быть использованы преподавателем при различных уровнях дополнительного контроля знаний в случае необходимости.

Рейтинговая система предполагает три уровня: первые два основываются на результатах выполнения практической работы, а третий – на проверке полученных теоретических знаний и выполняется в виде тестового контроля с использованием компьютера. Отдельные вопросы такого контроля приводятся в методических указаниях каждого занятия. Кроме того, после изучения каждой темы студенты готовят вопросы для тестового контроля, которые по решению преподавателя могут включаться в итоговые тесты.

Методические указания и задания по практической работе оформляются в виде дневника студента, в котором помимо заданий для выполнения приводятся требования по оформлению отчётности по каждому лабораторному и индивидуальному заданию, и выполняются в самом дневнике. Такой дневник студент может использовать в дальнейшем, как методическое руководство в своей практической деятельности.

## РОЗРОБКА СИСТЕМИ НАВЧАННЯ ТА КОНТРОЛЮ ЗНАНЬ З ДИСЦИПЛІНИ “ТЕОРІЯ ОПЕРАЦІЙНИХ СИСТЕМ”

А.М. Стрюк, О.В. Попова  
м. Кривий Ріг, Криворізький технічний університет  
[andrey\\_stryuk@mail.ru](mailto:andrey_stryuk@mail.ru), [oksanap@ingok.com.ua](mailto:oksanap@ingok.com.ua)

Теорія операційних систем – одна з найважливіших дисциплін, що викладається при підготовці студентів за спеціальністю “Програмне забезпечення автоматизованих систем”. В цьому курсі студенти знайомляться зі структурою операційних систем, функціонуванням різних модулів сучасних ОС, виконують лабораторні роботи, вивчаючи можливості сучасних операційних систем. Знання, що здобуваються студентами під час занять, будуть використовуватися без винятку на кожному кроці їх професійної діяльності, бо чи можна уявити собі спеціаліста в галузі програмування, який не знає досконально особливостей найважливішої комп’ютерної програми – операційної системи – і не вміє її використовувати.

Але багатство теоретичних методів, розроблених на сьогоднішній день, велика кількість створених на сьогодні операційних систем, кожна з яких має свої індивідуальні властивості і особливості, складає собою грандіозний матеріал, який викладач, нажаль, ніяк не в змозі розкрити в повному обсязі за односеместровий курс викладання цієї дисципліни. Виправити ситуацію може спонукання студентів до самостійної роботи, організувати яку в сучасних умовах студенту досить складно: бібліотечні фонди застарілі і не забезпечують студентів необхідною інформацією, книжковий ринок, хоча і різноманітний, але не всім по кишені.

Тимчасовий вихід із такої ситуації ми вбачали у наданні необхідних матеріалів в електронному вигляді. Всі матеріали, що були оцифровані – підручники, довідники, методичні посібники, – були надані в розпорядження студентів. Для більшої зручності і задля того, щоб матеріалами могли скористатися не тільки наші студенти, силами викладачів кафедри моделювання та програмного забезпечення був створений інтернет-сайт

www.asterra.by.ru, на якому були розташовані матеріали для самостійної роботи з дисциплін “Системне програмування” та “Теорія операційних систем”. Вже через кілька місяців існування сайту стало ясно, що робота виконана не даремно – надходили відгуки з різних куточків України, з Росії, з Казахстану. Найбільше порадувало те, що значно зросла зацікавленість студентів і матеріали широко використовувалися ними в самостійній роботі.

Але і такий метод показав деякі свої недоліки. По-перше, матеріали в електронному вигляді важко сприймалися студентами. Причини тому виключно психологічні – традиційні книги більш звичні для сприйняття. Поліпшити ситуацію можна, якщо зробити книгу інтерактивною, забезпечити можливість робити закладки, поліпшити навігацію по книзі. По-друге, ні паперова, ні електронна книга не в змозі проконтролювати те, як студент засвоїв викладенні в ній знання. Більшість підручників мають наприкінці розділів питання для самоконтролю, самостійні завдання тощо, але не можуть проконтролювати їх виконання, підказати студенту шляхи удосконалення своїх знань.

Якщо ми хочемо позбутися цих недоліків, ми повинні розробити інтерактивну комп’ютерну систему, яка надавала б студентам необхідний матеріал в зручній формі і контролювала його засвоєння. Розробку саме такої системи поставили на меті викладачі кафедри моделювання та програмного забезпечення Криворізького технічного університету. Програма, що отримала назву “Система навчання та контролю знань з дисципліни “Теорія операційних систем”, розробляється в рамках дипломного проекту студенткою спеціальності програмне забезпечення автоматизованих систем Поповою О.В. під керівництвом старшого викладача кафедри МПЗ Стрюка А.М.

Створювана система має складатися з 15 основних розділів – уроків – згідно з навчальною програмою дисципліни.

1. **Вступ.** Загальний погляд на операційні системи. Класифікація та структура операційних систем. Сучасні операційні системи, їх особливості.

2. **Підсистема керування процесами.** Визначення терміну “процес”. Опис процесу, стан процесу, керування процесами. Керування процесами та пам’яттю в сучасних операційних системах.

3. **Процеси і потоки.** Функціональність потоків. Багатопоточність. Потоки на рівні користувача та на рівні ядра. Симетрична багатопроцесорна обробка.

4. **Паралельні обчислення.** Принципи паралельних обчислень. Взаємні виключення: програмний підхід та апаратна підтримка. Використання семафорів, моніторів. Загальні принципи передачі повідомлень, їх синхронізація.

5. **Взаємне блокування процесів.** Принципи та причини взаємного блокування. Виявлення взаємного блокування. Запобігання та усунення взаємного блокування.

6. **Підсистема керування пам'яттю.** Загальні вимоги до керування пам'яттю. Розподілення пам'яті. Сторінкова та сегментна організація пам'яті.

7. **Віртуальна пам'ять.** Локалізація і віртуальна пам'ять. Сторінкова та сегментна організація. Комбінування сторінкової та сегментної організації. Захист та сумісне використання. Стратегія вибірки, стратегія розміщення, стратегія заміщення. Керування завантаженням.

8. **Планування.** Планування в системах з одним процесором. Типи планування процесора. Алгоритми планування.

9. **Багатопроцесорне планування та планування реального часу.** Багатопроцесорне планування процесів та потоків. Планування реального часу. Загальні характеристики систем реального часу.

10. **Керування пристроями вводу-виведення.** Пристрої вводу-виведення. Організація функцій вводу-виведення. Їх логічна структура. Використання та принципи буферизації.

11. **Планування дискових пристроїв.** Особливості роботи дискових пристроїв та необхідність їх планування. Існуючі алгоритми та апаратні технології.

12. **Файлова підсистема.** Поняття файлу. Системи управління файлами. Архітектура файлової системи. Функції керування файлами. Доступ до файлів. Каталоги файлів. Спільне використання файлів. Сучасні файлові системи.

13. **Розподіленні системи.** Розподіленні обчислення, архітектура клієнт/сервер. Кластери. Їх конфігурація та архітектура.

14. **Керування розподіленими процесами.** Міграція процесів. Механізми переносу процесів. Розподіленні глобальні стани.

Розподіленні взаємні виключення. Розподіленні взаємні блокування.

**15. Безпека.** Важливість безпеки. Зломщики та захист. Небезпека комп'ютерних вірусів. Захист від “троянських конів”.

Кожен з розділів буде розділено на тематичні підрозділи таким чином, щоб матеріали кожного підрозділу можна було легко розмістити на екрані комп'ютера. Це, на нашу думку, повинно полегшити сприйняття матеріалу та орієнтацію в системі. Перехід до наступного підрозділу можливий лише після того, як студент відповість на декілька запитань поданих у тестовій формі. Це повинно забезпечити контроль засвоєння опрацьованого матеріалу. Після кожного розділу студентам пропонується невеличка контрольна робота, під час якої студенти демонструють засвоєні ними знання. Якщо робота буде виконана незадовільно – студент повертається на початок розділу, а при задовільному результаті – має право перейти до вивчення наступного. В будь якому випадку система буде аналізувати відповіді студента і намагатися виявити слабкі сторони його знань. Якщо вони будуть знайдені – студенту буде запропоновано більш уважно проробити ті підрозділи, що були ним погано засвоєні. Ті розділи і підрозділи, що студент вже пройшов, тобто задовільно відповів на контрольні запитання, студент може проглядати в довільній послідовності. Для забезпечення цієї можливості, система буде створювати базу даних студентів, що працюють з нею, зберігаючи в ній результати роботи студента з системою. За бажанням студента система може дозволити повторно пройти те чи інше контрольне завдання і поліпшити попередню оцінку. Сам матеріал кожного розділу буде складатися з трьох частин:

1. *Основний матеріал*, який буде обов'язковим для вивчення, за яким будуть складені контрольні завдання.

2. *Додатковий матеріал*, де буде подана додаткова інформація для тих, хто хоче глибше засвоїти матеріали розділу. Тут студент зможе знайти багато цікавої інформації, але вивчення її не обов'язкове і засвоєння її системою не контролюється.

3. *Завдання для самостійної роботи* – зацікавленому у власних знаннях студенту надається змога поглибити їх, виконавши різні додаткові завдання. Але сама система контролювати їх виконання не буде.

Наприкінці курсу на студента чекає екзамен. Це і контроль засвоєння всього матеріалу і підготовка студента до реального екзамену, що чекає на нього наприкінці семестру.

Планується саме різноманітне використання цієї програми. Вона буде доступна для студентів в комп'ютерних класах університету, її можна буде використовувати на домашніх комп'ютерах. Для всіх бажаючих самостійно вивчати теорію операційних систем, навчаюча система буде розташована в мережі Internet на сайті [www.asterra.by.ru](http://www.asterra.by.ru) і, згодом, на сайті університету, який зараз знаходиться в стадії розробки.

Найважливішим є те, що в майбутньому планується розробка дистанційного курсу на базі навчаючої системи. Створювана система буде адаптована таким чином, щоб при роботі з нею мали змогу активно взаємодіяти як студенти так і викладач (т'ютор).



## **ПІДГОТОВКА СТУДЕНТІВ СПЕЦІАЛЬНОСТІ “ПРОФЕСІЙНЕ НАВЧАННЯ” ДО ВИКОРИСТАННЯ НОВИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ НАВЧАННЯ**

М.І. Стрюк

м. Кривий Ріг, Криворізький технічний університет

В сучасній педагогічній науці справедливо вважають, що студентів педагогічних спеціальностей необхідно озброювати прикладними, суто технологічними знаннями, що дало б можливість подолати наявний розрив між теоретичними знаннями і вміннями та навичками їх практичного застосування. Освоєння нових освітніх технологій означає не лише поповнення теоретико-методологічних знань студентів, а й формування професійних умінь проектувати, конструювати процес навчання, аналізувати його результати відповідно до нових інформаційних технологій навчання. Тому розв’язання проблеми формування готовності майбутнього інженера-педагога до роботи на основі глибокого розуміння сучасних педагогічних технологій має досить велике практичне значення. Технологічна грамотність такого фахівця дала б йому змогу глибше усвідомити своє істинне покликання, реально оцінити потенційні можливості, поглянути на педагогічний процес не лише з точки зору його перебігу, а й з позиції його кінцевого результату та способів його досягнення.

Проте, сьогодні проблема технологічної грамотності і культури інженера-педагога ще недостатньо розроблена в науці і практиці вітчизняної педагогічної освіти. Ця обставина обумовлює необхідність засвоєння студентами тих теоретико-методологічних основ технології навчання, що вже напрацьовані, і дають можливість самостійно вирішувати проблему розробки нових інформаційних технологій навчання і їх практичного застосування. У широкому вжитку поняття «технологія» означає сукупність прийомів і способів обробки або виробництва певних продуктів, або ж науку про такі способи. Стосовно сфери освіти про технологічні аспекти навчання писав ще Ян Амос Коменський, а стосовно процесу виховання уперше термін «технологія» запропонував А.С. Макаренко. У 60-х роках ХХ століття завдяки працям Б. Блума, Д. Брунера, Дж. Керола, С. Сполдінга,

Д. Хамбліна, Ю. Бабанського, В. Беспалька, П. Гальперина, Н. Щуркової, а в українській педагогіці – А. Алексюка, В. Бондаря, В. Вонсович, В. Лозової, І. Підласого, А. Фурмана розпочинається запровадження педагогічних технологій у практику. «Концепція педагогічної освіти» 1999 року звертала увагу вчених і практиків на посилення технологічного аспекту підготовки педагогів, на накопичену сучасну палітру освітніх технологій, напрацьовані підходи до гуманізації педагогічної взаємодії педагога з учнем і колективом у цілому.

