

Міністерство освіти та науки України
Національна металургійна академія України

Теорія та методика
навчання математики,
фізики, інформатики

Збірник наукових праць
Випуск 4

Том 3

Кривий Ріг
Видавничий відділ НМетАУ
2004

Теорія та методика навчання математики, фізики, інформатики: Збірник наукових праць. Випуск 4: В 3-х томах. – Кривий Ріг: Видавничий відділ НМетАУ, 2004. – Т. 3: Теорія та методика навчання інформатики. – 351 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 966-8506-094-3

ВИВЧЕННЯ ОСНОВ SQL В КУРСІ «БАЗИ ДАНИХ ТА ІНФОРМАЦІЙНІ СИСТЕМИ»

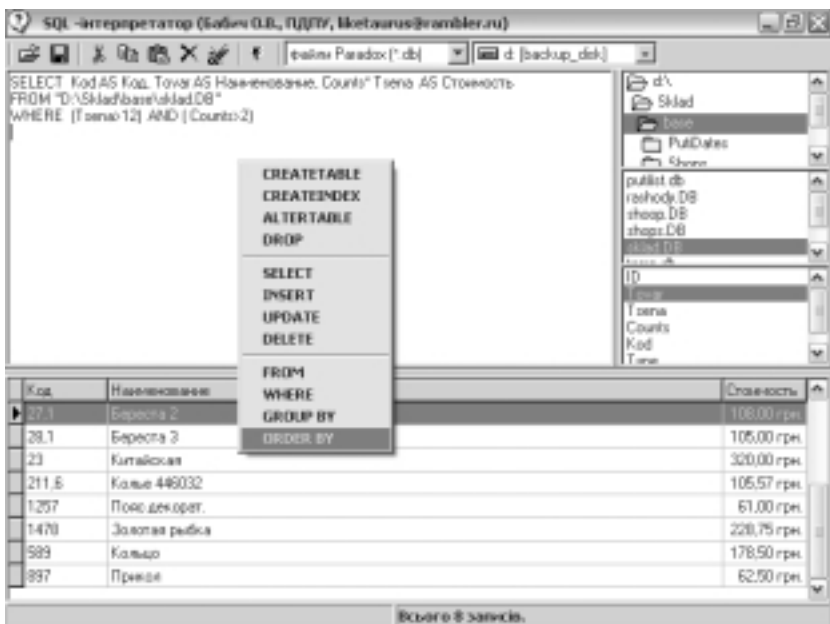
О.В. Бабич

м. Полтава, Полтавський державний педагогічний університет
ім. В.Г. Короленка
liketaurus@rambler.ru

Останнім часом до навчальних планів багатьох вузів нашої країни було включено дисципліну під назвою «Бази даних та інформаційні системи» [1]. Звичайно, особливу увагу цій дисципліні слід приділити під час підготовки спеціалістів зі спеціальностей «Програмування для ЕОМ та АС», «Інформаційні системи, банки та бази даних», «Комп'ютерна безпека» і т.ін., однак майбутнім вчителям інформатики цей предмет теж можливо знадобиться.

Традиційно до переліку тем, які вивчаються в межах цієї дисципліни, входить робота з Fox Base+ в діалоговому режимі, основи роботи з Fox Pro, ведення баз даних в MS Access та вивчення основ структурованої мови запитів (SQL). Якщо з попередніми темами не виникає жодних ускладнень, то з вивченням SQL є певні проблеми. Головна з них – яке середовище використовувати для роботи з запитамі? Звичайно, можна будувати запити в Fox Pro або Access, а потім переглядати SQL-інструкції, згенеровані конструктором. Можна працювати в Borland Database Desktop [2], однак робота з запитамі реалізована в ньому не найкращим чином. Найпростішим виходом з цієї ситуації є створення власного програмного продукту – по суті, редактора-інтерпретатора SQL-інструкцій з деякими додатковими функціями, тим більше, що використовуючи RAD Delphi, зробити це досить просто.

В рамках цього завдання автором було розроблено програму «SQL-інтерпретатор», яка допомагає вирішити проблему навчання SQL. Зовнішній вигляд вікна програми показано на малюнку.



Робота з програмою надзвичайно проста і не вимагає жодних спеціальних знань. Коротку інформацію про призначення кожного елементу вікна програми можна побачити в рядку статусу при наведенні покажчика миші.

Спочатку з випадального списку треба вибрати тип файлу бази даних (оскільки в курсі вивчається Fox Pro та Fox Base, які належать до сімейства dBase, а програма написана на Delphi, то стандартно підтримуються два формати: DB та DBF). Далі треба вибрати диск та каталог, в якому міститься файл БД, на основі якого буде побудовано запит. При виборі каталогу у відповідному списку виводиться його вміст. Одне натиснення лівої клавіші миші на будь-якому файлі вибраного типу виводить список полів бази даних. Тип та розмір вибраного поля виводяться в рядку статусу. Подвійне натиснення на файлі БД переносить його повне ім'я в подвійних лапках у робочу область програми. Аналогічним чином можна перенести назву вибраного поля. Для вставки ж основних SQL-інструкцій можна використати контекстне меню, яку відкривається по натисненню правої кнопки миші.

Створений запит можна запустити на виконання за допомо-

гою відповідної кнопки інструментальної панелі. При цьому результати його роботи відображаються у нижній частині вікна програми.

Вихідний SQL-текст можна писати і вручну без використання меню та елементів навігації, однак їх використання робить роботу з програмою ще простішою. Текст запити можна зберегти для подальшої роботи. Також, природно, присутні всі стандартні можливості редагування тексту.

Звичайно, програма вимагає подальшого вдосконалення в таких напрямках як, наприклад, підсвічування синтаксису, можливості вставки інших інструкцій та побудови виразів без використання клавіатури і т.ін. Також не слід забувати, що підтримка SQL в Borland Database Engine (BDE) реалізована не в повній мірі [3]. Однак, незважаючи на ці обставини, обираючи «SQL-інтерпретатор», викладач отримує просте абстрактне середовище для формування запитів, причому не прив'язане до конкретної системи керування базами даних, що дозволяє зосередитися саме на вивченні основ мови, а не роботи з інтерфейсом певного програмного пакету.

Література

1. Гайдамакин Н.А. Автоматизированные информационные системы, базы и банки данных. Вводный курс: Учебное пособие. – М.: Гелиос, 2002.
2. Borland Database Desktop Reference. Borland Software Corp., 2001.
3. Borland Database Developer's Guide. Borland Software Corp., 2001.

КОМП'ЮТЕРНЕ ТЕСТУВАННЯ З ІНФОРМАТИКИ: ПІДСУМКИ ПЕДАГОГІЧНОГО ЕКСПЕРИМЕНТУ В ОДЕСЬКІЙ НАЦІОНАЛЬНІЙ АКАДЕМІЇ ЗВ'ЯЗКУ

В.Ю. Баранов
м. Одеса, Одеська національна академія зв'язку
імені О.С. Попова
bwr@ukr.net

Термін «тестування» у наш час вживається, передусім, для іменування процесів масових «замірів» знань з використанням стандартизованих засобів («тестів») та процедур перевірки й оцінювання їх результатів. Існують ґрунтовні дослідження, присвячені методологічному, науковому та методичному забезпеченню відповідних процесів оцінювання знань (див., наприклад [3], [5]). Задача нашого дослідження полягала у вивченні можливостей систем комп'ютерного тестування знань (СКТЗ), пов'язаних із ставленням питань у сеансі тестування і, головне, коректним оцінюванням відповідей на них.

У комп'ютерному тестуванні здебільшого використовується методика так званого множинного вибору (multiple choice). Головною позитивною рисою цієї методики є простота комп'ютерної реалізації. Але навіть «досконалим» тестам типу multiple choice властиві серйозні психолого-педагогічні недоліки (див., наприклад, [3]). З іншого боку, практично всі сучасні системи, які допускають відкриті відповіді, по суті обмежуються традиційним методом «ключових слів». Інший підхід застосовано у розробленій в ОНАЗ під керівництвом проф. Г.В. Піддубного системі Control (див. [4], [6]). Потужна система ідентифікації відповідей вирізняє систему Control серед сучасних комп'ютерних систем навчання і контролю знань. Зокрема, в ній є допустимими відповіді натуральними мовами (з урахуванням можливості переставлення і заміни слів, наявності у відповіді граматичних помилок, і таке інше); відповіді – математичні формули (з точністю до еквівалентності), та багато інших типів відповідей.

Діапазон можливих типів тестових питань поширюється, коли враховується специфіка ідентифікації відповідей неелемента-

рної структури. Для оцінки відповіді, що складається з частин, які комп'ютер здатен розрізнити, кожна з частин можна оцінити окремо, а загальну оцінку отримати в результаті наступного об'єднання часткових оцінок. Алгоритми пошуку у відповіді її складових частин істотно залежать від характеру останніх. Одну з численних груп тестових питань «з частинами» складають питання, відповіді на які містять водночас текст і символічний вираз. Інша, особливо значуща група питань, припускає відповіді, що поділяються на достатньо самостійні у смисловому відношенні елементи. Нами були запропоновані алгоритми, і на їхній основі розроблені правила висновків для системи Control, призначені для оцінки двох зазначених типів відповідей ([1], [2]).

Для оцінки ефективності СКТЗ Control при організації автоматизованого контролю знань з курсу інформатики для вищих технічних навчальних закладів було проведено педагогічний експеримент. Означений експеримент було здійснено на базі ОНАЗ ім. О.С. Попова на протязі 2000–2003 років.

На формуючому етапі експерименту (2002–2003 н.р.) взяли участь студенти 1-го курсу ОНАЗ усіх спеціальностей (загалом 515 чоловік). Експеримент проводився з метою перевірки висунутої гіпотези про те, що *система Control, яка використовує раніше розроблені, а також запропоновані автором правила висновку, здатна розрізняти вірні і невірні відповіді на питання з курсу «Інформатика» таким чином, що оцінка, виставлена нею, буде достатньо близькою до оцінки викладача на усному іспиті.*

Попереднє тестування. На початку експерименту всі групи тестувалися за допомогою дидактичного тесту рівня знань. До множини тестових завдань були включені завдання, що відносяться до загальної початкової обізнаності щодо інформаційних технологій, а також завдання з теми «Лінійні програми». Підсумки тесту дозволили врахувати помилку даних експерименту, пов'язану з нееквівалентністю експериментальних і контрольних груп.

У тесті використовувалися, наприклад, такі питання:

1. *Допишіть пропущене слово (або слова), щоб отримати ПРАВИЛЬНЕ твердження:*

ОДИН БАЙТ МІСТИТЬ _____ БІТ

2. *Напишіть МАТЕМАТИЧНУ ФОРМУЛУ, що відповідає*

наступному оператору Pascal'я: $y := \arctan(x/\sqrt{1-\text{sqr}(x)})$

Відповідь: _____.

3. Нехай $A=5$, $B=4$, $C=12$, $D=2$. Напишіть ЧИСЛА, що будуть результатами наступних арифметичних операцій над даними ЦІЛОГО типу:

1. $A*B \text{ DIV } (C-D)$ 2. $(B+D) \text{ MOD } B$

Відповідь: 1. _____ 2. _____

Вплив. Після вивчення кожної з двох нових тем курсу – «Розгалужені програми» та «Циклічні програми» половина груп тестувалася за допомогою системи Control, а в інших групах проводилося письмове опитування. Задачі комп'ютерного тесту і письмового опитування були майже ідентичними – з точністю до поправок, які потрібно вносити у завдання комп'ютерного тесту, щоб врахувати специфіку використання дисплею і клавіатури замість ручки і паперу.