Вивчення перелічених матеріалів дозволяє студентам сформувати уявлення про технологію навчання як упорядковану сукупність і послідовність методів і процесів, що забезпечують реалізацію проекту дидактичного процесу і досягнення діагностованого результату; виділити системність, науковість, концептуальність, відтворюваність, діагностичність, ефективність навчання, його вмотивованість, алгоритмічність, інформаційність, оптимальність як основні характеристики технологій навчання. На теоретико-методологічному рівні студенти переконуються, що технологічному підходу притаманна спрямованість на вдосконалення діяльності навчання, підвищення її результативності, інструментальності, інтенсивності, що технологія педагогічної діяльності враховує об'єктивні дидактичні закономірності і, таким чином, забезпечує в конкретних умовах відповідність результату діяльності попередньо поставленим цілям. До того ж технологічність освітньої діяльності якнайкраще узгоджується із освітньою політикою держави, яка акцентує увагу на розвиткові особистості, її якостей, талантів і здібностей, задоволені особистісних потреб, інтересів та інше.

Ідеї особистісно і практично орієнтованої підготовки майбутніх педагогів присвятили свої праці теоретико-методологічного рівня вітчизняні психологи і педагоги І.Д. Бех, О.С. Падалка, І.Ф. Прокопенко, В.І. Євдокимов. Одним з провідних завдань вони вважають створення такого освітньо-розвивального середовища, у результаті взаємодії з яким у майбутнього педагога формується готовність до роботи на основі знання сучасних педагогічних технологій, розуміння ним своєї індивідуальної сутності, на основі якої виробляється особистісна педагогічна концепція і персональна технологія.

Проте, крім сформованості необхідної мотивації до засвоєння і використання сучасних технологій навчання, майбутньому інженеру-педагогу необхідно орієнтуватися в їх широкому спектрі, цілях, концептуальних положеннях, особливостях методики. З цієї точки зору надзвичайно важливим є навчальний курс «Освітні технології», запропонований доктором педагогічних наук Оленою Миколаївною Пехотою та її колегами. Основні положення курсу викладені у навчально-методичному посібнику «Освітні технології», виданому у Києві 2002 року. Курс має як дидактичну мету – знайомство студентів із широким спектром напрацьованих наукою і практикою педагогічних технологій, так і розвивальну – набуття навичок і досвіду здійснювати педагогічну діяльність в різноманітних концептуальних системах. Цінною є практична спрямованість курсу, що створює необхідні умови для перетворення студента з об'єкта професійної підготовки на суб'єкт індивідуального професійного розвитку. У посібнику представлені як загальноновизнані особистісно-орієнтовані освітні технології, так і технології, що активно розвиваються на сучасному етапі, а також малознайомі педагогічному загалу.

Певний досвід вивчення новітніх педагогічних технологій у відповідному спецкурсі накопичено і кафедрою інженерної педагогіки Криворізького технічного університету. З огляду на те, що основними характеристиками технологічного підходу у навчанні є постановка діагностичної мети із визначенням рівня засвоєння, здійснення об'єктивного контролю ефективності навчання і визначення рівня досягнення поставленої мети, досягнення кінцевого результату, особливу увагу приділено оволодінню студентами новітніми інформаційними технологіями навчання.

Педагогіка визначає нові інформаційні технології навчання як методологію і технологію навчально-виховного процесу з використанням новітніх електронних засобів навчання, й у першу чергу ЕОМ. Розробка комп'ютерних технологій навчання в Україні почалася з середини 70-х років ХХ століття переважно у вищій школі. Поштовхом до появи нових інформаційних технологій навчання у масовій школі й ПТУ стали заходи щодо забезпечення комп'ютерної грамотності учнів середніх навчальних закладів і впровадження електронно-обчислювальної техніки в навчальний процес, які вживаються з 1985 року відповідно до

урядових рішень. І хоча, незважаючи на величезний педагогічний потенціал нові інформаційні технології навчання не зайняли ще достойного місця у сучасній загальній і професійній школі, проте студентів – майбутніх інженерів-педагогів має надихати те, що слідом за процесом інформатизації суспільства, що виявляється у активному повсюдному використанні інформаційної техніки для виробництва, переробки, збереження і поширення інформації й особливо знань, обов'язково розгорнеться і активний процес інформатизації освіти, як процес забезпечення сфери освіти теорією і практикою розробки й використання сучасних нових інформаційних технологій, орієнтованих на реалізацію психолого-педагогічної мети навчання і виховання.

У підготовці студентів до застосування нових інформаційних технологій необхідно виходити з їх розуміння як сукупності методів і технічних засобів збирання, організації, збереження, опрацювання, передачі й подання інформації, що розширює знання людей і розвиває їхні можливості щодо керування технічними і соціальними проблемами. З цього визначення витікає і логіка подальшого вивчення нових інформаційних технологій студентами, оскільки їх складовими є цілі навчання, його закономірності і принципи, методи та засоби нових інформаційних технологій. Так, цілі навчання, його мета визначаються впливом інформатизації суспільства і тому основною метою нових інформаційних технологій навчання стає підготовка учнів до повноцінної життєдіяльності в умовах інформаційного суспільства, що передбачає формування умінь роботи з інформацією, дослідницьких умінь, умінь приймати оптимальні рішення, здійснювати цілісне інформаційне забезпечення, нарешті формування інформаційної культури учнів, розвиток їх творчого потенціалу, здібностей до комунікативних дій. Ці завдання мають вирішуватися у відповідності як із загально-дидактичними закономірностями і принципами навчання, так і з закономірностями та принципами навчання властивими новим інформаційним технологіям навчання, як то оптимальне поєднання індивідуальної і групової роботи, підтримка психологічного комфорту, принципами адаптивності, управління та іншими.

Широкі можливості надають і особливості властиві методиці нових інформаційних технологій навчання. Вони передбачають

застосування таких методів як системний аналіз, проектування, методи передачі, збереження та захисту інформації, безпаперові технології, методи колективного використання різноманітних інформаційних ресурсів. Причому застосування цих методів передбачає як фрагментарне використання комп'ютерів на уроках як тренажера або для демонстрації чи контролю знань, так і для дослідно-експериментальної роботи, дистанційного навчання, використання довідкових систем та виходу у інформаційні мережі.

Звичайно, найважливішою складовою підготовки студентів до використання нових інформаційних технологій є досконале оволодіння ними основними засобами нових інформаційних технологій, як апаратними (ЕОМ, персональні ЕОМ, локальні і глобальні мережі, пристрої введення-виведення, засоби збереження інформації, інше периферійне обладнання), так і програмними (програмні комплекси, інформаційні системи, системи машинної графіки, системи мультимедіа, системи штучного інтелекту тощо). На основі засвоєння теоретичного матеріалу, вироблених вмінь та навичок володіння апаратними та програмними засобами студенти працюють над складанням проектів уроків із позицій нових інформаційних технологій, що дозволяє їм набути первинний досвід практичного застосування нових інформаційних технологій навчання. Складені проекти уроків не лише обговорюються на практичних заняттях, а й обов'язково використовуються під час педагогічної практики з наступним аналізом результатів їх застосування. Таким чином, глибоке розуміння суспільного та педагогічного значення, мети та завдань нових інформаційних технологій, знання закономірностей та принципів властивих їм, належне володіння їх методикою, апаратними та програмними засобами дозволить студентам у майбутньому ефективно вирішувати проблему підготовки фахівців у галузі інформатики та обчислювальної техніки, підготовки користувачів засобів нових інформаційних технологій.

## **ПРИМЕНЕНИЕ ОБУЧАЮЩЕЙ ПРОГРАММЫ ЭМУЛЯЦИИ DOS ПРИ ИЗУЧЕНИИ КУРСА «ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И ПРОГРАММИРОВАНИЕ»**

Ю.А. Супрунова

г. Кривой Рог, Криворожский металлургический факультет На-  
циональной металлургической академии Украины  
j@alba.dp.ua

Как известно, одной из главных проблем изучения команд DOS является безопасность проведения лабораторных работ. Неподготовленный пользователь может неосторожным нажатием клавиш доставить много проблем системному администратору компьютерных классов. К тому же процесс форматирования или разбиения жесткого диска на разделы исследовать «вживую», мягко говоря, нецелесообразно. Однако без знания некоторых утилит и элементарных приемов работы в «чистом DOSe» нельзя подготовить грамотного пользователя. В связи с этим, мы предлагаем использовать программу, написанную нами на языке Turbo Pascal 7.0, которая хотя и не является полной эмуляцией среды DOS, но все же позволяет изучить основные команды для работы с файлами и каталогами, а также команды для работы с дисками. Еще одним преимуществом программы являются интерактивные подсказки – сообщения об ошибках, которые по сравнению со стандартными сообщениями операционной системы более конкретизированы и позволяют студенту исправить свои ошибки при следующей попытке.

Кроме того, программа проводит контроль знаний студента и, в зависимости от затраченного времени и количества сделанных ошибок, оценивает результат. Фамилия студента и его оценки записываются в текстовый файл.

Задания, которые необходимо выполнить студенту появляются в специальном окне в верхней части экрана. В остальном среда программы ничем не отличается от среды DOS.

В стандартном приглашении DOS отображается «текущий каталог». Программа «запоминает» названия каталогов и файлов, заданные пользователем и это учитывается при последующих

заданиях. В случае неправильного ввода имени файла или папки на экран выводится соответствующее сообщение. При вводе пользователем имени файла или папки проводится также проверка на недопустимые символы.



Команды DOS общесистемного назначения – например, команда CLS – очистка экрана, DATE – вывод либо установка текущей даты, TIME – ввод и установка системного времени, VER – вывод сведений о версии операционной системы, PATH – вывод либо установка пути поиска исполняемых файлов являются «резидентными», т.е. доступны в любой момент. При вводе команды EXIT происходит завершение работы программы без записи в файл результатов.

Набрав в командной строке стандартный запрос:

**[имя команды]/?** можно вывести на экран справку о любой из команд DOS. Информация о командах содержится в отдельных текстовых файлах. Причем файлы содержат более подробные сведения о командах, чем стандартная справка DOS. Таким образом, программа позволяет студентам на практике освоить такие команды для работы с файлами и каталогами, как COPY, CD, MD, RD, TYPE, REN, DEL. При этом учитываются ключи команд, которые определяют дополнительные функции команд.

Особый интерес, на наш взгляд, представляет блок заданий,

предназначенных для изучения команд работы с дисками и системными файлами (FORMAT, LABEL, VOL, FDISK, SYS, DISKCOPY).

Программа позволяет моделировать такие ситуации, как форматирование жесткого диска, разбиение его на логические разделы, копирование системных файлов безо всякого риска и ущерба для установленного программного обеспечения.

В заключение следует отметить, мы считаем, что теоретическое изложение материала никогда не сможет заменить практики использования тех или иных команд. Столкнувшись в реальных условиях с подобной задачей, студенты будут обладать определенными практическими навыками.



## ТЕСТУВАННЯ НАВЧАЛЬНОЇ УСПІШНОСТІ, ЯК ФОРМА ОРГАНІЗАЦІЇ ПОПЕРЕДНЬОГО КОНТРОЛЮ ЗНАТЬ СТУДЕНТІВ НА ЛАБОРАТОРНИХ ЗАНЯТТЯХ З ІНФОРМАТИКИ

І.Є. Фільо

м. Рівне, Український державний університет водного господарства та природокористування

Обов'язковою умовою підвищення якості навчання та стимулювання пізнавальної активності студентів є систематичний контроль засвоєння знань. Для виявлення сприймання студентами навчального матеріалу на лекції та під час самостійної роботи, може служити добре організований контроль на лабораторних та практичних заняттях. Розрізняють попередній, поточний, тематичний і підсумковий види контролю успішності. Як показує практика, на лабораторних та практичних заняттях з інформатики більш ефективним є попередній контроль знань студентів та їх підготовка до виконання роботи. Це, по-перше, стимулює студентів до відвідування лекцій з дисципліни, по-друге, привчає до самостійної роботи за межами університету, по-третє, навчає найкраще використовувати час роботи за комп'ютером. Викладач лише повинен вирішити, які методи попереднього контролю дають можливість повніше перевірити рівень засвоєння знань студентами.

Сьогодні, поряд із використанням традиційних форм і засобів організації контролю, доцільним є проведення *тестування навчальної успішності* (спеціально підготовлених завдань для виявлення знань студентів, що потребують коротких відповідей). Саме тестова перевірка, як метод контролю дає більше можливостей для перевірки самостійної роботи студентів і дозволяє керувати процесом засвоєння знань. Тестова і комп'ютерна система контролю має такі переваги:

- скорочує час на перевірку;
- ставить до всіх учнів однакові вимоги;
- усуває ефекти суб'єктивізму;
- сприяє дотриманню єдності вимог до оцінки знань;
- стимулює до самооцінки знань;

- об'єктивність оцінки не настроює студента проти викладача, а на подолання прогалин у знаннях;
- виключає випадковість в оцінці знань;
- дозволяє статистичну обробку одержаних результатів.

Недоліки цього методу перевірки, такі як, виявлення знання фактів, але не здібностей, усуваються впродовж виконання лабораторної роботи за комп'ютером. Саме на початку виконання лабораторної роботи важливими є засвоєнні теоретичні знання з курсу інформатики, а вже потім студенти формують навички роботи з тими чи іншими програмними продуктами.

Тестові завдання, що потребують конструювання відповідей, можна поділити на такі типи:

- завдання з простим вибором одноелементних відповідей;
- завдання з простим вибором багатоелементних відповідей;
- завдання з перехресним вибором одноелементних відповідей;
- завдання з перехресним вибором багатоелементних відповідей;
- завдання з поетапним вибором відповідей;
- завдання з альтернативними відповідями;
- завдання на заповнення пропусків;
- завдання на конструювання правильної відповіді.

Щоб максимально наблизити завдання до діалогу “викладач – студент”, необхідно використовувати такі прийоми конструювання тестів, які б допомагали перевіряти вміння студентів:

- правильно відтворювати набуті знання;
- характеризувати з різних боків або знаходити схоже в явищах, які вивчаються;
- вільно орієнтуватися в групі схожих понять, процесів, явищ;
- робити правильний вибір або приймати рішення у згорнутій формі;
- набувати чітке, однозначне розуміння явищ, процесів, понять.

Задовольнити такі вимоги на лабораторних заняттях по інформатиці допомагають тести, що містять у собі різноманітні

типи завдань. Як показує досвід, найліпшим є використання завдань з простим вибором одно- та багатоелементних відповідей, з перехресним вибором одноелементних відповідей, з альтернативними відповідями та завдання на заповнення пропусків.

Так, на поставлені в завданні питання студент повинен знайти в запропонованій сукупності відповідей правильні або для кожного питання, розміщеного в лівому стовпчику, вибрати однозначну відповідь із правого стовпчика. Поряд із правильними відповідями в тесті є і неправильні відповіді. Однак усі відповіді перемішані, і це виключає угадування. Таким чином, можливий лише цілеспрямований усвідомлений пошук правильної відповіді. У різних варіантах тесту можуть міститися однакові сукупності правильних і неправильних відповідей, однак вони розташовані в різних порядках (що дуже легко здійснити при комп'ютерному тестуванні) і мають різні коди, це може знизити можливість списування. Питання може ставитися у вигляді послідовності виконання команд і пропозиції завершити чи доповнити цю послідовність. Вірну відповідь можна прийняти за свідчення про знання не тільки основних понять, але і різних методів виконання завдання, що не маловажно при формуванні умінь роботи з програмним забезпеченням ЕОМ. Кожна лабораторна або практична робота з інформатики може містити такий етап тестування, який допоможе викладачу здійснити попередній контроль знань і підготовленості студентів до виконання лабораторних робіт.





Описана вище методика формування завдань для попереднього тестового контролю на лабораторних та практичних заняттях з інформатики є особливо ефективною при вивченні сучасних текстових редакторів, електронних таблиць, систем керування базами даних тощо. На допомогу викладачам інформатики для випробовування даного підходу пропоную тестові завдання для вивчення окремих можливостей електронних таблиць Microsoft Excel.

### **Тестовий контроль для вивчення основних можливостей електронних таблиць Microsoft Excel.**

Тема 1. Загальні відомості про електронну таблицю Microsoft Excel.

1. Документи, що створюються в середовищі Excel, називають:
  - a) листи; b) листи макросів;
  - c) книги; d) робочі листи.
2. За допомогою якого пункту меню користувач може змінити ім'я активного листа:
  - a) Файл; b) Правка; c) Окно; d) Формат.
3. Чим характеризується клітинка:
  - a) Адресою; b) Значенням; c) Об'ємом пам'яті;
  - d) Вмістом (текст, формула); e) Форматом.
4. Як утворюються або ідентифікуються структурні елементи ЕТ?
  1. Рядки
    1. Ідентифікуються за допомогою букв, які розміщуються по вертикалі у лівій частині екрана
  2. Клітинка
    2. Задається адресами лівої верхньої і правої нижньої клітинок
    3. Утворюється на перетині рядка і стовпця.
5. Як скопіювати виділений діапазон клітинок:
  - a) Правка→?→?;
  - b) ?→Копирование→?→?;
  - c) ?→Копирование→Правка;
  - d) ?→?→Вставить.

Тема 2. Використання формул і функцій в Microsoft Excel.

1. Яку кнопку використовують для активізації Майстра функцій:
  - a) 
  - b) 
  - c) 
  - d) 
2. Що містять формули:
  - a) Числа;
  - b) Знаки операцій;
  - c) Адреси клітинок;
  - d) Функції;
  - e) Розміри клітинок.
3. Яка з наведених функцій належить до категорії математичних функцій:

- a) КОРЕНЬ; b) МАКС; c) НЕ; d) ЕСЛИ.
4. Яка з наведених статистичних функцій використовується для обчислення середнього значення:  
a) МИН; b) МОДА; c) МАКС; d) СРЗНАЧ.
5. Яке значення виразу  $\text{МАКС}(15;20; \text{МИН}(45;25))$ :  
a) 15 b) 30 c) 20 d) немає вірної відповіді

Особливості систематичного контролю, здійснюваного за допомогою тестування (як комп'ютерного так і безмашинного), краще стимулюють студентів до навчання, спонукають до більшої витрати часу, щоб самостійно опрацювати лекційний матеріал та раціонально використовувати аудиторний час. Багаторазове використання тестування на лабораторних заняттях з інформатики, як під час аудиторних занять, так і у самостійній роботі студентів, сприяє підвищенню якості та ефективності сприймання лекційного матеріалу, зміцнює зворотній зв'язок “викладач – студент”.

#### Література

1. Адаменко О.В., Духовна М.М., Панченко Л.Ф., Кондратенко П.В. Тестові завдання для контролю знань в курсі “Обчислювальна техніка і технічні засоби навчання”: Навч.-метод. посібник. – К., 1996. – 84 с.
2. Інформатика. 10-11 класи. Тематичне оцінювання. / Упорядник: Фролова В.С., Семчук А.І., Царенко Н.В., Понятішин В.Г. – Тернопіль: Астон, 2002. – 76 с.
3. Фіцула М.М. Педагогіка: Навч. посібник для студентів вищих пед. закладів. – Тернопіль, 1999. – 198 с.

# РОЗВ'ЯЗОК СИСТЕМИ ЛІНІЙНИХ АЛГЕБРАЇЧНИХ РІВНЯНЬ В ЕЛЕКТРОННИХ ТАБЛИЦЯХ MICROSOFT EXCEL

І.Є. Фільо

м. Рівне, Український державний університет водного господарства та природокористування

В умовах науково-технічної революції, інтенсивного збільшення обсягу інформації, відновлення бази знань особливого значення набуває підготовка фахівців, що мають високий загальнонауковий і професійний рівень. Сьогодні важко представити сучасного фахівця без базових знань в області математики, інформатики, економіки й інших дисциплін. Найбільш ефективним способом досягнення й зміцнення цих знань є застосування обчислювальної техніки на кожному етапі навчання студентів. Уміння розв'язувати задачі на ЕОМ є однією з головних практичних цілей курсу інформатики в технічному університеті і дає можливість студентам розширити свої знання в області фундаментальних наук. Практичні та лабораторні роботи курсу інформатики повинні бути націлені не тільки на придбання навичок роботи з операційною системою й офісними програмами. Курс також повинен передбачати комп'ютерний розв'язок задач із курсу вищої математики, фізики, економіки й інших фундаментальних дисциплін.

Діяльність студента на лабораторних та практичних заняттях з інформатики полягає в оволодінні навичками, які допомагають йому, наприклад, при вивченні вищої математики, а також забезпечує інтеграцію цих навичок у більш широку систему майбутньої фахової діяльності. Таким чином можна сказати, що застосування комп'ютера для розв'язку типових математичних задач передбачає набуття студентом навичок:

- по-перше, точного дотримання приписів математичних операцій;
- по-друге, оперативного здійснення аналітичних перетворень та проміжних обчислень;
- по-третє, розуміння змісту математичних операцій;
- по-четверте, побудови точніших математичних моделей.



має один розв'язок. Помножимо ліву і праву частину (2) на  $A^{-1}$ , тоді  $A^{-1} \cdot A \cdot X = A^{-1} \cdot B$ , де  $A^{-1} \cdot A = E$  (одична матриця).

Після необхідних перетворень розв'язок лінійної системи *матричним методом* матиме вигляд

$$X = A^{-1} \cdot B, \quad (3)$$

де  $A^{-1}$  – обернена матриця до  $A$ . У нашому випадку  $A = \begin{pmatrix} 8 & 3 \\ 2 & 7 \end{pmatrix}$  і

$$B = \begin{pmatrix} 4 \\ 2 \end{pmatrix}.$$

Хід роботи:

1. Запустіть програму MS Excel.
2. У клітинку A1 введіть текст “Матриця коефіцієнтів”, у D1 – “Вільні члени”, у F1 – “Розв'язок”, у A4 – “Обернена матриця”.

3. У блок клітин A2:B3 введіть матрицю коефіцієнтів  $A$ , у блок клітин D2:D3 введіть стовпчик вільних членів.

4. Виокремте блок клітин (A5:B6), де будуть обчислені коефіцієнти оберненої матриці до  $A$ . Застосовуючи Мастер функцій, категорія Математические, вибрати функцію **МОБР** і занести відповідні значення елементів матриці  $A$  – A2:B3, натиснути кнопку ОК. Після цього встановити курсор на рядок формул, натиснути клавіші **Ctrl+Shift+Enter**.

5. Виділити діапазон клітин, де буде знайдено розв'язок – F2:F3. Застосовуючи Мастер функцій, категорія Математические, обирати функцію **МУМНОЖ** і занести відповідний діапазон матриці  $A^{-1}$  – A5:B6 та діапазон матриці  $B$  – D2:D3, натиснути кнопку ОК. Після цього встановити курсор на рядок формул, натиснути клавіші **Ctrl+Shift+Enter**.

6. У блоці клітин F2:F3 отримати розв'язок системи лінійних рівнянь. Результат роботи наведено:

Матриця коефіцієнтів		Вільні члени		Розв'язок
8	3	4		0,44
2	7	2		0,16
Обернена матриця				
0,14	-0,06			
-0,04	0,16			

7. Зберегти документ на диску.



Наведена інструкція перш за все передбачає інтенсифікацію та активізацію навчального процесу на практичній і лабораторній роботі з інформатики. За умов того обмеженого часу, який відводиться для вивчення дисципліни, важливе значення відіграє застосування таких засобів наочності.

Отже, застосування електронних таблиць Microsoft Excel для розв'язку типових математичних задач та друкованих роздаткових матеріалів дозволить студентам розширити та зміцнити свої знання, набуті при вивченні курсу вищої математики, а слабо підготовленим студентам на рівні з іншими оволодіти матеріалом теми. Викладачеві це дозволить максимально ефективно використовувати час та реалізувати принцип залучення студентів до навчальної діяльності незалежно від рівня їхніх попередніх знань із деяких розділів курсу вищої математики.

#### Література

1. Вища математика: основні означення, приклади і задачі. Навчальний посібник. Книга 1. /Г.Л. Кулініч, Л.О. Максименко та ін. – К.: Либідь, 1994. – 312 с.
2. Бронштейн И.Н., Семендяев К.А. Справочник по математике для инженеров и учащихся втузов. – М.: Наука, 1964. – 608 с.
3. Колмогоров А.Н., Ивашев-Мусатов О.С. и др. Алгебра и начала анализа, 10 кл. – М.: Просвещение, 1978. – 271 с.
4. Гарнаев А.Ю. Excel, VBA, Internet в экономике и финансах. – СПб.: БХВ-Петербург, 2002. – 816 с.