Припускалося, що студенти з експериментальних груп на час проведення підсумкового опитування у достатній мірі оволодіють навичками використання системи комп'ютерного тестування. При цьому адаптація до системи Control не займала значного навчального часу, так що студенти контрольних і експериментальних груп встигали оволодіти однаковою кількістю матеріалу. Заняття в контрольних і експериментальних групах відрізнялися тільки тим, що, в той час, коли студенти експериментальних груп тренувалися давати відповіді на питання комп'ютерного тесту, студенти контрольних груп відповідали на письмові або усні питання викладача.

Система Control має два різні режими роботи – тренувальний та екзаменаційний. При використанні системи Control у режимі тренування студент в разі подання ним неправильної відповіді бачить на екрані вірну (“зразкову”) відповідь; замість цього студенти контрольних груп отримують назад свої самостійні роботи, перевірені викладачем разом з його зауваженнями і вправленнями допущених помилок.

Наведемо приклади кількох питань з сеансів комп'ютерного тестування і письмового опитування.

Завдання з письмового опитування з теми «Розгалужені програми»:

Напишіть імена, якими позначаються у Pascal'і логічні (бу-

ліві) константи.

Відповідь: _____.

Аналогічне питання у сеансі комп'ютерного тестування було таким:

Запишіть (через КОМУ або ПРОБІЛ) імена, якими позначаються у Pascal'і логічні (булеві) константи.

Як видно з прикладу, завдання відрізняються тільки наявністю в комп'ютерному варіанті вказівки на необхідність вживання у відповіді певного розділювача слів.

Ще одне завдання з теми «Розгалужені програми»:

Запишіть на Pascal'і ЛОГІЧНИЙ ВИРАЗ, що матиме значення “істина” у випадку, коли число X знаходиться ПОЗА напівсегментом $[A, B)$.

У сеансі комп'ютерного тестування при застосуванні системи Control це питання може бути використане без будь-яких змін. В даному випадку студенту пропонується ввести відповідь у довільній формі; і можна показати, що будь-яка його відповідь буде адекватно оцінена системою.

Ось приклад двох завдань з теми «Циклічні програми»

Змінні яких БАЗОВИХ ТИПІВ Pascal'я можна використовувати як параметри циклу FOR?

Запишіть імена цих типів.

Відповідь: _____.

Визначте ЧИСЛОВЕ ЗНАЧЕННЯ змінної S після виконання наступних операторів:

```
s:=2.1; i:=5;
```

```
while (i<4) do begin i:=i+1; s:=s*i end;
```

Відповідь: _____.

В тестах з обох тем були присутні питання, відповіді на які у сеансі комп'ютерного контролю оцінювались за допомогою запропонованих автором правил висновку. Наприклад, у першому з наведених вище запитань до теми «Циклічні програми» потрібно було поділити відповідь на три частини: 1) «цілого типу» (або «типу integer»); 2) «символьного типу» (або «типу char»); 3) «логічного типу» (або «типу boolean»). (Зверніть увагу, що порядок розміщення всіх цих частин у відповіді студента може бути довільним). Якщо у відповіді зустрічався вираз: «дійсного типу» (або «типу real») чи «рядкового типу» (або «типу string»),

то відповідь оцінювалась як невірна.

Контрольне тестування. Студенти, які брали участь у тестуванні або у письмовому опитуванні, допускалися до підсумкової контрольної роботи за вивченими новими темами. Контрольна складалася не з тестових питань, а з задач, що передбачали написання програми мовою Pascal. Завдання з контрольної роботи були однакові для всіх груп, як контрольних, так і експериментальних. Ось приклад одного завдання з контрольної роботи:

Написати програму, що дозволяє ввести з клавіатури послідовність з 7 цілих чисел та знаходить КІЛЬКІСТЬ НУЛІВ у цій послідовності.

Проведення контрольної роботи регламентувалося відповідною інструкцією для викладачів. З метою досягнення одноманітності критеріїв оцінки контрольних робіт, усі контрольні роботи були перевірені одним викладачем кафедри.

Оцінка результатів. Контрольне тестування повинно було показати:

1) Співвідношення рівня знань студентів у контрольній і експериментальній групах. Припускалося, що в середньому студенти, які використовували при навчанні систему Control, покажуть результати не гірші, а можливо й кращі за студентів, які навчалися без застосування системи.

2) Результати контрольної роботи, вочевидь, повинні корелювати з підсумками здачі тесту або письмового опитування. Очікувалося, що в середньому відповідність між оцінками студентів в групі за контрольну роботу та підсумками комп'ютерного тестування мало відрізнятиметься від залежності між результатами контрольної роботи і підсумками письмового опитування.

Можна було припустити, що при не вповні адекватних параметрах налагодження оцінки відповідей системою Control (множина цих параметрів називається “дескриптором допустимої відповіді” [4], [6]), результати комп'ютерного тестування – порівняно з результатами письмового опитування – могли бути в середньому завищеними, заниженими, або ж відрізнятися у якийсь інших відношеннях.

Поточні та підсумкові результати експерименту було проаналізовано за допомогою методів математичної статистики. Ви-

явилося, що в цілому відповідність між оцінками студентів у групі за контрольну роботу та підсумками комп'ютерного тестування статистично несуттєво відрізняється від залежності між результатами контрольної роботи і підсумками письмового опитування.

Враховуючи отримані у ході проведення експерименту дані, можна стверджувати, що комп'ютерне тестування за допомогою системи Control дає не гірші результати, ніж «паперова технологія» проведення аналогічних тестів. Але, оскільки комп'ютерна технологія:

- 1) не потребує перевірки виконаних завдань викладачем;
- 2) не вимагає великого обсягу друкованих матеріалів;
- 3) а також надає студентам можливість майже необмеженого за часом тренування;

то можна говорити про переваги і доцільність застосування саме комп'ютерного варіанту контролю знань.

Отримані результати експерименту дозволяють рекомендувати викладачам інформатики у вищих технічних навчальних закладах використовувати СКТЗ Control в якості:

А) зручного і корисного тренажера, придатного для закріплення і повторення пройденого матеріалу;

Б) ефективного засобу автоматизованого підсумкового та поточного контролю знань з усіх тем курсу «Інформатика».

Експериментальне підтвердження висунутої гіпотези дозволяє зробити висновок, що застосування системи Control у зазначених функціях повинно сприяти більш ефективному використанню робочого часу викладачів, який буде витратитися більшою мірою не на перевірку виконання студентами простих завдань, а на додаткове консультування студентів з складних питань курсу.

Література

1. Баранов В.Ю. Методи оцінювання семантично різномірної відповіді у системі комп'ютерного тестування знань // Комп'ютерно-орієнтовані системи навчання: Зб. наук. праць. – К.: НПУ ім. М. П. Драгоманова, 2002. – Вип. 5. – С. 97-110.
2. Баранов В.Ю. Оценка синтаксически разнородного ответа в системе компьютерного тестирования знаний // Научный вестник

- ник ПДПУ ім. К. Д. Ушинського. – Одеса: ПДПУ ім. К. Д. Ушинського, 2001. – Вип. 10-11. – С. 29-34.
3. Беспалько В.П. Элементы теории управления процессом обучения. Ч. II. Измерение процесса обучения. – М.: Знание, 1971. – 72 с.
 4. Леоненко Л.Л., Поддубный Г.В. Теория подобия конечных последовательностей и ее приложение к распознаванию образов // Автоматика и телемеханика. – М., 1996. – №8. – С. 119-131.
 5. Талызина Н.Ф. Теоретические основы контроля в учебном процессе. – М.: Знание, 1983. – 96 с.
 6. Поддубный Г.В. Интеллектуальные системы и их использование в компьютерных технологиях обучения (на примере автоматизированной системы тестирования знаний Control). – Одесса: УГАС им. А. С. Попова, 1997. – 28 с.

ИСПОЛЬЗОВАНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В ОБУЧЕНИИ ИНФОРМАТИКЕ

Е.В. Белова, А.С. Мустафа, Д.А. Фадеев
г. Харьков, Украинская инженерно-педагогическая академия
ikt@uipa.kharkov.org

Информационные технологии (ИТ) в информатике играют двоякую роль: с одной стороны, они составляют предмет изучения, с другой стороны, могут выступать как средство изучения дисциплины. В качестве последнего информационные технологии обеспечивают индивидуализацию обучения с учетом психофизиологических свойств обучаемого, его уровня подготовки, возможность самостоятельного освоения учебного материала. Для преподавателя – это возможность проведения всестороннего контроля знаний обучаемых.

В Украинской инженерно-педагогической академии (УИПА) принята следующая традиционная система изучения дисциплин цикла «Информатика».

Основной учебный материал по изучению информационных технологий разделен на части (модули). Каждый модуль, как правило, включает лекцию, аудиторный контроль, лабораторную работу и домашнее задание.

Аудиторный контроль проводится обычно в начале лабораторной работы и служит для оценки усвоения ранее пройденного материала. Оценка производится преподавателем, что можно определить с позиций управления как обратную связь.

Лабораторная работа проводится под управлением преподавателя. Здесь управление осуществляется как прямое воздействие в виде ссылок на лекционный или справочный материал, обратная связь – в виде помощи, выдачи разъяснений, подсказок, когда студент не может самостоятельно справиться с заданием, а также оценки результатов выполнения лабораторной работы.

Обязательным в учебном процессе по дисциплине является самостоятельное выполнение домашних заданий, которое предполагает закрепление навыков и умений, полученных в результате очных занятий. Здесь обратная связь реализуется за счет проверки и оценки преподавателем правильности выполнения зада-

ния, а также разъяснения допущенных ошибок и нерациональных подходов к решению.

Следует отметить, что в рассматриваемой системе разработаны и используются в качестве средств обучения отдельные элементы ИТ, такие как электронные методические указания, обучающие и контролирующие программы. Однако основная, ведущая роль принадлежит преподавателю. Это определяет основные недостатки системы:

- качество учебного процесса полностью зависит от квалификации преподавателя;
- дефицит времени индивидуального общения с каждым студентом;
- ориентация учебного процесса на слабых студентов;
- большие затраты времени на выполнение рутинной работы (проверка результатов выполнения однотипных заданий).

Следует признать также неэффективность проведения лекций в больших потоках даже при использовании технических средств общего пользования (плакаты, проекционные устройства с экранами и пр.), что можно объяснить утратой психологической связи лектора с каждым из слушателей. Анализ конспектов даже самых старательных студентов показывает, что они не всегда записывают необходимые пояснения, несмотря на прямые указания лектора. Это снижает ценность конспекта при его дальнейшем использовании.

Исходя из вышеизложенного, можно выделить следующие основные направления применения ИТ в обучении:

- представление учебного материала, традиционно излагаемого на лекциях, в виде электронных учебных пособий (ЭУП);
- использование электронных методических указаний (ЭМУ) при проведении лабораторных работ;
- использование контролирующих систем для оценки усвоения учебного материала.

Опыт показывает, что использование ЭУП как при самостоятельном изучении, так и при групповых занятиях в компьютерном классе, обеспечивает необходимую концентрацию внимания и сосредоточенности, что не всегда достигается в услови-

ях лекции. Кроме того, электронные учебные пособия обычно содержат контрольные вопросы, обеспечивающие самоконтроль обучаемого. В то же время работа с электронными учебными пособиями не исключает потребности в формировании студентом краткого письменного (печатного) отчета-конспекта (ключевые слова, определения и т.п.), что способствует осмыслению и запоминанию изучаемого материала, умению выделять главное. В некоторых случаях это может быть перечень (распечатка) ответов на вопросы для самопроверки.

С точки зрения структуры традиционные методические указания – это просто последовательный текст, относящийся к теме данной лабораторной работы. Ответы на «неясные» вопросы, диагностика и устранение ошибок в процессе выполнения работы, оценка правильности полученного результата – в компетенции преподавателя, который проводит лабораторную работу. При использовании электронных методических указаний эти функции могут быть реализованы программно.

Использование гипертекстовой технологии позволяет обеспечить методические указания подробными ссылками на ранее изученный материал, необходимый для проведения данной лабораторной работы. В гиперссылки закладывается информация о расшифровке возможных сообщений об ошибках и способах их устранения.

При разработке контролирующего программного обеспечения большое значение имеет используемый тип ответа. При широко распространенном тестовом контроле, основанном на выборе правильного ответа из списка, включающего ряд неправильных ответов, существует ряд негативных сторон: возможность угадывания правильного ответа, возможность произвольного запоминания неправильного ответа, трудность подбора «правдоподобных» неправильных ответов. Лучшие результаты дает ответ словом (числом) или фразой, включающей ключевые слова.

Важна также реакция системы на ответ, так называемая «обратная связь» [1]. При простой обратной связи обучаемый получает оценку ответа: «верно» или «неверно». Далее программа может либо отправить его на повторное изучение темы, либо сообщить ему правильный ответ. При сложном типе обратной

связи учащийся не только информируется о правильности ответа, но ему поясняется, почему ответ является неправильным.

Реализация рассмотренных направлений использования ИТ существенно преобразует модель учебного процесса. Так основное управляющее воздействие при всех видах учебных занятий перекладывается на компьютер. Преподавателю отводится роль консультанта. Это несколько не умаляет его роли в учебном процессе, так как только квалифицированный преподаватель может быть главным идеологом такой системы, возможно, ее разработчиком и уж, во всяком случае, пользователем-заказчиком. Кроме того, такой подход к обучению может способствовать переключению преподавателя на решение со студентами творческих задач, связанных, например, с модернизацией и усовершенствованием существующей системы обучения.

В УИПА в состав дисциплины «Информатика» для студентов инженерно-педагогических специальностей включен раздел «Информационные технологии в обучении», в котором они знакомятся с основными принципами проектирования и функционирования компьютерных обучающих систем.

В рамках самостоятельной работы студенты этих специальностей уже в течение многих лет привлекаются к созданию обучающих и контролирующих программ для различных учебных дисциплин, включая и информатику. Ранее разработки выполнялись на базе универсальных языков программирования, редакторов текста, что ограничивало возможности таких программ по представлению информации.

Студентами УИПА, соавторами настоящей статьи разработана информационная система подготовки обучающих программ [2], основанная на использовании новейших информационных технологий.

Использование систем визуального программирования позволяет значительно улучшить интерфейс обучающих программ, обеспечить простоту и наглядность использования в соответствии с современными требованиями [3].

Опыт показывает, что участие студентов – будущих инженеров-педагогов в разработке программного обеспечения для компьютерного обучения способствует как формированию профессиональных педагогических навыков, так и углубленному изуче-

нию учебного материала дисциплины «Информатика».

Следует отметить, что эмпирическое исследование эффективности обучения с помощью компьютера, проведенное зарубежными специалистами, подтверждает целесообразность и необходимость расширять применение ИТ в обучении. Так, в [1] сообщается, что обучение с помощью компьютера привело к 30%-ному снижению времени обучения, к 3% снижению накладных расходов и к увеличению интенсивности деятельности учащегося с 15 до 66%. Отмечается также удовлетворенность учащихся обучением с использованием новых технологий.

Кроме того, в связи с распространением в настоящее время идей дистанционного обучения разработка и использование электронных учебных пособий и электронных методических указаний является чрезвычайно актуальной проблемой.

Из вышеизложенного следует, что действующая система обучения информатике требует существенной перестройки в отношении расширения использования компьютера как средства обучения. Для этого необходимы разработка соответствующего методического и программного обеспечения, проведение комплекса организационных мероприятий, направленных на изменение структуры и методики проведения занятий.

Литература

1. Крейгер Дж. Ф. Обучение с помощью компьютера (автоматизированное обучение)./ Информационные технологии в бизнесе. Под ред. М. Желены. – СПб: Питер, 2002. – С. 799-809.
2. Мустафа А., Фадеев Д. Информационная система подготовки обучающихся программ. / Жизненные ценности студенчества в начале XXI века: Программа и материалы X Международной студенческой научной конференции, Харьков, 12 апреля 2003 г. – Х.: Изд-во НУА, 2003. – С. 111.
3. Крейгер Дж. Ф. Взаимодействие человека и компьютера./ Информационные технологии в бизнесе. Под ред. М. Желены. – СПб: Питер, 2002. – С. 865-876.

ЕЛЕКТРОННИЙ ПІДРУЧНИК “ПРОГРАМУВАННЯ НА ОБ’ЄКТ PASCAL У СИСТЕМІ ОБ’ЄКТНО- ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ DELPHI 6”

М.А. Бондаренко, О.С. Іванченко
м. Харків, Українська інженерно-педагогічна академія
ikt@uipa.kharkov.org

Міністерство освіти і науки України у грудні 2000 р. запропонувало концепцію дистанційного навчання, яка буде реформувати систему освіти на основі нових прогресивних педагогічних технологій та науково-педагогічних досягнень.

Сучасний стан на ринку освітніх програм в Україні характеризується великими змінами, пов’язаними з новими інформаційними технологіями, новими методами отримання професійних знань.

Традиційні форми системи вищої освіти повинні внести зміни у доставку знань навчаючим. Основним інноваційним напрямком у системі вищої освіти є широке впровадження в усі форми навчання засобів комп’ютерного навчання. Інтерес до комп’ютерного навчання залежить від рівня розвитку обчислювальної техніки. Дослідження показали, що у середньому ефективність комп’ютерного навчання перевищує 30%. Підвищується якість запам’ятовування інформації, розвиток пізнавальних, творчих навичок, уміння орієнтуватися в інформаційному просторі. За допомогою відео-супроводження, гіпертекстових систем, електронних підручників, навчальна інформація стає доступнішою для сприймання та зрозумілішою.

Потреби у спеціалістах нових сфер діяльності потребують нові засоби навчання. Сучасний студент самостійно вибирає засоби та методiku навчання. Зникають застарілі поняття циклічного вивчення матеріалу. Навчальний матеріал подається у вигляді лабораторних робіт, практичних занять, модулів тощо. Тепер слід подавати сучасний навчальний матеріал, який в умовах конкуренції дуже швидко змінюється. Поряд з сучасними підручниками слід розробляти електронні підручники на основі об’єктно-орієнтованого програмування (ООП).

Застосування таких систем ООП, як Delphi, Visual Basic, C++

тощо дозволяє розробляти різні навчальні посібники, що надають можливості:

- а) вивчити не лише змістовий матеріал, але і методичну інформацію, яка раніше не входила в освітні стандарти;
- б) швидкого доступу до необхідного навчального матеріалу.

Студенти потребують навчання за допомогою мережі Internet, безпосереднього спілкування з лектором за допомогою комп'ютера, використання електронних підручників, які мають велику кількість гіперпосилань. Сьогодні в Україні має місце тенденція збільшення кількості користувачів мережі Internet, що дає змогу тісніше спілкуватися із західними навчальними закладами.

Вищий інженерно-педагогічний навчальний заклад не може бути конкурентноспроможним без організації електронного навчального простору. Альтернативні підходи до навчального процесу слід застосовувати із традиційним навчанням в аудиторії. Наприклад, застосування навчальної дисципліни «Прикладне програмування» (теоретичні відомості, лабораторні роботи, більше 30-ти запитань для самостійного вивчення кожної лабораторної роботи, завдання для виконання кожної із 21 лабораторних робіт, багато прикладів виконання кожної лабораторної роботи), оформлено у вигляді електронного підручника (обсягом 33 друк. аркушів).

Студенти академії можуть використовувати цей матеріал у комп'ютерних класах.

Інтерактивне навчання з застосуванням технологій Internet, intranet, multimedia тощо стає одним із предметів навчально-методичної роботи по інформації сфери освіти. Дистанційне навчання стає загальною потребою усього життя. Воно має велике майбутнє.

Дистанційне навчання дає можливість приймати участь у різних інтерактивних дискусіях та використовувати інформацію, яка знаходиться в різних географічних районах. Наприклад, на спеціальності 7.010104.36 «Професійне навчання. Комп'ютерні технології в управлінні та навчанні» навчаються студенти в м. Харкові, в м. Артемівську Донецької обл., в м. Стаханові Луганської обл. Електронний підручник «Прикладне програмування в системі ООП Delphi 6» дає змогу використовувати його в різних

географічних містах. Навчання виконується за індивідуальним розкладом, який надає комп'ютерний канал tutor-учень.

Електронний підручник “Програмування на Object Pascal в системі Delphi 6” побудовано за такою схемою:

1. Розроблено і видано навчальний посібник «Програмування на Object Pascal у середовищі Delphi 6» в обсязі навчального матеріалу 3-го і 4-го семестрів, дисципліни “Прикладне програмування”, спеціальності 7.010104.36 (автор – М.А. Бондаренко).

Написано 21-у лабораторну роботу, кожна із яких містить теоретичний матеріал, методичні вказівки, індивідуальні завдання та приклади виконання лабораторних робіт із усіх розділів дисципліни в обсязі 3-го і 4-го семестрів.

2. методичні вказівки до лабораторних робіт з дисципліни «Прикладне програмування» в обсязі навчального матеріалу 3-го семестру спеціальності “Комп'ютерні технології в управлінні та навчанні” (спеціальність 7.010104.36).

Створено друковані та електронні версії методичних вказівок до наступних лабораторних робіт з дисципліни «Прикладне програмування», зазначених у робочій програмі даної дисципліни в обсязі 3-го семестру:

1) знайомство із середовищем об'єктно-орієнтованого програмування Delphi 6;

2) знайомство з мовою об'єктно-орієнтованого програмування Object Pascal;

3) консольна програма;

4) введення-виведення даних в Object Pascal;

5) лінійний обчислювальний процес;

6) обчислювальний процес, що розгалужується;

7) циклічні обчислювальні процеси.

8) одновимірні масиви.

3. Методичні вказівки до лабораторних робіт з дисципліни «Прикладне програмування» в обсязі навчального матеріалу 4-го семестру.

Створено друковані та електронні версії методичних вказівок до наступних лабораторних робіт з дисципліни «Прикладне програмування», зазначених у робочій програмі даної дисципліни в обсязі 4-го семестру:

1) двовимірні масиви в Delphi 6;

- 2) записи в Delphi 6;
- 3) обробка графічної інформації в Delphi 6;
- 4) рядки та символи;
- 5) розв'язання нелінійних рівнянь;
- 6) метод простих ітерацій;
- 7) метод дотичних;
- 8) чисельне інтегрування.

4. Електронний навчальний посібник з дисципліни «Прикладне програмування» в обсязі навчального матеріалу 3-го та 4-го семестрів.

Створено ЕНП, що дозволяє працювати з електронними версіями лабораторних робіт навчального посібника (конспектами лекцій за темами 3-го і 4-го семестрів та методичних вказівок до цих лабораторних робіт).

Електронний навчальний посібник з дисципліни «Прикладне програмування» дозволяє на досить високому рівні вивчати навчальний матеріал студентам очної форми навчання під час самостійної роботи, студентами заочної форми навчання і використовувати його при дистанційному навчанні.

Розроблений електронний навчальний посібник містить теоретичний матеріал, методичні вказівки, індивідуальні завдання і приклади виконання лабораторних робіт із усіх розділів дисципліни «Прикладне програмування».