## **ИСПОЛЬЗОВАНИЕ ВИРТУАЛЬНЫХ МАШИН В КУРСЕ «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ЭВМ»**

А.А. Хараджян

г. Кривой Рог, Криворожский государственный педагогический университет  
khaa@kpi.dp.ua

При организации учебного процесса в компьютерных аудиториях всегда стремятся свести к минимуму возможность повреждения операционной системы и программного обеспечения, установленного на компьютерах, как случайно, так и преднамеренно (поскольку установка всего необходимого программного обеспечения требует от 3 до 5 часов). Для предотвращения повреждения ПО используют два подхода: организационный и технический. В первом случае ужесточается контроль со стороны преподавателей и лаборантов, но в большинстве случаев это оказывается неэффективным. Во втором случае устанавливается операционная система, позволяющая ограничить доступ пользователей к файлам. При этом студенты имеют доступ только к рабочему каталогу, а доступ к системе делается только для чтения. Для решения этих задач широкое применение нашли системы Windows NT и Linux. Эти системы позволяют достаточно гибко управлять доступом к ресурсам системы. Однако использование этих систем затрудняет проведение лабораторных работ по курсам, предусматривающим полный доступ к файловой системе и операционной системе (по разделам «Файловые системы», «Администрирование операционной системы»), т.к. любые административные изменения (форматирование жесткого диска, изменение режима загрузки и др.) в работающей системе недопустимы. Содержать отдельный класс или даже несколько машин для проведения лабораторных работ такого характера – недопустимая роскошь в связи с необходимостью после фатальных ошибок переустанавливать операционную систему.

Наиболее эффективным решением этой проблемы является применение виртуальных машин, например VMWare. Данная программа позволяет создать виртуальную машину как в ОС Windows NT, так и в Linux.

Разработанный нами лабораторный практикум по курсу «Программное обеспечение ЭВМ» рассчитан на работу в ОС Linux. Для проведения лабораторных работ была установлена программа VMWare версии 3.0. В данной виртуальной машине создан диск размером 1 Гб, который был распределен следующим образом: 300 Мб отведено для основного раздела Ext3, 100 Мб – для раздела подкачки Swap, оставшееся пространство оставлено свободным для проведения лабораторных работ. В раздел Ext3 установлена ОС ALT Linux 2.0 с минимально необходимым набором программного обеспечения. Студенты имеют административный вход в эту операционную систему (root). Копия файла виртуального диска хранится на сервере, и при необходимости может быть переписана на любую машину в аудитории.

Такой подход обеспечил быструю восстанавливаемость системы в случае ее повреждения. Кроме того, существенно уменьшилось время необходимое на восстановление операционной системы.

В качестве примера рассмотрим лабораторную работу «Работа с файловой системой».

1. На виртуальном диске sda свободное пространство отсутствует. Поэтому для создания нового раздела необходимо отключить и удалить раздел swap. Создать файл подкачки swap и подключить его к системе.

1.1. Отключение активного раздела подкачки.

```
swapoff /dev/sda2
```

1.2. Создание нового файла подкачки в разделе sda1

```
dd if=/dev/zero of=/var/swapfile ds=1024 count=32768
```

```
mkswap /var/swapfile
```

1.3. Активизация нового файла подкачки

```
swapon /var/swapfile
```

1.4. Удалить раздел sda2, используя программу fdisk.

2. Создать новый DOS-раздел, равный половине свободного пространства диска, используя программу fdisk.

3. Создать новый Ext-раздел, равный половине свободного пространства диска, используя программу fdisk.

4. Отформатировать созданные разделы.

5. Используя программу hdparm, просмотреть параметры

диска

```
hdparm /dev/sda
```

6. Монтирование файловой системы.

При монтировании корневой каталог монтируемой файловой системы подключается к одному из каталогов корневой файловой системы.

```
mount <устройство> <каталог>
```

Первый параметр является именем файла-устройства для монтируемого логического устройства, содержащего подключаемую файловую систему. Второй – имя уже существующего каталога, под которым монтируется файловая система.

Смонтировать файловую систему дискеты в каталог /home/root/disketa

Чтобы выяснить, какие файловые системы смонтированы в данный момент, надо вызвать команду `mount` без параметров.

7. Демонтирование файловой системы

Если файловая система больше не используется, ее можно демонтировать командой `umount`

```
umount <устройство>
```

Результатом демонтажа является разрыв связи между корневым каталогом демонтируемой файловой системы и каталогом корневой файловой системы, в котором производился монтаж.

8. Синхронизация файловых систем

При обменах с файлами система поддерживает в оперативной памяти множество буферов, свободных блоков, `inode`, а также супер-блок. Для корректного завершения операций с системой необходимо перед ее выключением переписать все данные из этих буферов и блоков в дисковую память. Выполнение этих действий называется синхронизацией файловой системы и реализуется командой `sync`:

```
sync
```

9. Используя команду `ln`, создайте жесткую и символическую ссылки.

## ДОСВІД РОЗРОБКИ АВТОМАТИЗОВАНОЇ НАВЧАЛЬНОЇ СИСТЕМИ

М.О. Цензура

м. Київ, Київський національний торговельно-економічний університет

Зростання обсягу інформації, що відводиться на самостійне вивчення дисциплін, зменшення кількості лекційних занять як на денній так і на заочній формі навчання, вимагає розробки і впровадження в навчальний процес нових технологій організації самостійної роботи студентів.

Запропонована автоматизована навчальна система (АНС) є програмною оболонкою, наповнення якої визначається викладачем, що веде дисципліну. При повній реалізації усіх можливостей, наявних в АНС, студенти одержують можливість в інтерактивному режимі знайомитись з конспектом лекцій, новими поняттями і співвідношеннями між ними, самостійно контролювати рівень своїх знань через розв'язування типових задач і тестування.

Навчальна система універсальна, орієнтована на самостійне навчання студентів по будь-якому предмету та курсу.

Перед початком розробки АНС під Windows була поставлена задача застосувати:

- у електронному конспекті лекцій гіпертекстові посилання, графічні об'єкти, формули, мультимедії та списки ключових слів та фраз.

- для пояснення рішень типових задач з тем застосовувати об'єкти типа таблиці, графіки, діаграми, формульні вирази, а також забезпечити можливість переходу по гіперпосиланню до відповідного розділу електронного конспекту.

З цією метою були проаналізовані можливості застосування настільних СУБД. Аналіз показав, що при створенні навчальної системи найбільш прийнятним буде застосування СУБД Access 97 або Access 2000, як сховищ інформації. У СУБД прийнято, що великий текст може зберігатись у текстовому форматі ASCII. З іншого боку, є можливість зберігати відформатовані тексти та об'єкти Windows, як впроваджені об'єкти OLE.

Тому було вирішено, програмне забезпечення написати у середовищі Visual Basic 6.0. Задачі, тести та пояснені до них зберігати у таблицях СУБД Access, а також додатково застосувати ще два програмних засоби:

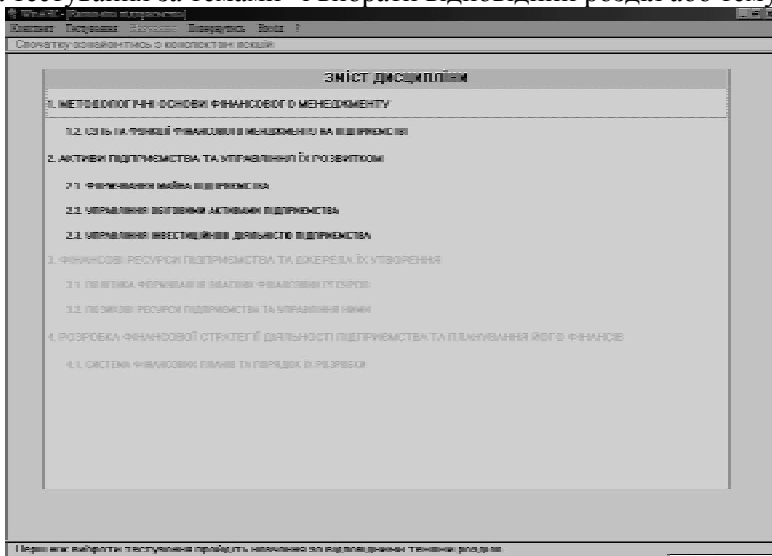
- технологію HTML Help для створення електронного конспекту лекцій;
- технологію WIN Help для створення пояснень до типових задач у стандарті HELP середовища Windows.

Програма АНС складається з двох програмних модулів:

- ◆ перший включає систему наповнення, коригування бази даних і налагоджування системи тестування;
- ◆ другий модуль – систему автоматизації процесу навчання.

У систему можна закладати курси з різних дисциплін економічного спрямування. Кожний курс розбивається на окремі розділи, а кожний розділ на теми. Теми можуть бути складовими частинами розділу курсу, а можуть бути самостійними, наприклад, вступ, висновки, глосарій, список використаної літератури і т.д.

Робота з АНС починається з вибору предмета, що вивчається. Для цього необхідно звернутися до меню і вибрати пункт “Вибір курсу “. Потім необхідно звернутися до пункту “Навчання та тестування за темами” і вибрати відповідний розділ або тему.

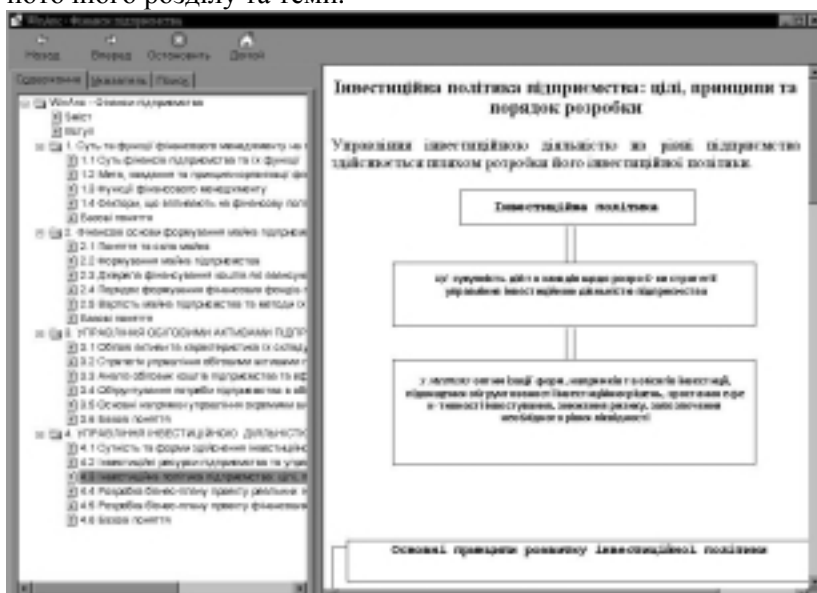


Система запропонує перейти в режим “Тестування” за матеріалами обраного розділу або в режим “Навчання”, якщо вибрана одна з тем розділу. Режим “Навчання” дозволяє переглянути:

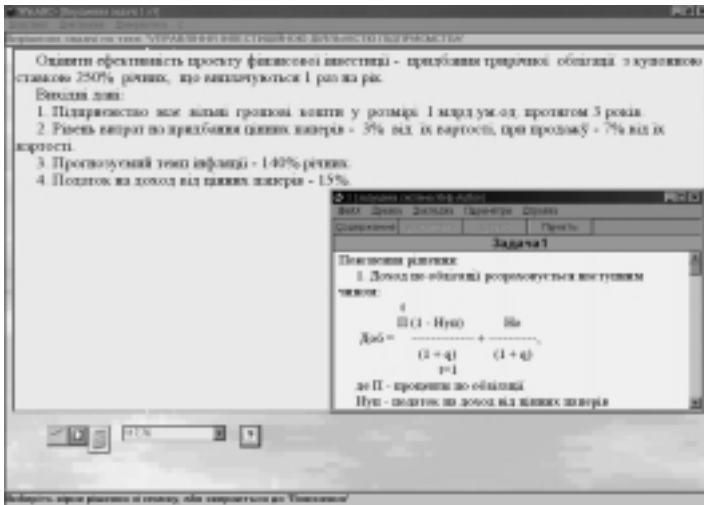
- зміст типових контролюючих тестів, що будуть винесені на тестування;
- типові розрахункові задачі.