РОЛЬ ІНФОРМАЦІЙНОЇ СИСТЕМИ В УДОСКОНАЛЕННІ РЕГУЛЮВАННЯ ІМПОРТНИХ ОПЕРАЦІЙ В УКРАЇНІ

Л.Л. Вакуленко

м. Біла Церква, Білоцерківський державний аграрний
університет

ВАК12@btsau.kiev.ua

Управління економікою ґрунтується на інформації та породжує нову інформацію. Умови ринкової (конкурентної) економіки висувають серйозні вимоги до якості, своєчасності, повноти, вірогідності економічної інформації, глибини аналізу економічних показників. Водночас кожна система існує не відокремлено, а під дією як суміжних систем, так і навколишнього середовища. Кількість таких впливів безмежна, але враховуються тільки ті з них, які суттєво впливають на досліджувані параметри системи. Ці впливи називаються входами [2]. Входи поділяють на керуючі та збурювальні. Оскільки всі системи – взаємозалежні, кожна з них, у свою чергу, впливає на зовнішнє середовище. Особливості цього впливу визначаються виходом системи. Вихід і вхід системи є взаємозалежними, між ними існує прямий причинно-наслідковий зв'язок, що виявляється у функціонуванні системи. Окрім вхідних і вихідних параметрів, система характеризується множиною змінних. При дослідженні системи найбільший інтерес викликає залежність між її входом і виходом. Відповідно зміну вихідних параметрів під впливом вхідних кваліфікують як перетворення системи.

Цілеспрямованим впливом однієї системи (підсистеми) на іншу, який має на меті змінити її поведінку в певному напрямі (відповідно до заданої мети), є управління. З цього випливає, що система, яка реалізує процес управління, складається як мінімум із двох частин керованої і керуючої. Керована система – це виробничо-технічна система, а керуюча – це система вищого рівня. Механізм процесу регулювання імпортними операціями в Україні є дуже складним. Розкрити його зміст допомагає кібернетичний підхід, який розглядає тільки інформаційні процеси [3].

Інформаційна система є сукупністю інформаційних процесів для задоволення потреби в інформації різних рівнів прийняття

рішень. Відповідно вона забезпечує нагромадження, передачу, збереження, оброблення та узагальнення інформації “знизу вгору”, а також конкретизацію інформації “зверху вниз”.

До головних завдань інформаційної системи в регулюванні імпорتنих операцій в Україні належать:

- виявлення джерел інформації;
- збирання, ресстрація, оброблення та видача інформації, що характеризує стан виробництва й управління;
- розподіл інформації.

Найважливіша властивість інформаційної системи – єдність інформації, яка регулюється, що визначає єдине інформаційне забезпечення системи управління. Вхідною інформацією, властивості якої наведено в табл. 1, користуються всі органи управління.

Таблиця 1. Властивості вхідної інформації

<i>Вид властивості</i>	<i>Характеристика</i>
Організаційно-структурна	Відповідає структурі системи управління
Організаційно-економічна	Надійність. Потужність. Пропускна здатність. Усталеність. Економічність. Ефективність
Функціональна	Порядок функціонування, змінюваність у зв'язку з розвитком об'єкта управління, надмірність інформації
Споживча	Порядок взаємодії зі споживачами інформації, своєчасність її доставки, взаємозв'язок і взаємозалежність елементів інформації

Структурно інформаційна система складається з таких компонентів:

- власне інформації;
- системи оброблення інформації;
- входу;
- виходу;
- внутрішніх і зовнішніх каналів.

У Державному Стандарті України ДСТУ 2874-94 дано таке визначення інформаційної системи [4]:

Інформаційна система – система, яка організовує пам'ять і маніпулювання інформацією щодо проблемної сфери.

Характерною рисою інформаційної системи є те, що людина виступає активним учасником інформаційного процесу. Це виявляється в умовах функціонування автоматизованого робочого місця, коли кінцевий користувач здійснює введення інформації в систему, підтримує її в актуальному стані, обробляє інформацію і використовує здобуті результати в управлінні [1]. Інформація служить способом опису взаємодії між джерелом й одержувачем інформації. Те саме повідомлення одному одержувачеві може давати багато інформації, а іншому – мало або нічого.

Потенційні можливості інформаційної системи в удосконаленні регулювання імпорتنих операцій реалізуються через їх функції, до яких належать:

- обчислювальна – вчасно і якісно виконує оброблення інформації в усіх аспектах, що цікавлять систему управління;
- слідкувальна – відстежує і формує всю необхідну для управління зовнішню та внутрішню інформацію;
- запам'ятовувальна – забезпечує безупинне накопичення, систематизацію, збереження і відновлення всієї необхідної інформації;
- комунікаційна – забезпечує передачу потрібної інформації в задані пункти;
- інформаційна – реалізує швидкий доступ, пошук і видачу необхідної інформації;
- регулювальна – здійснює інформаційно-керуючий вплив на об'єкт управління і його ланки при відхиленні їхніх параметрів функціонування від заданих значень;
- оптимізаційна – забезпечує оптимальні розрахунки в міру зміни цілей, критеріїв та умов функціонування об'єкта управління;
- прогнозна – визначає основні тенденції, закономірності та показники розвитку об'єкта управління;
- аналізаторна – визначає основні показники техніко-економічного рівня виробництва господарської діяльності;
- документувальна – забезпечує формування всіх обліково-звітних, планово-розпорядничих, конструкторсько-

технологічних та інших форм документів.

Таким чином, місія інформаційної системи в регулюванні імпортних операцій в Україні – це виробництво інформації, створення інформаційного і технічного середовища.

Розглядаючи дану систему управління, ми виокремили три рівні управління: стратегічний, тактичний та оперативний. Кожний з цих рівнів управління має свої завдання, при вирішенні яких виникає потреба в інформації, тобто інформаційні запити до інформаційної системи. Ці запити звернені до відповідної інформації в інформаційній системі [1]. Інформаційні технології дозволяють опрацювати запити і, використовуючи наявну інформацію, сформулювати відповідь на ці запити. Таким чином, на кожному рівні управління з'являється інформація, що служить основою для прийняття відповідних рішень. Запити до інформаційної системи і, отже, процедури формування відповіді на них можна поділити на рутинні й нерутинні. Рутинні процедури характеризуються заданістю початкової і вихідної інформації, а також визначеністю алгоритму отримання останньої з першої. Виділення рутинних задач і процедур опрацювання інформації дозволяє їх формалізувати, а надалі й автоматизувати.

Отже, в основі інформаційної системи – процес виробництва інформації. Зростання обсягів інформації в інформаційній системі організацій, потреба в прискоренні й більш складних способах її переопрацювання зумовлюють необхідність автоматизації роботи інформаційної системи, тобто автоматизації опрацювання інформації.

Література:

1. Гужва В.М. Інформаційні системи і технології на підприємствах: Навч. посібник. – К.: КНЕУ, 2001. – С. 22.
2. Інформаційні системи і технології в економіці: Посібник для студентів вищих навчальних закладів / За редакцією В.С. Пономаренка. – К.: Видавничий центр “Академія”, 2002. – С. 26-31.
3. Юхименко П.І., Леоненко П.М., Ільєнко А.А. Економічна історія: Навч. посіб. – Біла Церква, 2002. – 320 с.
4. www.ukrstat.gov.ua

ИЗУЧЕНИЕ ЯЗЫКА SIMNET II В РАМКАХ ДЕЯТЕЛЬНОСТНОГО ПОДХОДА

К.Ю. Васильев, А.В. Кривилёв

г. Днепропетровск, Днепропетровский национальный
университет

Деятельностный подход к обучению студентов фундаментальным дисциплинам, в том числе и информатике, можно проводить в ходе решения практических задач путем моделирования сложных систем с использованием пакетов прикладных программ (ППП), таких как Matlab, Modelica, EASY5, SIMNET, TORA и др. К классу сложных систем относят технологические, производственные, энергетические комплексы, а также системы автоматизации управления. На сегодняшний день имитационное моделирование является одним из наиболее мощных средств исследования подобных систем. Разработчики программного обеспечения предлагают современному инженеру целый ряд разнообразных средств моделирования, которые позволяют не только описывать сложные динамические системы, но и проводить с ними эксперименты. Наиболее полное исследование общесистемных проблем можно получить в результате моделирования с применением современных технологий, реализованных в специализированных пакетах визуального моделирования.

В этих пакетах предоставляется возможность описывать структуру системы и её поведение в визуальной форме и отображать результаты экспериментов в виде графиков, гистограмм, схем, анимационных роликов и других наглядных форм. Кроме того, в рассматриваемых пакетах реализована технология объектно-ориентированного моделирования, которая нацелена на повторное использование экземпляров моделей с внесёнными изменениями.

Из большого разнообразия существующих в настоящее время пакетов визуального моделирования повышенный интерес вызывают специализированные ППП, так как при изучении технических предметов они во многом опережают универсальные пакеты, например, по объёму и времени освоения.

Примером специализированного пакета является SIMNET II

[1]. Этот процедурно-ориентированный язык разработан для имитации дискретных моделей, которые можно рассматривать как систему очередей. Язык сформирован на сетевом принципе, который использует четыре типа узлов: узел **источника**, в которого создаются транзакции (запросы на обслуживание), узел **очереди**, где они ожидают обслуживания, узел сервисов, где выполняется **обслуживания**. Четвёртый, **дополнительный**, тип узла предназначен для расширения возможностей языка.

Узлы SIMNET II соединены **дугами**. Дуги выполняют следующие функции: 1) управляют переходом транзакций между узлами, 2) занимаются сбором статистических данных, 3) производят арифметические расчёты.

В процессе имитационного моделирования SIMNET II отслеживает последовательную обработку транзакций, помещая информацию о них в файлы трех типов: календарь событий, очередь, средства обслуживания.

Рассмотрим объект и постановку задачи его анализа в среде SIMNET II, например, исследование временных характеристик при передаче информации по интеллектуальным сетям (ИС) связи [2]. Существенным моментом для ИС является особенность, состоящая в том, что скорость обработки вызовов падает с ростом их «интеллектуальности». Эту проблему можно считать актуальной, поскольку её решение позволяет ускорить и упростить процесс ввода новых и коррекции существующих информационных услуг, уменьшить задержки передачи информации, что сказывается на экономичности ИС, увеличении пропускной способности телекоммуникационных сетей и повышении качества обслуживания абонентов.

На рисунке 1 показана укрупненная схема построения ИС.

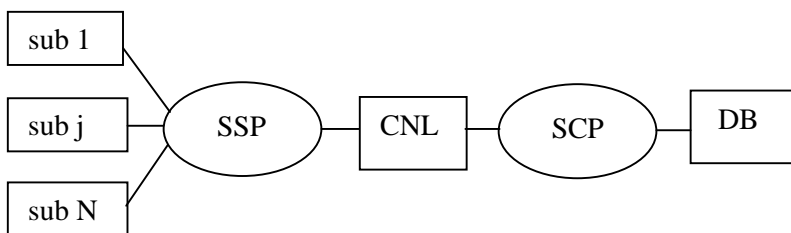


Рис. 1. Укрупненная схема ИС

Здесь sub j – j -ый абонент, $j = 1..N$; SSP – узел коммутации услуг; CNL – сеть сигнализации; SCP – узел предоставления услуг; DB – база данных. В узлах SSP и SCP и в канале CNL учитываются задержки и случайные появления (с известными функциями вероятностных распределений) помех и сообщений.

В качестве примера рассмотрена ИС, на входе которой сообщения появляются в соответствии с экспоненциальным распределением со средним 7. Время передачи информации подчиняется распределению Пуассона со средним 6. Случайное воздействие помехи на сигнал описывается равномерным распределением на временном интервале [600, 650]. Искаженная информация передается повторно.

С применением SIMNET II можно получить для данного примера следующие результаты: средняя временная задержка + значение ожидания – 21.88; процент транзакций с нулевым временем ожидания – 13.85; максимальное время на передачу сообщения – 96.55; среднее время использования средства обслуживания – .8141 и т. п.