У навчальній системі передбачено контроль за успішним проходженням контролюючих тестів. Якщо студент успішно пройшов контрольне тестування за поточним розділом, він допускається до вивчення наступного.

Теоретичний матеріал подається блоками в залежності від поточного розділу та теми.



Після самостійного вивчення теми студенту пропонується практично закріпити теоретичний матеріал через вирішення типових розрахункових завдань. Якщо студент недостатньо засвоїв матеріал, він має можливість по клавіші <F1> звернутися до вбудованої довідки й ознайомиться з послідовністю вирішення задачі.

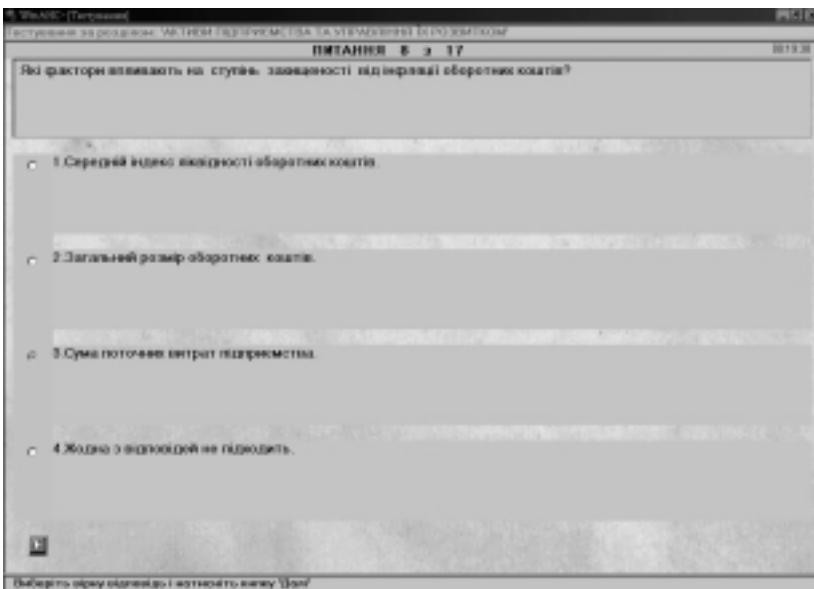


Для самостійного закріплення теоретичного матеріалу, студенту пропонується ознайомитись з навчальними тестами. Відмінність від контролюючих тестів полягає у наданні підказки “Вірна відповідь” або “Невірна відповідь” та при натисканні на клавішу <F1> отримувати розгорнуте пояснення.



Для закріплення матеріалу та отримання допуску до вивчення наступного розділу пропонується самостійно пройти тестування по обраному розділу.





Тестовий контроль здійснюється по методиці “питання-відповідь”, то б то студенту задається питання, а він повинен вибрати правильну відповідь з чотирьох варіантів. При кожному зверненні до системи тестування порядок відображення питань та порядок відповідей змінюється випадковим чином. Після вибору відповіді, студента позбавлено можливості повернутись до цього питання. При тестуванні враховується час на кожну відповідь, якщо протягом 30 секунд відповідь на питання неотримано, система рахує, що студент користується додатковою літературою і додає понижуючий ваговий коефіцієнт. Через 1 хвилину, після появи питання навіть вірна відповідь невраховується.

Після проведення контрольного тестування з розділу, оцінка не виставляється, а відображається повідомлення про загальну кількість питань, кількість та відсоток вірних відповідей, а також рекомендації щодо засвоєння навчального матеріалу.

РЕЗУЛЬТАТИ ТЕСТУВАННЯ			
за розділом: АКТИВИ ПІДПРИЄМСТВА ТА УПРАВЛІННЯ ЇХ РОЗВИТКОМ			
Кількість запитань: 17		Кількість вірних відповідей: 4	
Відсоток вірних відповідей: 23,53%			
Рівень знань по розділу недостатній. Вивчіть учбовий матеріал більш уважно.			
№	Питання	Вірно відповідей	Всього відповідей
1	Які з перелічених коштів підприємства не входять до складу об'язків?	2	4
2	Як визначається індекс прибутковості (рентабельності) проекту реальним?	2	4
3	Який з перелічених факторів (при інших рівних умовах і господарюванні)?	3	3
4	Який з перелічених коефіцієнтів ліквідності дозволяє дати оцінку?	1	2
5	Які з перелічених факторів не впливають на розмір потреби підприємства?	3	4
6	Що не є платіжним резервом у формі амортизаційних отримань підприємства?	1	3
7	Як з приведених ситуацій не може розглядатися як інвестиція?	2	2
8	Які фактори впливають на ступінь збереженості на інформації оборотних?	2	3
9	По фінансових інвестицій належать інвестиції, які:	3	4
10	По реальних інвестицій належать:	3	3
11	Яким чином формується статутний фонд державного підприємства?	3	2
12	Яку ознаку з точки зору управління ліквідністю слід дати ситуації, при якій	1	4

Якщо студент за навчальний матеріал поточного розділу отримав інтегровану оцінку якості знань менше 70%, то йому пропонується повторити вивчення розділу та блокується перехід до наступного.

Підсистема адміністрування призначена винятково для адміністратора системи та упорядників методичного забезпечення електронного курсу з дисципліни.

Підсистема адміністрування дозволяє:

- створювати базу знань для нового курсу;
- реєструвати користувачів;
- завантажувати підготовані txt-файли з практичними завданнями, тестами та розгорнутими відповідями у відповідні таблиці бази даних;
- налагодження модуля діагностики наслідків тестування шляхом завдання “Кількості тестів для оцінки набутих знань” та “Термін часу на проведення тестування” для кожного з розділу навчальної дисципліни;
- переглядати конспект, тестові питання та практичні завдання разом з поясненнями до вірних відповідей.

## ИЗЛОЖЕНИЕ РАЗДЕЛА EXCEL В КУРСЕ ИНФОРМАТИКИ ДЛЯ ЭКОЛОГОВ

С.Н. Чаплыгина

г. Днепропетровск, Приднепровская государственная академия  
строительства и архитектуры

Особенность пакета Excel состоит в его широких возможностях как для непосредственного построения электронных таблиц и графической их иллюстрации с помощью диаграмм, так и для выполнения широкого спектра вычислительных операций. Последнее, в частности, обуславливается большой библиотекой встроенных функций. Многие экологические задачи могут быть решены с использованием этого пакета. Поэтому в курсе информатики для экологов работе в этом пакете уделяется особое внимание.

Значительное количество экологических задач носят вычислительный характер и сводятся к счету по формулам. В этом случае использование Excel избавляет от необходимости составлять и отлаживать программы и процесс решения задачи состоит в непосредственном вводе формул.

Обработку данных можно осуществить и без компьютера, с калькулятором например, как это и делалось до недавнего времени. Такой процесс занимает много времени, но главный недостаток такого подхода состоит в том, что для повторения тех же вычислений с изменившимися исходными данными, потребуется столько же времени и труда. Кроме того, для наглядного представления данных в виде диаграмм и графиков, которые компьютер строит автоматически, пришлось бы затратить дополнительное время.

Многие численные методы решения математических задач, такие, например, как решение систем линейных алгебраических уравнений, трансцендентных уравнений, дифференциальных уравнений и систем дифференциальных уравнений легко реализуются в рассматриваемом пакете. К таким задачам сводится ряд экологических задач.

К особенностям задач экологии также можно отнести необходимость в обработке больших объемов информации. Возмож-

ность работы с большими массивами информации, большое количество встроенных функций для выполнения различных операций с массивами данных делает эффективным применение Excel и к подобным задачам.

Так, студенты в рамках специальных дисциплин проводят натурные измерения по определению загазованности отдельных районов города. Стандартные схемы расчетов, проводимых по результатам этих измерений, представляются в виде таблиц, часть столбцов в которых заполнены результатами измерений, а остальные являются результатами статистической обработки, которая может быть проведена средствами Excel. Такого типа задачи, как будто специально сформулированные под рассматриваемый пакет, еще совсем недавно проводились с помощью калькулятора. Можно было бы привести еще ряд подобных примеров.

Удобным средством решения оптимизационных задач является встроенное средство «Поиск решения». С его помощью можно решать задачи, сводящиеся к задачам линейного и нелинейного программирования. Требуется лишь правильно сформулировать задачу для Excel, и оптимальное решение будет найдено быстро и автоматически.

Наконец, практически во всех видах деятельности приходится создавать те или иные базы данных и работать с ними. В тех случаях, когда не предполагается сложных видов работ с большими базами данных имеет смысл воспользоваться способами создания и работы с базами, предлагаемыми пакетом Excel.

В курсе Информатики при изучении раздела работы с электронными таблицами студенты-экологи нашей академии выполняют следующие лабораторные работы:

1. Создание электронных таблиц, содержащих вычисляемые столбцы, и графическое представление имеющихся в них данных с помощью диаграмм.
2. Выполнение расчетов для математической и статистической обработки данных.
3. Реализация численных методов решения математических задач.
4. Решение задач оптимизации.

При выполнении лабораторных работ формируются умения

студентов составлять конкретный план действий, реализовывать эти действия, анализировать полученные результаты и, в случае несоответствия их ожидаемым, корректировать план действий.

К положительным аспектам использования Excel с методической точки зрения следует отнести наглядность и последовательность получения результатов.

Здесь хотелось бы подчеркнуть, что учебников, в которых освещается круг рассмотренных задач, нет. Методические указания к проведению лабораторных работ и наборы вариантов заданий приходится полностью и самостоятельно разрабатывать преподавателю, ведущему эту дисциплину.

В заключение хочу отметить, что решение одних и тех же задач путем составления программ и с помощью пакета Excel осуществляется студентами с разным интересом и желанием. Программирование обычно вызывает больше трудностей, а работа с Excel происходит с большим желанием. А, как известно, положительные эмоции стимулируют деятельность. Относительно программирования в последние годы отмечается определенное сопротивление его изучению, и не только со стороны студентов, но и выпускающих кафедр.

## **ВАРІАТИВНІСТЬ ЗМІСТУ, ФОРМИ І МЕТОДИ ПОДАННЯ НАВЧАЛЬНОГО МАТЕРІАЛУ З ВИКОРИСТАННЯМ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

Т.І. Чепрасова  
м. Луцьк, Волинський державний університет  
імені Лесі Українки

Розвиток інформаційних технологій здійснює суттєвий вплив на суспільство в цілому, і саме цю нову технологічну реальність повинна в першу чергу враховувати система вищої освіти. Тому навчальний процес у вищому навчальному закладі повинен бути організований і забезпечений так, щоб студент розв'язував більшу частину навчальних і організаційних завдань з використанням всього комплексу засобів сучасних інформаційних технологій. Це дозволить забезпечити новий рівень освітнього процесу та інтегрувати діяльність студентів, викладачів, організаторів навчального процесу.

Пропонується наступна структура електронних навчально-методичних матеріалів для забезпечення ефективного вивчення фундаментальних дисциплін у вищому навчальному закладі:

- навчальна програма курсу, плани усіх форм занять, графік поточного і підсумкового контролю;
- конспекти лекцій та ілюстративно-презентаційні матеріали для супроводу лекційних курсів, завдання для практичних і лабораторних занять, завдання для самостійної роботи;
- методичні рекомендації щодо підготовки курсових і дипломних робіт за курсом;
- скановані тексти зі спеціальної літератури та матеріали Інтернет-ресурсів, використані викладачем при підготовці курсу;
- гіпертекстові електронні підручники базового рівня з використанням засобів тестового контролю;
- програмні засоби навчального призначення, інтегровані зі спеціалізованими та прикладними програмами або спеціалізовані діяльнісні середовища для комп'ютерного моделювання задач предметної галузі та дослідження їх

на основі комп'ютерних експериментів.

Розміщення і збереження цих матеріалів проводиться на сервері факультету на відповідних Web-сторінках викладачів, які забезпечують підготовку і читання курсів за навчальним планом.

Реалізовані компоненти цієї структури визначають існування та функціонування освітнього інформаційного середовища, в якому передбачено єдиний доступ для всіх його учасників – викладачів, організаторів, студентів. Створене предметно-орієнтоване середовище характеризується гнучкістю, варіативністю змісту і форми подачі матеріалу, забезпечує підтримку ініціативи викладача і студента, виступає своєрідною картотекою знань, поданих з використанням різних методичних і технологічних прийомів.