ППП SIMNET II представляет собой гибкий инструментарий имитационного моделирования. Может широко использоваться при анализе работы банков, проверке радиоаппаратуры, оптимизации загрузки автотранспорта, в сфере связи.

ППП SIMNET II выполняется в режиме MS DOS, эмулированном в среде Windows, и требует до 0.5 Мбайт оперативной памяти.

ППП SIMNET II доступен в сети Internet по адресу: www.williamspublishing.com. Более детальные сведения представлены на сайте [3].

Работа в среде SIMNET II носит научно-исследовательский характер и готовит студентов к осуществлению будущей профессиональной деятельности по избранной специальности.

Литература

1. Таха Х.А. Введение в исследование операций. – М.: Издательский дом «Вильямс», 2001. – 912 с.
2. Лихтциндер Б.Я., Кузякин М.А., Росляков А.В., Фомичев С.М. Интеллектуальные сети связи. – М.: Эко-Трендз, 2000. – 205 с.
3. Internet – адрес: <http://web.ineg.uark.edu/simnet2/simnet.htm>

ОБЩИЕ ТЕНДЕНЦИИ РАЗВИТИЯ ЭКСПЕРТНЫХ СИСТЕМ В ИССЛЕДОВАНИИ УЧЕБНОГО ПРОЦЕССА

И.Н. Вдовиченко

г. Кривой Рог, Криворожский филиал Института экономики и новых технологий
VIV@alba.dp.ua

Введение

Хотя экспертные системы считают новой технологией, но концепции их построения стары. Еще 700 лет до нашей эры ученые Вавилонии разработали сложную систему правил вида «если-то», описывающую эмпирические взаимосвязи между явлениями в жизни общества. В конце тридцатых годов XX столетия британский ученый Алан Тьюринг утверждал, что машина должна не просто обрабатывать вычисления, а моделировать процесс рассуждения. Алан Ньюэлл и Герберт Саймон из университета Карнеги-Меллона предложили схему процессов, лежащих в основе решения проблем человеком. Все это подготовило базу для появления первых экспертных систем.

1. Периоды развития экспертных систем

Историю исследований в этой области, начиная с 1950 года и по сей день, можно разделить на 4 периода [1].

1. Классический период

Фундаментальная идея этого периода получила название поиска в пространстве состояний. Множество проблем можно рассматривать в таком порядке:

1. Исходное состояние проблемы.
2. Тест завершения. Проверка, достигнуто ли конечное состояние.
3. Множество операций, которые можно использовать для изменения текущего состояния.

Одним из способов представления такого концептуального пространства есть граф, в котором состояния соответствуют узлам, а операции – дугам. Поиск решения аналогичен поиску пути на графе в пространстве состояний. Для решения использовался алгоритм с двумя основными состояниями: поиск в глубину и

поиск в ширину. Это способ грубого перебора. Данные программы нельзя отнести к классу систем, основанных на знаниях, они относятся к программам, обладающим способностью избирательно анализировать пространство состояний, что значительно повышает скорость и эффективность анализа.

В области машинного поиска логического вывода были достигнуты успехи в генерации формальных математических доказательств, но они бесплодны в менее формализованных областях. Для экспертных систем моделирование путей поиска решений должно использовать вероятностные гипотезы человека.

Возникла необходимость использовать знания в конкретной предметной области. Для преодоления трудностей, связанных со сложностью формализованного поиска в пространстве состояний, развитие экспертных систем пошло по пути привлечения опыта экспертов. Рассматривались как детали поведения объектов в конкретной ситуации, так и стратегия логического вывода в определенной предметной области.

II. Романтический период

С середины 60-х до середины 70-х внимание исследователей сосредоточилось на способности машины воспринимать естественный язык человека, вести осмысленный диалог. Кульминационным моментом этого периода была разработка Т. Виноградом в 1972 г. системы SHRDLU, которая понимала большое множество слов английского языка и делала выводы в ограниченной области. Шанк и Колби в 1973 году разработали структуру сценариев для описания поведения в реальной ситуации. Применение сценариев можно рассматривать в свете проблемы представления знаний на более высоком уровне, чем в процедурном варианте. В сценариях необходимо, кроме эпизода в целом, учитывать множество мотиваций, подразумеваемых, но не формулируемых человеком вслух. Эта зависимость от контекста и является главной проблемой компьютерного понимания естественного языка.

В этот период были предложены следующие конструкции представления знаний в экспертных системах: логическая модель, продукционная модель, фреймовая модель, модель семантической сети.

Без результатов, полученных в это время, вряд ли разработ-

чики современных экспертных систем располагали бы таким разнообразием функций и структур.

III. Период модернизма

Этот период продолжался с середины 70-х до конца 80-х годов. Он характеризовался значительным прогрессом в области экспертных систем и так называемой «зимней спячки» в области «чистого» искусственного интеллекта.

На этом уровне стало очевидным, что эвристические возможности экспертных систем определяются представлением в явной форме соответствующих знаний, доступных программе, а не применением каких-то сложных оценочных функций.

Основные усилия были направлены на разработку методов разбиения знаний эксперта-человека на модули. Они извлекаются по мере выяснения, как поступают в типичных ситуациях, а потом к ним прибавляются исключения из таких ситуаций. Такую систему несложно модифицировать и функционально расширить. Изменение знаний не влечет за собой изменение основного программного кода.

IV. Период постмодернизма

Время с конца 80-х гг. до наших дней можно отнести к этому периоду. Происходящее в нем во многом определяется развитием Internet-приложений, в частности интеллектуальных агентов и советчиков, упрощающих извлечение информации. Активно используются экспертные системы для обработки текстов и изображений, которые надо извлекать из разных источников, анализировать, классифицировать, индексировать, обобщать, интерпретировать и т.п.

Эти подходы должны сочетаться со статистическими и вероятностными подходами, поскольку сейчас приходится иметь дело с огромными и все увеличивающимися объемами информации, доступной по Internet и другим информационным сетям.

Еще на ранних этапах развития экспертных систем высказывались идеи, что экспертная система может содержать не одну форму представления знаний. Современные экспертные системы ознаменовали дальнейшее продвижение по этому пути – в них объединились традиционные программы решения проблем и элементы самообучения, критического анализа. В них могут использоваться разные методики, например анализ прецедентов и

анализ пояснений, методы, основанные на подобии и т.д. К этому периоду относится появлением многих гибридных систем, таких как SCALIR, ANAPRON, CBR Express, ODYSSEUS и др. Обозначился новый подход в системах, охватывающих все области жизни современного общества – педагогику, юриспруденцию, социологию. Традиционный подход, требующий создания баз знаний огромного объема, часто нетривиальных, очень проблематичен. Поэтому была сделана попытка использования нейронных сетей для сочетания статистических методов, баз знаний и смысловых связей между объектами. Примеры этому системы Neuron Data, SCALIR. Некоторые исследователи надеются, что в дальнейшем развитие экспертных систем будет продвигаться в этом направлении [3].

2 Анализ экспертных систем

Экспертные системы, являясь частью искусственного интеллекта, существенно отличаются от обычных систем обработки данных.

Во-первых, традиционные компьютерные программы детерминированы, а экспертная система строит собственный алгоритм решений для достижения каждой новой цели.

Во-вторых, в отличие от традиционной программы, которая строится главным образом из линейных отношений, экспертная система обрабатывает произвольные символьные выражения. Цель экспертной системы состоит в выдаче рекомендаций, основанных на предсказываемом поведении объектов и событий.

В-третьих, традиционная программа следует математическим правилам, а экспертная система строится на эвристических рассуждениях, что существенно для оценки учебного процесса.

В-четвертых, экспертная система предоставляет пользователю информацию о том, как найденное системой решение было получено.

Все экспертные системы основываются на базах знаний, полученных из опыта экспертов, используют символьные выражения фактов и соответствующих атрибутов, имеют эвристическую машину вывода, могут работать с недоопределёнными знаниями, решают проблему, используя человеческий опыт. Рассматривая экспертные системы, почти всегда недооценивают их возможно-

сти. Ряд хорошо описанных экспертных систем решают свои задачи на таком же уровне, что и человек-эксперт и выше.

Сейчас экспертные системы используются практически во всех отраслях коммерции и управления: от финансовых служб и производства до систем военного назначения и образования.

Выводы

Экспертные оценки в исследовании учебного процесса – это особый, специфический раздел со своими методами и способами. Его надо рассматривать, как способ, который расширяет диапазон критериев выбора решений. Экспертиза в педагогике должна обеспечивать гармоничность отношений, искореняя возможные конфликты. Ее важной функцией есть опережение принятия каких-либо серьезных решений. Следовательно, важность правильно выполненной экспертизы трудно переоценить.

Необходимо отметить, что сейчас фактически не существует в науке адекватных моделей и подходов, которые описывают сложные переходные процессы, учитывающие интересы личности, общества, государства. Они только разрабатываются, опробуются в научных центрах представителями различных направлений.

Литература

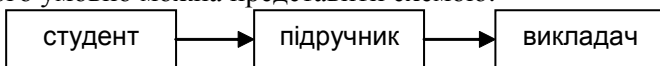
1. Джексон П. Введение в экспертные системы. – М.: Вильямс, 2001. – 622 с.
2. Нейлор К. Как построить свою экспертную систему. – М.: Энергоатомиздат, 1991. – 286 с.
3. Гаврилова Т.А. Базы знаний интеллектуальных систем. – Санкт-Петербург: Питер, 2001. – 382 с.

СУЧАСНИЙ СИСТЕМНИЙ ПІДХІД ДО ВИКЛАДАННЯ ОСНОВ ПРОГРАМУВАННЯ

В.В. Войтенко, А.В. Морозов
м. Житомир, Житомирський державний технологічний
університет
mav@zt.ukrtel.net

Швидкість процесів інформатизації суспільства вимагає створення інформаційного середовища, впровадження у навчальний процес нових методик і технологій.

Раніше наголос ставився на тому, що спочатку студент слушає викладача, і тільки потім вивчає підручник. В даний час перевага надається особистісно-орієнтованому навчанню, парадигму якого умовно можна представити схемою:



Саме таку структуру має система освіти в передових країнах світу. Студент повинен навчитися сам відшукувати і опанувати необхідний матеріал. Викладач є організатором самостійної пізнавальної діяльності учнів та при необхідності консультантом, що вимагає від нього високої компетентності і майстерності.

В області програмування останнім часом відбуваються зміни, які й зумовили необхідність оновлення навчальних планів та програм для підготовки студентів початкових курсів, що навчаються за спеціальностями 7.080403 “Програмне забезпечення автоматизованих систем” та 7.091401 “Системи управління і автоматички” у Житомирському державному технологічному університеті. Починаючи з часів заснування факультету інформаційно-комп’ютерних технологій, в перші дисципліни загальної спеціалізації з програмування у вузівську програму незмінно включалася алгоритмічна мова Паскаль. Вона дійсно, у порівнянні з іншими мовами програмування високого рівня, найкраще підходила для початкового ознайомлення студентів молодших курсів з основами алгоритмізації та програмування. Проте час багато що змінює: мова Паскаль стала частиною вивчення предмету “Інформатика” у старших класах середньої школи. Таким чином, переважна більшість вчорашніх школярів, приходячи до

вузу на перший курс, вже має не лише початкові навички у програмуванні, а й відповідний чималий досвід програмування мовою Паскаль.

Виходячи з вищевказаних об'єктивних причин, замість вивчення основ програмування протягом першого семестру на прикладі мови Паскаль, було віддано перевагу мові Сі, яка раніше розглядалася, починаючи з другого семестру. Також для подальшого вивчення мов програмування PHP, Perl, Java, JavaScript та ін. потрібне знання синтаксису мови Сі.

Для опанування студентами курсу “Алгоритмічні мови та основи програмування” ми прагнули створити такий підручник, який був би доступним за розумінням студентам із різним ступенем підготовки. Також не менш важливою причиною було й те, що з даного предмету майже зовсім немає україномовних підручників.

Перша частина даного посібника містить стислий, та водночас досить повний виклад мови Сі у відповідності до її стандарту ISO/IEC 14882. На простих прикладах показано засоби застосування мови для розв'язання практичних задач. Усі теоретичні відомості супроводжуються простими та зрозумілими прикладами. Викладення матеріалу за зростанням від простого до більш ускладненого допоможе краще зорієнтуватися тим студентам, хто знайомий з програмуванням на іншій мові. Крім того, до розділу включено опис усіх блочних мовних конструкцій побудови алгоритмів, що може бути особливо корисним початківцям.

Друга частина посібника (II семестр вивчення) присвячується мові С++, яка раніше розглядалася на старших курсах. На сьогодні об'єктно-орієнтоване програмування вже не є новою парадигмою, яка останнім часом зазнає суттєвих змін та модифікацій. На численних прикладах, наведених у другій частині посібника, студенти ознайомлюються з положеннями об'єктно-орієнтованої парадигми та засобами їх реалізації мовою С++.

У третьому розділі посібника містяться завдання для лабораторних та практичних робіт, які згруповані за темами у варіантах. Студенти, навчаючись другий рік по даному посібнику, позитивно оцінили приблизно однакову їх складність, що виключить випадкову упередженість при виборі варіанту. Також є позитивним, що студенти із зацікавленістю розв'язують задачі на

складання ефективних алгоритмів, які у попередні роки пропонувалися на шкільних та вузівських олімпіадах з програмування і які теж наведені у посібнику. Це дозволяє викладачу здійснювати індивідуальний підхід до навчання студентів.

У додатках наведено прототипи найбільш широкоживаних функцій мови Сі та С++, які згруповані за належністю до стандартних бібліотек. При безпосередньому написанні програм цей розділ допоможе уникнути труднощів, пов'язаних з використанням довідників та вбудованого HELPa мови, особливо для тих, хто недостатньо володіє англійською мовою.

У даному посібнику не розглядається програмування під Windows та інші середовища, що є предметом вивчення інших дисциплін.

Враховуючи побажання студентів та викладачів, нами було створено електронний варіант посібника, який можна знайти на сайті університету та на диску-додатку.

Електронні підручники стали реаліями часу. Вони мають переваги перед традиційними:

- дають можливість студентам самостійно вивчати навчальний матеріал;
- заощаджують час студентів на пошук необхідної літератури в бібліотеці;
- відрізняються меншою вартістю;
- мають більшу мобільність при оновленні змісту (легко коригувати та вдосконалювати).

Література

1. Войтенко В.В., Морозов А.В. С/С++. Теорія та практика. – Житомир: ЖДТУ, 2003.
2. Освіта. – 2003. – № 9.

ПІДГОТОВКА ІНЖЕНЕРІВ-ПЕДАГОГІВ В УНІВЕРСИТЕТІ

Т.В. Волкова

м. Бердянськ, Бердянський державний педагогічний університет

Методика професійного навчання є невід'ємною частиною загального блоку професійної підготовки інженера-педагога. Вона спирається на спеціальні знання і уміння студентів, придбані при вивченні курсів педагогіки, психології, спеціальних комп'ютерних курсів: теорії інформації, історії розвитку обчислювальної техніки, програмного і апаратного забезпечення, теорії алгоритмів, методів програмування, принципів побудови баз даних, логічного програмування баз знань.

Основними вихідними положеннями в структурі підготовки інженерів-педагогів комп'ютерних технологій є: підпорядкування мети підготовки спеціалістів сучасному стану і тенденціям інформатизації суспільства, широкому впровадженню комп'ютерних технологій в усіх сферах соціальної практики, розвитку нової парадигми управління і гуманістичних моделей управління працею, інформатизація освіти як комплекс соціально-педагогічних перетворень [1, 2, 4, 6, 9-12, 16], пов'язаних з насиченням освітніх систем інформаційною продукцією, засобами і технологією з метою раціоналізації інтелектуальної діяльності, радикального підвищення ефективності та якості підготовки спеціалістів з новим типом мислення відповідно до вимог інформаційного суспільства і формування нової інформаційної культури мислення шляхом індивідуалізації навчання [13-15].

Відповідно до сучасного стану і тенденцій інформатизації суспільства слід визначити основні поняття в структурі цілей, змісту, форм і методів навчально-виховного процесу: роль інформації в суспільному розвитку; теорія управління, її функціональний підхід до об'єктів управління і гуманістичні моделі управління працею; комп'ютерні технології як засіб створення автоматизованої інфраструктури суспільства; інформаційна культура як складова професійної культури інженера-педагога (рис. 1).

На основі цих понять слід визначити особливості реалізації дидактичних принципів підготовки інженерів-педагогів.



Рис. 1. Взаємозв'язок основних понять

1. Політехнічний принцип професійного навчання реалізується завдяки формуванню загальнопрофесійних політехнічних і спеціальних знань про наукові основи інформатизації суспільства, про принципи функціонування сучасної техніки та основи комп'ютерних технологій.

2. Принцип професійної мобільності реалізується завдяки такій побудові змісту освіти, при якій спеціаліст здатен освоювати нові спеціальності та нові комп'ютерні технології, підвищувати свою кваліфікацію в умовах швидко змінюваних поколінь комп'ютерної техніки і впровадження нових комп'ютерних технологій.

3. Принцип моделювання професійної діяльності в навчальному процесі реалізується завдяки застосуванню таких форм організації навчального процесу, при яких один вид діяльності виступає способом розв'язання предметних задач іншого з урахуванням складних міждисциплінарних зв'язків між основними складовими, які визначають зміст професійної діяльності: рух інформаційних потоків, застосування комп'ютерних систем для їх переробки, функціональний підхід теорії управління до впровадження комп'ютерних технологій, відповідність гуманістичним моделям управління.

4. Принцип поєднання навчання і виробничої праці реалізується завдяки виробничій практиці, у процесі якої відбувається практичне оволодіння сучасних проблем формування і використання інформаційних ресурсів, застосування на практиці

положень теорії інформаційних потреб, інформаційне, апаратно-програмне забезпечення сучасних комп'ютерних систем, застосування методів планування і проектування технологічних процесів переробки інформації; набуття навиків організаторської діяльності в галузі інформатизації підприємств і організацій, системного аналізу об'єктів впровадження комп'ютерних технологій.

5. Принцип економічної доцільності професійного навчання забезпечується також виробничою практикою, яка має зв'язок освіти з виробництвом, що сприяє плануванню підготовки в навчальних закладах спеціалістів з урахуванням потреби в них на ринку праці.

6. Принцип комп'ютеризації навчального процесу реалізується введенням до навчального процесу комп'ютерно-орієнтованих і телекомунікаційних технологій.

Метою вивчення курсу є ознайомлення студентів з сучасною концепцією багатоступеневого неперервного навчання в умовах інформатизації освіти; формування готовності до професійної діяльності в умовах інформаційного середовища; формування менталітету особистості; формування здатності до самореалізації особистості, яка володіє теорією і практикою комунікацій в інформаційному суспільстві; формування потреби і здатності до неперервної самоосвіти на основі отриманого рівня інформаційної освіти в умовах швидкого розвитку інформатизації; підготовка майбутнього інженера-педагога до організації та проведення управлінської діяльності з використанням комп'ютерних технологій за напрямками: ведення статистичних баз даних; планово-економічне забезпечення; ведення архівів; планування і контроль за виконанням прийнятих рішень; діловодство; інформаційно-правове забезпечення; діагностика і моніторинг навчально-виховного процесу; моделювання процесів управління системою; моніторинг управління.

До теоретичної бази знань відносяться: знання про інформацію, інформаційні процеси, цілі й завдання застосування комп'ютерних технологій в навчальних закладах освіти та на підприємствах і організаціях; функції управління закладами освіти; сутність концепції мотиваційно-цільового управління; основи дистанційного навчання [1-4, 6, 9-14, 17].

Практичні навички використання засобів сучасних комп'ютерних технологій передбачають уміння складати плани на поточний рік; здійснювати оперативне планування навчально-виховного процесу; складати тематичні плани і конспекти уроків для різних розділів курсів комп'ютерних технологій; готувати демонстраційні електронні дидактичні матеріали до уроків, моделювати і аналізувати уроки; використовувати ресурси Інтернету та Інтранету для організації самостійної роботи і підготовки до уроків; визначати форми і методи оптимального здійснення контрольної-оцінної діяльності, використовувати модульно-рейтинговий контроль; планувати процес використання засобів комп'ютерних технологій для організації зворотного зв'язку у системі «педагог-студент»; вести статистичні бази даних студентів і співробітників; створювати і супроводжувати базу даних бібліотечного комплексу; проводити діагностику і моніторинг навчально-виховного процесу; виконувати моделювання процесів управління системою; створювати алгоритми кваліметричного стандарту діяльності кожної управлінської структури [2-3, 5, 7, 8].

Курс включає лекційний цикл, лабораторний практикум, написання курсової роботи та відвідування об'єктів впровадження комп'ютерних систем та закладів освіти.

На лекційних заняттях студентам викладаються загальні питання методики професійного навчання, розкриваються цілі й задачі предмета, роль і місце інформації в суспільному розвитку; роль теорії управління, проводиться аналіз функцій управління; роль і місце комп'ютерних технологій як засобів створення автоматизованої інформаційної інфраструктури суспільства; поняття інформаційної культури як складової професійної культури інженера-педагога; кваліметричний підхід до вимірювання рівня досягнень цілі, визначаються критерії оцінки роботи навчального закладу, перспективи управління освітою. На лабораторних практикумах студенти поглиблюють вивчення науково-теоретичних основ методики професійного навчання і оволодівають сучасними методами, навичками педагогічного експерименту з використанням комп'ютерних технологій в навчальному процесі. У ході курсового проектування студенти самостійно проводять наукові дослідження, виробляють науково-практичні

положення застосування комп'ютерних технологій в управлінській діяльності підприємств і організацій та навчально-виховного закладу освіти. У процесі відвідування уроків у закладах освіти студенти спостерігають за ходом занять, потім проводять їх аналіз за планом; аналізують види і форми діяльності навчального закладу; визначають об'єкти, що підлягають комп'ютеризації.

Заняття з курсу МПН організуються таким чином, щоб забезпечити формування достатньо повних і систематичних знань у майбутнього інженера-педагога, який вміє використовувати комп'ютерні технології в управлінській діяльності. На основі цієї бази інженер-педагог зможе адаптувати свої знання в умовах конкретного підприємства, організації як об'єктів впровадження комп'ютерних систем та навчального закладу.

Вивчення курсу «Методика професійного навчання» здійснюється за модульним принципом, який має такі професійно орієнтовані програмні установки, склад і організацію навчального матеріалу, методи, форми діяльності студентів, засоби контролю і управління навчально-виховним процесом, які забезпечують адаптацію студентів до потреб професійної діяльності. Технологічність визначається критеріями: а) концептуальністю – дидактична установка, основана на особистісно-орієнтованому підході до навчання; б) системністю – цілісність, логічна послідовність, взаємозв'язок і взаємообумовленість всіх складових тематичних модулів; в) цілеспрямованістю – передбачає продуктивне засвоєння знань, розвиток творчих здібностей і рефлексію студента, формування системного-цілісного знання з тематичних розділів і курсу в цілому; г) керованістю – доступність поетапного контролю з боку викладача, коректування і можливість варіювання змісту модулів, їх складових з урахуванням виявленого рівня та індивідуальності студента; д) ефективністю – інтенсифікація і оптимізація пізнавальної діяльності, в результаті якої досягається усвідомлення і стійкість запам'ятовування отриманих знань; е) відтворенням – можливість застосування технології в інших навчальних закладах.