Матеріально-технічне забезпечення освітнього інформаційного середовища включає парк обчислювальної техніки та інформаційно-обчислювальну мережу вищого навчального закладу, наявність і досить високі характеристики її виходу в Інтернет, комп'ютерний читальний зал бібліотеки.

Методичний аспект організації підготовки матеріалів проводиться викладачем-розробником курсу, а до програмної організації залучаються студенти спеціальності "Інформатика".

Розробка електронних навчально-методичних матеріалів здійснюється студентами в межах їх підготовки як фахівців з інформатики при виконанні індивідуальних завдань, частини курсових і дипломних робіт кафедри прикладної математики (залежно від обсягу поставленого технічного завдання), що цілком відповідає вимогам і змінам суспільного, наукового, технічного середовищ щодо адаптації до них вищої освіти. У процесі розробки та впровадження електронних навчальних засобів студенти навчаються раціональних прийомів застосування сучасної обчислювальної техніки для вирішення конкретних практичних проблем і для самостійного здобування та впорядкування знань, стають активними учасниками освітнього інформаційного середовища.

Вирішення проблеми створення електронного навчально-методичного забезпечення для фундаментальних дисциплін з новими формами та методами його подання можливе за рахунок того, що варіативна частина освітньо-професійної програми спе-

ціальності “Інформатика” напряму підготовки “Прикладна математика” містить спецкурси, що враховують нові досягнення у відповідних галузях. З достатньо розвинутою інфраструктурою Інтернету та реальними потребами розширення освітньої підготовки людей виникає необхідність у вивченні можливостей, досвіду та запровадження дистанційного навчання, що, як основу, використовує електронне подання навчально-методичних матеріалів.

Базова підготовка бакалаврів прикладної математики на четвертому році навчання дозволяє включити в систему їхніх знань і умінь питання, що стосуються розробки, реалізації та впровадження дистанційного навчання. Тому протягом семестру для бакалаврів читається спецкурс “Технологія дистанційної освіти” загальним обсягом 108 годин, з них – 14 годин лекційних, 16 годин лабораторних, 24 години семінарських, 54 години самостійних занять.

Організація навчального процесу здійснюється таким чином: на лекції викладач знайомить студентів з інформацією, що стосується різних підходів до тлумачення дистанційного навчання, його видів, способів здійснення та реалізації, необхідного організаційного, матеріально-технічного, програмного, психолого-педагогічного забезпечення, переліку основних центрів здійснення дистанційної освітньої діяльності у нас в країні і за кордоном.

Лабораторні заняття призначені для розгляду технології пошуку необхідної інформації в Інтернет, визначення критеріїв її добору, класифікації і структурування за визначеними показниками щодо особливостей дистанційного навчання, способів представлення і збереження знайдених матеріалів для подальшого використання. Заняття проводяться в комп’ютерній залі, в якій з кожного робочого місця є доступ в Інтернет за допомогою інформаційно-обчислювальної мережі університету з досить високими характеристиками.

Самостійна робота організовується так, що кожен студент у вільний від занять час у комп’ютерній залі чи в іншому місці здійснює пошук в мережі Інтернет інформації, необхідної для розкриття поставленого перед ним індивідуального завдання, що стосується безпосередньо технології дистанційної освіти. Поста-



новка цих залікових завдань передбачає не тільки збір статичної інформації, але і її представлення у вигляді міні-електронного курсу з використанням гіпертекстових та мультимедійних засобів, з ілюстрацією фрагментів розроблених дистанційних курсів. Необхідною складовою цього представлення є анований файл використаних Інтернет-ресурсів. Така діяльність передбачає застосування студентами отриманих знань і вмінь з курсів фундаментальної та фахової підготовки, а також їх творчий розвиток в нових умовах.

Семинарські заняття зі спецкурсу будуються на основі презентації та захисту зібраних студентами матеріалів, їх обговорення, зазначення позитивних якостей та недоліків розглянутих ними систем. Електронний варіант виконаного залікового завдання розміщується на Web-сайті факультету і є основою для зарахування студентіві спецкурсу "Технологія дистанційної освіти". Для кафедри залишається добірка матеріалів, які викладачі використовують при удосконаленні змісту, форм, методів навчання фундаментальних дисциплін.

Організація діяльності, результатом якої є створені і готові до використання електронні навчально-методичні матеріали для забезпечення навчального процесу з фундаментальних дисциплін, визначає викладача і студента як рівноправних партнерів у здійсненні процесу надходження і трансформації знань. Молодий фахівець за рахунок комплексного застосування досить повного комплексу засобів інформаційних технологій отримує високий рівень фахової підготовки, здатний самостійно опанувати нові знання, готовий до стрімких змін в умовах практичної діяльності, володіє прийомами "віртуального" погляду на світ.

## **ПЛАН КОМПЬЮТЕРНОГО УЧЕБНОГО ПОСОБИЯ ДЛЯ КУРСА «ДИСКРЕТНЫЕ УСТРОЙСТВА АВТОМАТИКИ»**

Д.А. Шелист

г. Кривой Рог, Криворожский металлургический факультет На-  
циональной металлургической Академии Украины

В настоящее время очевидно, что персональный компьютер способен быть универсальным техническим средством обучения, при помощи которого представляется возможным не только осуществить подачу фактического учебного материала, но и обеспечить соответствующую подготовку, отвечающую индивидуальному уровню подготовленности с учётом уровня знаний каждого обучаемого, расстановку, темп учёбы и многое другое. Попытки реализовать построение компьютерного учебного комплекса по курсу дискретных устройств автоматики в основном связаны со сложностями в программе курса микроэлектроники и микропроцессорной техники, то есть, не усвоив материал по курсу микроэлектроники и микропроцессорной техники, изучить курс дискретных устройств на компьютерной модели будет достаточно сложно. Поэтому для того, чтобы учащийся смог пройти курс успешно, в разрабатываемой программе обучения предлагаются специально созданные обучающие подсказки, о которых речь пойдет ниже.

Освоение курса дискретных устройств автоматики может быть полезно в равной степени студентам очно-заочной (вечерней), заочной и дневной формы обучения как новый, дополнительный вид самостоятельных занятий.

Данная работа представляет пособие по курсу дискретных устройств автоматики, направленное на освоение теоретического материала по разделам, рассматриваемым в курсе, а также на обучение студентов решению стандартных задач, возникающих в процессе обучения.

Компьютерное пособие разбивается на несколько частей, которые тесно связаны между собой (рис. 1).



Рис. 1.

ПРОГРАММА КУРСА определяет характер связи отдельных частей курса, их суть, а также порядок обращения к ним.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ выступает как информационная база системы. Она разбита на разделы; главы, темы, параграфы. Кроме того, параграфы имеют разбиение, которое соответствует смысловым порциям, относящимся к отдельным понятиям, величинам, их характеристикам, единицам.

Компьютерная программа для обучения строится так, чтобы сложить наиболее полное представление об интересующей обучающегося проблеме. Этому способствует применение подсказывающего курсора, открывающего ссылки на те или иные разделы в теории курса или собранные в решебнике, вопроснике или глоссарии, и высвечивая на экране порции текста. Там, где считается возможным и целесообразным, текст может дополняться фрагментами компьютерной анимации.

ВОПРОСНИК – это достаточно большое количество вопро-

сов различной сложности, сформулированных для ответов в аналитической и смешанной формах.

Его можно использовать в двух режимах: тренажёра и контролёра. В режиме тренажёра указанные ответы содержат ссылки на смысловые порции, выделенные в теоретической части и в глоссарии, относящиеся к содержанию правильного и неправильного ответа.

В режиме контроля из вопросов формируют небольшие тестовые наборы, охватывающие содержание темы. По результатам тестирования программой выставляется оценка, соответствующая знанию теории.

ЭНЦИКЛОПЕДИЯ в данном случае должен быть не только толковым словарем, но и небольшой энциклопедией курса дискретных устройств, в котором указываются всевозможные типы, характеристики и параметры используемых в курсе обучения элементов. Также она должна быть связана со смысловыми порциями теоретической части и дополнять краткие технические сведения формулами.

Оформление словаря возможно в виде таблицы, удобной для пользователя при решении задач или рассмотрении вопроса.

РЕШЕБНИК предназначен для обучения пользователя правильному подходу к решению задач по теме. На примере нескольких стандартных задач путём создания системы вопросов и ответов проводится первичный анализ.

При этом делается попытка расчленить задачу на элементарные условия и требования. В совокупности условий нужно видеть объект и его характеристики; если объектов несколько, то выяснить их отношение. Анализ объектов и их отношений позволяет мысленно сформировать модель объекта и схематически описать её.

Передача функциональных шагов компьютеру сопряжена с особыми трудностями и потерями. Однако здесь можно делать ставку на активизацию творческих способностей пользователя, его умение целенаправленно пользоваться энциклопедией, теоретической частью и системой помощи. Необходимо, чтобы за счёт точно и корректно сформулированных вопросов обучаемый не только получил правильный ответ на поставленный вопрос, но и смог проанализировать полученный результат.

ЗАДАЧНИК содержит задачи для самостоятельного решения и контрольные задания. И те, и другие обеспечены системой пошаговой помощи по просьбе пользователя. Однако при выполнении: контрольной работы каждое обращение за помощью сопровождается начислением штрафных баллов.

Учитывая широкий спектр способностей обучаемых, предусмотрены не только пошаговое выполнение действий задачи, но и возможность её быстрого решения. Тогда пользователь может сразу перейти к последней операции с составлением структурных схем и моделированию процесса на симуляторе.

При этом программа сама выставляет оценку о проделанной работе – таким образом исключаются нюансы усталости преподавателя или списывания студента у студента, так как задания разные и на выполнение их дается строго определенное аудиторное время (для дневной формы обучения).

## ИСПОЛЬЗОВАНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ПРИ ИЗУЧЕНИИ ФУНДАМЕНТАЛЬНЫХ ДИСЦИПЛИН

А.В. Шматко, Е.А. Панина

г. Харьков, Академия пожарной безопасности Украины  
fd.apbu@list.ru

В части использования тестов в учебном процессе одним из лидеров являются США, в которых Служба Образовательного тестирования через сеть представительств и филиалов организует по стране участие миллионов людей в процессе тестирования. При этом американское министерство образования никоим образом не обязывает выпускников школ заполнять тесты, а университеты – засчитывать их результаты. В развитых странах образование – это свободный рынок, и такие тесты, как SAT (Scholastic Aptitude Test), которые фиксируют уровень готовности к обучению в высшей школе, оказываются удобным инструментом для взаимного выбора – ВУЗ подбирает студентов, а студенты – ВУЗ. Примерно по аналогичным принципам действуют и другие системы образовательного тестирования, более знакомые нам – TOEFL, GMAT, GRE [1, 2].

Системы профессионального тестирования (тесты Microsoft, Oracle, Novel и др.) получили особенное распространение среди технических специалистов. Широко распространены системы сертификации в области бухгалтерского учета и финансов – CPA (Certified Public Accountant), ACCA (The Association of Chartered Certified Accountants) и ряд других

Существует также целый ряд универсальных систем тестирования, которые помогают получить подтверждение своего уровня знаний в определенной области и учитываются работодателями – например, система Brainbench (<http://www.brainbench.com>), которая в полной мере реализует принципы дистанционного прохождения теста, и позволяет любому зарегистрировавшемуся пользователю (а их сейчас более двух миллионов) пройти тесты по более чем 200 различным специализациям.

В целом, системы тестирования на Западе построены весьма

технологично и используют весь спектр современных методов – от использования специального программного обеспечения до дистанционного тестирования через сеть Интернет. Такие методы позволяют снизить практически до нуля вероятность ошибок при обработке результатов теста, исключить «человеческий фактор» из процедуры проверки и сэкономить время.

В настоящее время в Украине и в странах ближнего зарубежья Интернет получил активное развитие. В декабре 1999 года в России появился интересный Интернет-проект, посвященный тестированию – Examen.ru (<http://www.examen.ru>), разработанный компанией Websoft (<http://www.websoft.ru>). Здесь можно найти более сотни самых разных экзаменов и тестов, которые можно сдать в режиме он-лайн, при этом все они являются качественной имитацией тех тестов и экзаменов, которые приходится сдавать в реальной жизни. Однако разработчики не ограничились только собственно тестами – на сайте можно найти огромное количество справочных и консультационных материалов о том, как нужно сдавать тесты. Пока на этом сайте нет собственных авторских разработок, но не это было целью создателей – они создали прообраз того механизма, те принципы, которые могут быть использованы более широко. В 2000 году сайт получил национальную Интел Интернет премию и был признан лучшим в номинации «Образование и наука» [3, 4].