Кожен студент протягом навчального періоду формує на магнітному носії персональну «Робочу електронну папку», виконуючи конспектування, документування і архівування навчального матеріалу, складання схем базових знань, реєстрацію своїх

успіхів, відношення до пройдених тем і ступінь їх засвоєння, оформлення своїх пропозицій з практичного застосування отриманих знань.

Тематичне планування навчального матеріалу

№ з/п	Тема	Кількість годин		
		Всього	Лекцій	Лаборатор.
1	Модуль 1. Комп'ютерні технології в управлінні та навчанні	6	6	
2	Інформаційна освіта та інформаційна культура	2	2	
3	Концепція інформатизації освіти в Україні	2	2	
4	Концепція управління «по цілям» в системі освіти	2	2	
5	Модуль 2. Методика використання КТ в управлінській діяльності	20	10	10
6	Методи управління	2	2	
7	Методика використання КТ для забезпечення функції управління – планування	4	2	2
8	Методика використання КТ для забезпечення функції управління – організація	6	2	4
9	Методика використання КТ для забезпечення функції управління – «мотивація»	4	2	2
10	Методика використання КТ для забезпечення функції управління – контроль	4	2	2
11	Модуль 3. Методика використання КТ у навчально-виховному процесі	18	10	8
12	Методика використання КТ для підготовки уроків і лекцій	4	2	2
13	Методика використання КТ для підготовки практичних, лабораторних і семінарських занять	4	2	2
14	Методика використання КТ для організації самостійної підготовки студентів	2	2	
15	Методика використання КТ для контролю і оцінювання результатів навчання	4	2	2
16	Методика використання КТ у позакласній і позашкільній роботі	4	2	2

№ з/п	Тема	Кількість годин		
		Всього	Лекцій	Лаборатор.
18	Модуль 4. Методика використання КТ для діагностики і моніторингу навчально-виховного процесу	6	4	2
19	Методика кваліметричного підходу до вимірювання рівня досягнення мети	2	2	
20	Критерії оцінки роботи навчального закладу в кваліметричному вимірюванні	4	2	2

**Структура «Робочої електронної папки»
з дидактичної одиниці:**

Структурний елемент	Короткий опис
1. Тема і підтема дидактичної одиниці	Мета вивчення теми, її значення в системі дисципліни і професійної підготовки в цілому
2. План роботи з тематичного розділу	Перелік підрозділів теоретичних, та лабораторних занять і завдань для самостійної роботи (теми доповідей і рефератів, які рекомендовані для розробки на вибір студентами. Етапи тренінгу і контрольного тестування.
3. Функціональна роль комп'ютерних технологій	Теоретичні відомості про можливість використання КТ. Базове і прикладне програмне забезпечення при вивченні теми.
4. Електронний щоденник	Анкета студента: зміст плануючих і пройдених часових етапів пізнавально-діяльнісного процесу, реєстрація оцінок відповідної роботи
5. Робочі електронні зошити з дисципліни	Контрольні, перевірочні, самостійні аудиторні практичні роботи, виконані на комп'ютері з теми.
6. Діагностичні тести	Система діагностичних тестів з пройденої теми, реєстрація виконаної роботи.
7. Електронний глосарій	Список термінів з визначеннями і поясненнями
8. Електронна база	Блок-схема базових знань для практичного

Структурний елемент	Короткий опис
довідкових даних	використання в професійній діяльності (поняття, розрахункові формули, алгоритми, принципові схеми, графіки, діаграми, слайди).
9. Коментарі та самодіагностика	Думка студента про якість викладання, своє розуміння навчального матеріалу, практичне значення отриманих знань, придбані уміння і навички для професійної діяльності. Висновки на перспективу.
10. План презентацій електронної папки	Опис слайд-презентацій, результати роботи експертної групи з оцінки РЕП, оцінка проведеної презентації

Література

1. Глушков В.М. Основы безбумажной информатики. Изд. 2-е. – М.: Наука. Гл. ред. физ-мат. лит., 1987. – 552 с.
2. Дмитренко Г.А. Стратегічний менеджмент у системі освіти. Навч. посібник. – К.: МАУП, 1999. – 176 с.: іл.
3. Денисенко С.И. Рейтинг как комплексное средство контроля учебной деятельности студентов. // Инновации в образовании. – 2002. – №1.
4. Інформатизація освіти України: стан, проблеми, перспективи // Комп'ютер у школі і сім'ї. – 2003. – № 6.
5. Єльнікова Г.В. Використання Excel у процесі діагностики управлінської компетенції керівника ЗНЗ // Комп'ютер у школі і сім'ї. – 2003. – № 6.
6. Красильникова В.А. Інформатизація образования: понятійний апарат // Информатика и образование. – 2003. – №4.
7. Мельник П.В. Система підготовки кадрів для державної податкової служби // Комп'ютер у школі і сім'ї. – 2000. – № 3.
8. Парашенко Л.І., Леонський В.Д. Інформаційні технології в управлінні ліцеєм // Комп'ютер у школі і сім'ї. – 2003. – № 6.
9. Полат Е.С.и др. Новые педагогические технологии в системе образования: Учеб. пособие для студ. пед.вузов и системы повыш. квалиф. пед. кадров / Е.С. Полат, М.Ю. Бухаркина, М.В. Моисеева, А.Е. Петров. Под ред. Е.С. Полат. – М.: Из-

- дательский центр «Академия», 2001. – 172 с.
10. Прокопенко І.Ф. Інформаційне суспільство і освіта // Комп'ютер у школі і сім'ї. – 2003. – № 1.
 11. Прокопенко І.Ф., Биков В.Ю., Раков С.А. До питання інформатизації вищих навчальних закладів // Комп'ютер у школі і сім'ї. – 2002. – № 4.
 12. Прокопенко І.Ф., Биков В.Ю., Раков С.А. Інформатизація вищих навчальних закладів – як мрії перетворити на дійсність // Комп'ютер у школі і сім'ї. – 2003. – № 2.
 13. Ракитов А.И. Новый подход к взаимосвязи истории, информации и культуры: пример России. // Вопросы философии. – 1994. – № 4.
 14. Роберт И.В. О понятийном аппарате информатизации образования // Информатика и образование. – 2003. – №1.
 15. Суханов А.П. Информация и прогресс. – Новосибирск: Наука, 1988. – 192 с.
 16. Чмыхов А.Н. Пути решения управления вузами. // Инновации в образовании. – 2002. – №3.
 17. Шолохович В.Ф. Информационные технологии обучения. // Информатика и образование. – 1998. – № 2.

ЗМІСТ ТА ЗАВДАННЯ КУРСУ “ІНФОРМАТИКА ТА КОМП’ЮТЕРНА ТЕХНІКА” ДЛЯ СТУДЕНТІВ АГРАРНИХ ЗАКЛАДІВ ОСВІТИ III-IV РІВНІВ АКРЕДИТАЦІЇ

Л.П. Воронець, В.А. Сергієнко
м. Суми, Сумський національний аграрний університет
voronecl@mail.ru

Останнім часом інформація стає одним з важливих інструментів реалізації державної політики щодо розвитку сільського господарства, а інформаційна система – тим ланцюгом, який поєднує численні виробничі й агросервісні формування, управлінські структури та агронауку.

Кінцевою метою інформатизації АПК є створення умов для швидкого впровадження сучасних технологій у виробництво, прийняття оптимальних управлінських рішень, ефективного функціонування ринкових і фінансово-кредитних механізмів, підтримання міжгалузевих зв’язків та інтеграції АПК в світову економічну систему.

Основними завданнями інформатизації агроосвіти постають: широке використання інтранет-технологій та Web-технологій, поєднання розробки централізованих баз даних колективного користування з розподіленими по навчальних закладах документальними базами даних з можливістю віддаленого доступу, активне використання сучасних комп’ютерних технологій в навчальному процесі, широке впровадження новітніх досягнень по дистанційному навчанню, реалізація дистанційної інтерактивної консультативної служби, створення загальнодержавної комп’ютерної інформаційної мережі Агроінформосвіта [1].

Підготовка спеціалістів-аграріїв до виконання поставлених завдань відбувається в процесі вивчення курсу “Інформатика і комп’ютерна техніка”. Вивчення цього курсу в аграрних закладах освіти III-IV рівнів акредитації повинно сприяти виконанню цілого комплексу органічно взаємопов’язаних завдань, які у загальнонавчальному та професійному аспектах умовно можна об’єднати у групи:

1) завдання, що передбачають формування у студентів загально навчальних знань, умінь, навичок з науки інформатика,

комп'ютерної техніки та їх використання для підвищення ефективності вивчення інших навчальних дисциплін.

2) завдання, що мають суто спеціальне призначення і передбачають вивчення комплексу спеціальних її підрозділів для формування знань, практичних умінь та навичок застосування можливостей науки інформатики, комп'ютерної техніки та відповідного програмного забезпечення для вирішення суто фахових завдань з обраної професійної діяльності.

Комп'ютерна підготовка студентів аграрних закладів освіти починається з першого курсу і, залежно від спеціалізації, продовжується 3-5 років (див. табл. 1). Спочатку для всіх спеціальностей викладається предмет “Інформатика і комп'ютерна техніка”, який є початковим етапом безперервної комп'ютерної підготовки студентів. Головна мета вивчення курсу “Інформатика і комп'ютерна техніка” на 1-2 курсах – “формування у майбутніх менеджерів сучасного рівня інформаційної та комп'ютерної культури, набуття практичних навичок роботи на сучасній комп'ютерній техніці і використання сучасних інформаційних технологій для розв'язання різноманітних задач в практичній діяльності за фахом” [2, 3]. Тобто навчання спрямоване на формування інформаційної культури використання ПК, розуміння професійної сутності задач, ролі та місця ПК в професійній діяльності. Невипадково більша частина навчальних годин відводиться практичним і лабораторним роботам. Це пов'язано із специфікою самого навчального предмету, для якого комп'ютер є основним предметом, об'єктом і, одночасно, засобом його вивчення.

У підготовці спеціалістів з вищою освітою, зокрема аграріїв, важливим елементом є формування прогностичних вмінь управління економічними процесами з орієнтацією на чітко визначений кінцевий результат. При цьому до спеціально-професійних прогностичних умінь відносяться: висування цілей та завдань; вибір способів їх досягнення; передбачення результату можливих відхилень та небажаних явищ; виявлення етапів, динаміки досліджуваного процесу, розподіл часу, планування розв'язання поставлених завдань тощо. Тому науковою основою прогнозування є знання сутності, логіки та закономірностей досліджуваного процесу. Це обумовлює введення в курс “Інформатика і

комп'ютерна техніка” навчальних предметів типу “Математичне програмування”, “Економетрія”, “Дослідження операцій”, “Інформаційні технології”, “Інформаційні системи в міжнародному менеджменті”, які сприяють формуванню також і проектних вмінь.