В настоящее время в Академии пожарной безопасности Украины (АПБУ) на кафедре фундаментальных дисциплин применяется автоматизированная система проверки знаний (АСПЗ TEST-2000). Этот технологический подход к процессу обучения позволяет практически по каждой теме проверить индивидуальные знания каждого курсанта или студента с минимальными затратами учебного времени. Постоянный контроль дисциплинирует обучающихся и значительно увеличивает процент усвоения материала.

TEST-2000 предназначен для автоматизации управления процессом систематически проводимой проверки знаний студентов и курсантов по курсам и темам различных дисциплин, изучаемых в учебном заведении.

Применение тестирования позволяет оценить собственно знания, а не способность учащихся излагать свои мысли, поэто-

му оно не может полностью заменить другие виды контроля. Тем не менее, использование тестов в учебном процессе готовит выпускников АПБУ к системе профессионального тестирования, которое получает все большее развитие на рынке труда Украины.

#### Литература.

1. Беляцкий Н.П., Велесько С.Е., Ройш П. Менеджмент. В 8 кн. Кн. 7. Менеджмент персонала 2000: Учеб. пособие.– Мн.: БГЭУ, 1998.
2. Гутгарц Р., Чебышева Б. Автоматизированная система проверки знаний // Управление персоналом. – М: – 2000. – №5.
3. Анюшин В.В., Анкипович Е.С., Горенкова Г.А., Мухина Е.С., Пропой Н.А., Чаиркина Л.Н. Организация учебной деятельности студентов в рамках модульно-рейтинговой системы обучения. // Новые технологии в науке и образовании. – Новосибирск, 1998. – С. 90-92.
4. Муравьев А.А. Особенности технологического подхода в обучении. // Новые технологии в науке и образовании. – Новосибирск, 1998. – С. 5-7



## ИЗУЧЕНИЕ МАТЕМАТИЧЕСКОГО ПРОГРАММИРОВАНИЯ: КОМПЬЮТЕРНЫЙ ПОДХОД

А.И. Шурдук, Л.Н. Шенгерий

г. Полтава, Полтавский университет потребительской коопера-  
ции Украины  
mach@pci.poltava.ua

В составе современной модели высшей школы Украины можно выделить несколько основных тенденций. Во-первых, это сокращение общего количества аудиторных учебных часов. В частности, это касается математических дисциплин. Для обеспечения высокого уровня подготовки специалистов в таких условиях актуализируется роль самостоятельной работы студентов. Во-вторых, это безусловно позитивная тенденция внедрения компьютерных технологий в учебный процесс. В программе дисциплины «Математическое программирование», разработанной Киевским институтом экономики и менеджмента «ЭКОМЕН», которая была утверждена 19 октября 2001 года, явным образом указывается на необходимость использования компьютеров в процессе преподавания этой дисциплины. Внедрение компьютерных технологий значительно расширяет возможности усовершенствования процесса обучения студентов дисциплине «Математическое программирование» путем создания учебно-контролирующего программного обеспечения. В таком контексте возникает важная проблема обеспеченности соответствующими программными материалами на уровне каждого преподавателя, учебной дисциплины, кафедры и т.д.

На кафедре высшей математики и физики Полтавского университета потребительской кооперации Украины (ПУПКУ) разработана и апробирована учебно-контролирующая программа изучения темы “Транспортная задача” (см. рис. 1).

Предполагается построение опорного плана задачи методами северо-западного угла и минимального элемента. Оптимальный план задачи находим с помощью метода потенциалов. Среди существенных характеристик разработанной программы можно выделить такие:

## Транспортная задача.

Имеется четыре пункта поставки однородного груза  $A_1, A_2, A_3, A_4$  и пять пунктов  $B_1, B_2, B_3, B_4, B_5$  потребления этого груза. На пунктах  $A_1, A_2, A_3, A_4$  находится груз соответственно в количестве  $a_1, a_2, a_3, a_4$ . В пункты  $B_1, B_2, B_3, B_4, B_5$  требуется доставить соответственно  $b_1, b_2, b_3, b_4, b_5$  ед. груза. Заполните матрицу-таблицу исходными данными (мощности поставщиков и потребителей, тарифы перевозок) по своему варианту и найдите такой план закрепления потребителей за поставщиками однородного груза, чтобы общие затраты по перевозкам были минимальными.

		Мощности потребителей				
		$b_1$	$b_2$	$b_3$	$b_4$	$b_5$
Мощности поставщиков		10	50	50	50	50
$a_1$	90	9	8	3	4	7
$a_2$	60	7	5	5	11	8
$a_3$	30	4	2	6	9	5
$a_4$	40	7	4	3	8	4

Модель закрытая

Рис. 1. Постановка задачи

- простота и доступность в работе. Для успешного овладения практическими навыками решения транспортной задачи с помощью этой программы не требуются знания Excel. В программе не предусматривается использование Excel-функций;
- каждый студент имеет возможность решать свой вариант задачи. Одинаковыми являются количество “производителей” и “потребителей”;
- процесс решения транспортной задачи делится на  $n$  шагов, на каждом из которых предусматривается возможность самоконтроля. Диагностируется правильность вычисления значений потенциалов и найденной общей стоимости перевозок, оптимальность или не оптимальность полученного плана задачи (см. рис. 2);
- значительно упрощается механизм осуществления контроля выполнения задачи преподавателем и существенно уменьшается время на его реализацию;
- оптимально учитываются индивидуальные темпоральные показатели каждого студента, которые зависят от его способностей и уровня обученности;

**Матрица перевозок (изменяемые ячейки)**

	10	50	50	50	50		
90	10	50	30			верно	0
50			20	30		верно	-2
30				20	10	верно	0
40					40	верно	1
Потенциалы потребности	9	8	3	9	5		
	верно	верно	верно	верно	верно		

*Решение не оптимально !*

Суммарная стоимость перевозок: 1400

Рис. 2. Опорный план задачи

– реализована возможность параллельного усвоения теоретического материала данной темы. Если у студента возникает потребность повторения теоретического материала, достаточно навести указатель манипулятора “мышь” на соответствующую метку и воспользоваться подсказкой.

– студенты самостоятельно переходят в случае необходимости к закрытой модели транспортной задачи;

– студенты имеют возможность самостоятельно убедиться в существовании альтернативных оптимальных планов перевозок некоторой транспортной задачи.

Разработка и внедрение данной программы в учебный процесс ПУПКУ позволило:

- 1) помочь студентам в самостоятельной работе при изучении темы «Транспортная задача»;
- 2) осуществить эффективный самоконтроль знаний и умений студентами по этой теме;
- 3) повысить качество контроля преподавателями самостоятельной работы студентов.

## ВИВЧЕННЯ БАЗ ДАНИХ НА ОСНОВІ МАТЕРІАЛІВ ВІТЧИЗНЯНОЇ НАУКИ ТА ТЕХНІКИ

Т.Є. Щепакіна

м. Бердянськ, Бердянський державний педагогічний університет

До пакету прикладних програм MS Office входить система управління базами даних MS Access, яка вивчається у загально-освітніх навчальних закладах на ряду з вивченням інших прикладних програм пакету MS Office. Але на вивчення теми “Бази даних. Системи управління базами даних” у програмі курсу “Основи інформатики та обчислювальної техніки” [1] приділяється більша кількість годин, а саме 16, ніж на вивчення інших тем. Це обумовлено практично-прикладною багатofункціональністю, складністю даної теми.

Від вчителя інформатики залежить, як він підготує учнів до сприйняття даної теми. Необхідно зауважити, що вивчення даної теми у високій мірі залежить від успішності вивчення попередньої теми “Електронні таблиці”, без цього неможливо засвоїти принципи побудови та основні функції електронних баз даних, роз’яснити важливість впорядкування та пошуку, зберігання та обробки інформації у базах даних. Отже, на основі систематичного вивчення електронних таблиць і СУБД здійснюються внутрішньопредметні та міжпредметні зв’язки між темами курсу “Основи інформатики та обчислювальної техніки”. Також реалізується принцип переходу від простого до складного: цілком зрозуміло, що СУБД мають більші можливості по обробці великих масивів інформації.

Необхідно підкреслити, що робота з сучасними базами даних вимагає у достатньо високій степені знання певних тем з курсу програмування (масиви, операції над масивами, посилання на різні типи даних, сортування масивів тощо). Також необхідні уяви про такі математичні поняття як множини, операції над множинами з курсу алгебри. Таким чином можна зробити висновок про те, що вивчення теми “Бази даних. Системи управління базами даних” є своєрідним підсумковим етапом навчання інформатики, який поєднує знання та навички, отримані при вивченні інших тем та предметів. Звідси випливає, що при вивченні теми

“Бази даних” вчитель повинен скоординувати свою роботу з вчителями математики.

Не можна не підкреслити важливість вивчення даної теми у курсі інформатики у загальноосвітніх навчальних закладах у аспекті формування інформаційної культури сучасного учня. Як відомо, характерною тенденцією останніх років є активне впровадження нових інформаційних технологій у навчальний процес. У системі освіти комп’ютер використовується і як потужний обчислювальний засіб, і як спосіб збереження величезних масивів інформації з різних галузей знань [3, с. 64-69]. Задача вчителя підготувати учня до набуття належного рівня інформаційної культури, вміння реалізувати свої знання у подальшому навчанні у вищих навчальних закладах, вміти автоматизувати процеси по обробці даних у системі шкіл та суспільстві. Вважаємо, що не має потреби більш широко обґрунтовувати необхідність вивчення теми “Бази даних. Системи управління базами даних”, це уявляється у достатній степені очевидним.

Успіх засвоєння та вивчення тем, навіть найскладніших, з курсу “Основи інформатики та обчислювальної техніки” залежить від рівня доступності та зрозумілості учнями навчального матеріалу. Кваліфікований вчитель завжди прагне до яскравості та емоційності навчального процесу, особливо якщо мова йде про використання комп’ютерно-орієнтованих технологій навчання. Одним із шляхів стимулювання мимовільної уваги, внутрішньої афективної мотивації є використання фактичного матеріалу із галузі досягнень вітчизняної науки і техніки [2, с. 6-9].

Не можна забувати і про виховні впливи в ході навчання, а вивчення розглянутої теми на основі характеристик на основі досягнень вітчизняної науки і техніки може надавати всі передумови виховання в учнів національної свідомості.

Отже, одним із шляхів, яким можливо досягти яскравості навчального процесу, може бути вивчення теми “Бази даних. СУБД” з використанням науково-технічного матеріалу, наприклад, досягнення всесвітньо відомого авіаційного конструкторського бюро ім. О.А. Антонова. Але у даній методиці створення однієї таблиці не достатньо для повної реалізації можливостей СУБД. Тому, зміст новизни методики проведення таких занять полягає в тому, що учням пропонується створити базу даних на

основі двох зв'язаних таблиць, матеріали до яких були підібрані і опрацьовані учнями при вивченні попередніх тем: “Літаки КБ “Антонов”, “Персоналії у галузі найвідоміших розробок вітчизняної науки і техніки”. Ці матеріали розглядалися при вивченні теми “Електронні таблиці”, де створювалися лише окремі таблиці з даними, виконувалися розрахунки, пошук та впорядкування даних.

Назви полів першої таблиці впливатимуть із постановки задачі: Ключове унікальне поле, Назва літака, Модифікації, Рік створення, Екіпаж (чол.), Швидкість максимальна (км/год), Практична стеія (м), Дальність дії (км), Довжина розбігу (м), Довжина пробігу (м), Максимальна злітна маса(кг), Розмах крил (м), Кількість двигунів, Тип двигуна, Потужність двигуна (к.с.) (рис. 1).

**Літаки конструкторського бюро "Антонов"**

Код	3	Довжина пробігу	33000
Назва	Ан-225 "Мрія"	Максимальна злітна маса	600000
Модифікації	-	Вантажопідйомність	250000
Рік створення	1998	Розмах крил	88
Екіпаж	6	Кількість двигунів	6
Швидкість максимальна	850	Потужність двигуна	23400
Швидкість пробігу	750		
Практична стеія	11000		
Дальність дії	96000		
Довжина розбігу	33000		

Запис: 1 з 3 | Режим форми

Рис. 1. Приклад створення таблиці “Літаки конструкторського бюро “Антонов”

Назви полів таблиці “Персоналії”: Ключове унікальне поле, ПІБ, Дата народження, Місце народження, Освіта, Місце роботи, Наукова ступінь, Розробки, Друковані матеріали, Відзнаки, Публікації.

Від учнів вимагається:

- створити структуру бази даних, ввести необхідні дані,
- впорядкувати дані та шукати дані у БД за відповідними критеріями, запропонованими вчителем
- створити індексні поля, створити зв’язки між елементами в таблицях
- створити для таблиць форми та звіти
- навчитися створювати прості запити.

За допомогою створення зв’язку між таблицями “Літаки” та “Персоналії” встановлюється відношення кожного запису однієї таблиці до другої. Наприклад, встановлюється зв’язок між полями *Ім’я розробника* таблиці “Літаки” до *ПІБ* у таблиці “Персоналії”.

Науково-технічне наповнення баз даних допоможе не тільки ефективно працювати з СУБД, але й підвищить рівень зацікавленості до навчання, обумовить систематичність вивчення тем курсу “Основи інформатики та обчислювальної техніки”, сформує всі сторони мотиваційної сфери учення школярів.

#### Література

1. Жалдак М.І., Морзе Н.В., Науменко Г.Г. Програма для середніх закладів освіти “Основи інформатики та обчислювальної техніки”. Рекомендовано Міністерством освіти України. – К.: Перун, 1996. – 24 с.
2. Лапінський В.В. Використання фактографії вітчизняної науки і техніки на уроках інформатики // Комп’ютер у школі та сім’ї. – 2001. – №6 – С. 6-9.
3. Щепакіна Т.Є. Активізація навчально-пізнавальної діяльності учнів при навчанні інформатики як складова інформатизації сільської школи / Збірник наукових праць Бердянського державного педагогічного університету (Педагогічні науки). – №6. – Бердянськ: БДПУ, 2002. – 220 с.

## Зміст

<i>О.В. Бабич.</i> Створення інтерактивних навчальних посібників за допомогою WebEXE.....	3
<i>А.Н. Бакал.</i> К вопросу о концепциях современной информатики.....	6
<i>Д.Е. Бобылев.</i> Проблема введения в содержание курса “Численные методы решения задач на ЭВМ” (для технических вузов) материала о методе граничных элементов.....	11
<i>М.А. Богданевич.</i> Організація навчання оперативного персоналу на основі методики алгоритмів дій.....	14
<i>М.А. Бондаренко.</i> Проблеми вивчення дисципліни “Інформатика та обчислювальна техніка”.....	19
<i>О.М. Боско.</i> Особливості вивчення прикладного програмного забезпечення у вищому навчальному закладі.....	23
<i>В.К. Буряк, С.М. Ракова.</i> Програмування в X Window мовою Паскаль.....	26
<i>І.П. Вакалюк, В.М. Якимчук, І.Ю. Ванджура, О.В. Ткач-Мотуляк, Н.І. Скробач.</i> Місце новітніх технологій в системі постійного інтегрованого тренінгу майбутніх спеціалістів.....	29
<i>Л.В. Васильєва.</i> Електронні підручники у вивченні комп’ютерних наук.....	31
<i>С.В. Вельма.</i> Структурна організація системи моделей репрезентації процедурних знань дисципліни “Інформаційні технології в галузі”.....	33
<i>В.М. Власенко, В.І. Сфіменко, Р.В. Ромадін.</i> Організація системи управління навчальним процесом на основі результатів поточного контролю з використанням комп’ютерних технологій.....	39
<i>В.В. Войтенко, А.В. Морозов.</i> Система on-line тестів для контролю знань з програмування та інформатики.....	44
<i>Т.В. Волкова.</i> Розв’язування прикладних задач в електронних таблицях.....	45
<i>А.Ю. Гайда.</i> Перспективы применения операционной системы Linux в учебном процессе.....	56
<i>Е.М. Галбай, Н.Н. Дудникова.</i> Комплексная автоматизация проверки знаний студента для специальности «Экономическая кибернетика».....	62
<i>М.С. Головань.</i> Використання методу проектів у процесі	



вивчення інформатики та комп'ютерної техніки в економічному вузі .....	67
<i>М.С. Головань, М.О. Антонченко.</i> Формування якісних знань учнів з використанням експертних систем .....	72
<i>Л.П. Голубєв, Ю.М. Пилипенко, Ю.П. Юрачківський.</i> Методика навчання основам інформатики і програмування у VBA .....	80
<i>К.Г. Гриценко.</i> Методика навчання основам інформатики і програмування у VBA .....	98
<i>Л.В. Гришко.</i> Порівняльний підхід до навчання основам програмування студентів технічних спеціальностей ВНЗ .....	101
<i>О.В. Гришпенюк, П.М. Григорук.</i> Досвід використання відкритого програмного забезпечення в курсі інформатики .....	107
<i>А.І. Дзундза, Н.М. Лосєва.</i> Використання ЕОМ як могутній засіб підвищення ефективності навчання у ВНЗ .....	110
<i>В.А. Долока, Л.А. Дубинец, Н.Л. Кириленко.</i> Проектирование электронных компонентов ЭВМ при помощи IP CORE генератора .....	115
<i>В.С. Єрємєєв, С.М. Прийма.</i> Активізація навчально-пізнавальної діяльності при вивченні програмування з використанням модельно-символічної технології .....	117
<i>В.И. Засельский, С.В. Швед.</i> Использование системы имитационного моделирования в учебном процессе кафедры фундаментальных дисциплин .....	131
<i>З.Г. Зуйкова.</i> Учебный вычислительный эксперимент, направленный на углубление знаний по термодинамике .....	134
<i>О.В. Ключко.</i> Формування конкретно-предметних знань з інформатики як першооснови розвитку професійної культури студентів-аграріїв .....	136
<i>Г.К. Кожевников, В.В. Сосновский, Л.А. Панова.</i> Применение технических средств обучения при изложении дисциплин компьютерного профиля .....	147
<i>В.В. Коломенська.</i> Розміщення задач на моделювання в шкільному курсі інформатики .....	151
<i>М.Г. Коляда.</i> Формування інформаційної культури майбутніх економістів на основі предметного моделювання .....	153
<i>Ю.М. Красюк.</i> Використання задач дослідницького характеру в процесі навчання інформатики у вищих закладах	

освіти економічного профілю .....	166
<i>И.А. Кривель, А.Н. Моргун.</i> Из опыта обучения студентов вузов программированию на языке Паскаль .....	174
<i>С.В. Куклина, Н.В. Моисеенко.</i> Концепция курса "Основы защиты информации" .....	178
<i>Н.А. Леонова.</i> Использование пакета символьной математики Maxima в курсе "Компьютерные технологии в научных исследованиях" .....	183
<i>Р.В. Лихачов.</i> Декілька питань про методи викладання ознайомчих курсів інформатики у ВНЗ .....	188
<i>С.С. Лихман.</i> Возможности написания обучающих программ под Windows .....	195
<i>Л.Б. Лищинська.</i> Деякі аспекти впровадження інформаційних технологій при викладанні комп'ютерних дисциплін у вищій школі .....	200
<i>В.Г. Логвіненко.</i> Інтернет як засіб розвитку пізнавальної самостійності студентів .....	204
<i>І.М. Лукаш.</i> Місце об'єктно-орієнтованого аналізу у процесі навчання об'єктно-орієнтованого програмування .....	208
<i>Г.Ю. Маклаков.</i> Особенности использования мультимедийных технологий в преподавании фундаментальных дисциплин в высшей школе .....	211
<i>Г.Ю. Маклаков.</i> Современные компьютерные технологии и информационная безопасность личности .....	215
<i>Г.Г. Маклакова.</i> Разработка мультимедийных приложений средствами Web-технологий .....	219
<i>В.Л. Малорян.</i> Проблемы обучения современному программированию .....	222
<i>І.В. Настенко.</i> Особливості викладання курсу комп'ютерної графіки .....	232
<i>О.Г. Онишко.</i> Формування деяких професійних умінь фахівця з вищою технічною освітою на заняттях з інформатики ..	236
<i>О.Г. Онишко.</i> Розвиток мислення студентів під час навчання інформатиці у вищому технічному навчальному закладі як компонент їхньої готовності до майбутньої професійної діяльності .....	242
<i>В.В. Осадчий.</i> Місце відкритого програмного забезпечення в обробці графічної інформації .....	247

<i>М.Е. Отставнов.</i> Свободное программное обеспечение школьной информатики .....	251
<i>Е.А. Панина, Л.В. Гусева.</i> Автоматизированная система проверки знаний Test-2000.....	261
<i>Н.В. Перегинец, И.А. Теплицкий.</i> Развитие алгоритмического стиля мышления младших школьников средствами языка Лого .....	264
<i>Л.І. Побережна.</i> Використання комп'ютерно-орієнтованих методик при викладанні фундаментальних дисциплін .....	271
<i>А.П. Полищук, С.А. Семериков.</i> О составлении и реализации учебных планов по кибернетическим дисциплинам в высшей школе .....	273
<i>А.Л. Полянничко.</i> Программа автоматизированного тестирования и контроля учебной деятельности .....	280
<i>Н.І. Праворська.</i> Активізація пізнавальної діяльності студентів під час вивчення інформатики засобами підручника .....	284
<i>Ю.Ф. Рева, М.А. Кислова.</i> Экспертно-обучающие системы рейтинга знаний, умений, навыков на основе компьютерных технологий обучения .....	294
<i>С.І. Романенко.</i> Методичні основи навчання інформаційного пошуку та проблема індивідуалізації знань студентів.....	302
<i>В.Г. Рябцев.</i> Метод оценки эффективности тестов диагностирования микросхем памяти .....	307
<i>Л.О. Савчук.</i> Психолого-педагогічні аспекти взаємозв'язку пізнавальної діяльності та використання комп'ютерної техніки при вивченні дисциплін інформаційного циклу .....	312
<i>Н.І. Соколянська.</i> Організація процедур доступу до інформаційних ресурсів ТРІАС .....	316
<i>В.А. Стороженко, А.Ю. Вакула.</i> Организационные и методологические проблемы преподавания дисциплины «Информатика и компьютерная техника» в высшей школе .....	320
<i>А.М. Стрюк, О.В. Попова.</i> Розробка системи навчання та контролю знань з дисципліни “Теорія операційних систем” .....	324
<i>М.І. Стрюк.</i> Підготовка студентів спеціальності “Професійне навчання” до використання нових інформаційних технологій навчання .....	329
<i>Ю.А. Супрунова.</i> Применение обучающей программы .....	

эмуляции DOS при изучении курса «информатика, вычислительная техника и программирование».....	334
<i>І.Є. Фільо.</i> Тестування навчальної успішності, як форма організації попереднього контролю знань студентів на лабораторних заняттях з інформатики.....	337
<i>І.Є. Фільо.</i> Розв'язок системи лінійних алгебраїчних рівнянь в електронних таблицях Microsoft Excel.....	342
<i>А.А. Хараджян.</i> Использование виртуальных машин в курсе «Программное обеспечение ЭВМ».....	346
<i>М.О. Цензура.</i> Досвід розробки автоматизованої навчальної системи.....	349
<i>С.Н. Чаплыгина.</i> Изложение раздела Excel в курсе информатики для экологов.....	355
<i>Т.І. Черасова.</i> Варіативність змісту, форми і методи подання навчального матеріалу з використанням інформаційних технологій.....	358
<i>Д.А. Шелист.</i> План компьютерного учебного пособия для курса «Дискретные устройства автоматики».....	362
<i>А.В. Шматко, Е.А. Панина.</i> Использование информационных технологий при изучении фундаментальных дисциплин.....	366
<i>А.И. Шурдук, Л.Н. Шенгерий.</i> Изучение математического программирования: компьютерный подход.....	369
<i>Т.Є. Щепакіна.</i> Вивчення баз даних на основі матеріалів вітчизняної науки та техніки.....	372

Наукове видання

**Теорія та методика навчання  
математики, фізики, інформатики**

**Випуск 3**

**В 3-х томах**

**Том 3**

Підп. до друку 24.02.2003  
Бумага офсетна №1  
Ум. друк. арк. 20,08

Формат 80x84 1/16.  
Зам. №2-2403  
Наклад 500 прим.

Видавничий відділ Національної металургійної академії України  
50006, м. Кривий Ріг-6, вул. Революційна, 5

---

E-mail: cc@kpi.dp.ua