Таблиця 1

Структура безперервної комп'ютерної підготовки
студентів Сумського національного аграрного університету з
курсу “Інформатика і КТ”

Факультет	1 курс	2 курс	3 курс	4 курс	5 курс	
Економічний	“Інформатика та КТ” “Математичне програмування”	“Економетрія” “Дослідження операцій” “Комп'ютерні мережі та телекомунікації”	“ІТ в міжнародному менеджменті” – <i>для спец. МЗД</i>	“Інформаційні системи в менеджменті” – <i>для ЗЕЗ, ОПЗ</i>	<i>для магістрів:</i> “АРМ менеджера” “Основи теорії досліджень” <i>предмети професійної спрямованості:</i>	
					<i>спец. ІСМ</i>	“Проектування баз даних” “Електронна комерція” “Системи підтримки прийняття рішень” “Моделювання економіки” “Безпека інформаційних систем” “Ефективність інформаційних систем”
					<i>спец. АМ</i>	“Автоматизована система обробки інформації”

Факультет	1 курс	2 курс	3 курс	4 курс	5 курс	
						<i>спец. ЗЕЗ, ОПЗ, МЗД</i>
Обліково-фінансовий	“Інформатика та КТ”	“Інформатика та КТ”	“Математичне програмування”	“Економетрія” “Інформаційні системи і технології”	“Організація інформаційних систем”	
Інженерно-технологічний	Механізація	“Основи комп’ютерної техніки”	“Прикладна математика”	-----	“Ергономіка”	<i>для магістрів:</i> “Інформаційні технології” “Інформаційні системи”
	Технологія переробки молока та м’яса	“Комп’ютерна техніка та програмування”	“Комп’ютерна техніка та програмування”	“Мат. моделі в розрахунках на ЕОМ”	-----	“Методи та засоби комп’ютерної обробки інформації”

Факультет	1 курс	2 курс	3 курс	4 курс	5 курс
<i>Зооінженерний</i>	-----	“Обчислювальна техніка”	-----	-----	“Моделювання технологічних процесів”
<i>Агрономічний</i>	“Інформатика та КТ”	-----	-----	-----	-----
<i>Ветеринарний</i>	“Основи інформатики і моделювання”	-----	-----	-----	<i>для магістрів:</i> “Інформаційні системи”

Отже, проведений аналіз змісту курсу “Інформатика і комп’ютерна техніка” дає підстави вважати:

1. Зміст курсу “Інформатика і комп’ютерна техніка” в основному складається з двох органічно взаємопов’язаних частин. Перша частина пов’язана з виконанням групи завдань, спрямованих на формування загальнонавчальної системи знань, умінь, навичок з науки інформатика, користування КТ з відповідним ПЗ. Друга частина курсу спрямована на виконання групи професійно орієнтованих завдань, пов’язаних, у першу чергу, із удосконаленням, розширенням, поглибленням системи знань, умінь, навичок з науки інформатики, професійної підготовки випускника-спеціаліста, з готовністю застосування КТ і відповідного ПЗ у майбутній професійній діяльності.
2. Вивчення курсу “Інформатика і комп’ютерна техніка” передбачається здійснювати послідовно протягом всього терміну навчання (5 років) з поступовим розширенням та поглибленням системи знань, умінь, навичок з даного курсу.

Література:

1. Програма інформатизації агропромислового комплексу на 2000-2005 роки і на період до 2010 року (проект). Редакція 1, Київ, 1999. – 54 с.
2. Галузеві стандарти вищої освіти. Освітньо-кваліфікаційні характеристики та освітньо-професійні програми підготовки бакалаврів та спеціалістів. – ВЦ КНТЕУ, 2001.
3. Галузеві стандарти вищої освіти. Освітньо-кваліфікаційні характеристики та освітньо-професійні програми підготовки магістрів. – ВЦ КНТЕУ, 2001.

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ НАВЧАЛЬНОГО ПРИЗНАЧЕННЯ З ЕКОЛОГІЇ ЛЮДИНИ

Ю.О. Дорошенко¹, Н.В. Семенюк²

¹ м. Київ, Інститут педагогіки АПН України

² м. Хмельницький, Хмельницький державний університет
dua159@ukrpost.net

Постановка проблеми. Існуючі суперечності між потребами суспільства у кадрах і наявним рівнем їх підготовки у закладах освіти свідчать про необхідність якісних змін, радикального перегляду всієї методичної системи підготовки фахівців у відповідності з сучасними потребами держави, суспільства й кожної окремої особистості. Зазначене вимагає, зокрема, конкретизацію цілей, оновлення і “опракличення” змісту та осучаснення організаційних форм, методів і засобів підготовки фахівців з вищою освітою. Одним із шляхів реалізації зазначеного є застосування універсального та створення спеціалізованого програмного забезпечення навчального призначення та впровадження відповідних педагогічних технологій у навчальний процес вищої школи, що дозволить підвищити ефективність навчання та сприятиме індивідуалізації, диференціації і активізації навчально-пізнавальної діяльності студентів.

Одним із навчальних курсів підготовки майбутніх інженерів-екологів є “Екологія людини”, що вивчається на другому чи третьому році навчання у ВНЗ відповідного профілю. Зазначений навчальний курс реалізовано за авторською програмою (54 аудиторних години, з яких 18 годин займає цикл лекцій та 36 годин відводиться під лабораторно-практичні заняття) й впроваджено у навчальний процес Хмельницького державного університету на умовах експериментальної апробації.

Аналіз останніх досліджень. Загальні підходи до створення електронних засобів навчання різноманітного функціонального призначення (як складників програмно-методичних комплектів) висвітлено у [2, 3, 5–7, 13]. Формуванню систем навчання біолого-екологічного спрямування, створенню сучасних комп’ютерних засобів навчання, методики їх застосування та конструюванню й впровадженню відповідних педагогічних тех-

нологій присвячено ряд дисертацій, зокрема, [12, 14, 15]. Узагальнено склад авторського програмно-методичного комплексу (ПМК) з екології людини показано у [11], окремі його складові – у [4, 8, 9].

Формулювання цілей статті (постановка завдання). Завданням цієї статті є узагальнена презентація комп'ютерної підтримки (програмних засобів навчального призначення) курсу “Екологія людини”, що увійшли до складу відповідного ПМК.

Основна частина. Зміст і структурна побудова дисципліни “Екологія людини” та концептуальні основи і технологічні особливості організації й проведення навчального процесу з означеного курсу представлені у [11]. Нині закінчується розробка його навчально-методичного забезпечення, зокрема, програмного забезпечення навчального призначення (ПЗНП).

За функціональною ознакою виділено такі узагальнені напрямки розробки і впровадження електронних засобів навчання:

1) у навчальну діяльність студентів під керівництвом викладача (лекції, колоквиуми, практичні заняття, лабораторні роботи тощо);

2) у дистанційне навчання;

3) у самостійну освітню діяльність студентів під час аудиторних занять чи позааудиторної роботи.

Визначено такий склад електронних засобів навчання: навчальний посібник “Екологія людини” у гіпертекстовому та PDF-форматі, що є змістовою копією друкованого посібника [10]; конспект лекцій; методичний посібник для викладача; спеціалізоване програмне забезпечення для проведення лабораторного практикуму; комп'ютерна система педагогічного контролю; база тестів; тлумачно-термінологічний словник; дистанційний курс.

Впровадження електронних засобів навчання зумовлюється сучасною об'єктивною тенденцією до збільшення частки індивідуальної роботи студента у навчальних програмах вузів, розвитком і поширенням дистанційної освіти і необхідністю створення відповідних навчально-методичних та інформаційних ресурсів. При цьому, створюються умови для реалізації особистісно зорієнтованого навчання, активізується навчально-пізнавальна діяльність студентів, відбувається індивідуалізація навчання із надан-

ням студентам більшої самостійності щодо його конкретної організації.

Основною функціональною складовою навчального курсу визначено лабораторний практикум. Для його проведення створено спеціалізований програмний комплекс, який системно поєднує усі компоненти комп'ютерної підтримки курсу. Програмний комплекс структурно складається з керуючого модуля й окремих програмних одиниць.

У процесі визначення структурної побудови та конструювання змісту лабораторного практикуму провідним дидактичним принципом обрано *моделювання професійної діяльності інженера-еколога*. Системним доповненням до зазначеного принципу (разом із загальновідомими) стали також принципи *професійної мобільності, поєднання навчання і виробничої праці, комп'ютеризації і інформатизації навчального процесу*.

Лабораторний практикум має комплексний характер і структурно складається з практичних і лабораторних робіт, заняття-колективуму, творчого колективного проекту та індивідуально-групового захисту виконаних робіт та проекту.

Визначальною ознакою пропонованого лабораторного практикуму є обов'язкове активне використання студентами різноманітного інструментарію сучасних інформаційних технологій та останніх досягнень з екології, психології, медицини, гігієни тощо. Під час виконання кожного складника лабораторного практикуму застосовується спеціалізоване програмне, дидактичне і методичне забезпечення та відпрацьовуються певні комп'ютерно-інформаційні технології.

Приклад комп'ютерної реалізації психологічного тестування, що входить до однієї з лабораторних робіт, наведено на рис. 1.

На рис. 2 продемонстровано карту місцевості, яка використовується під час виконання творчих групових проектів та визначення і нанесення меж районів з певним забрудненням.

Моніторинг навчального процесу потребує постійного (чи періодичного) визначення рівня навчальних досягнень студентів. Тому однією з актуальних задач організації сучасного навчального процесу є реалізація автоматизованого (комп'ютерно-орієнтованого) контролю знань із використанням останніх дося-

гнень теорії тестування та новітніх інформаційних технологій.

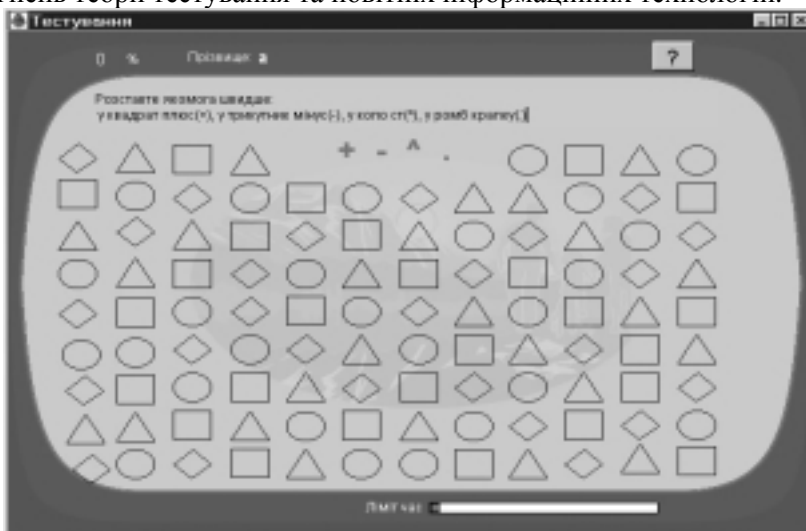


Рис. 1.

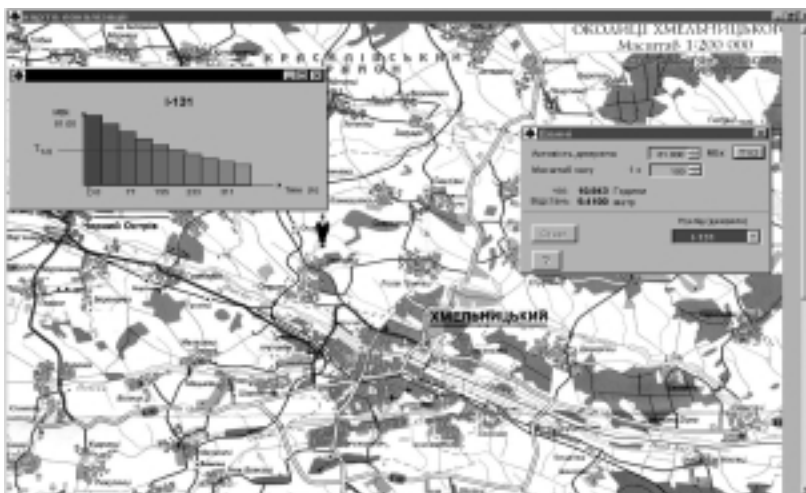


Рис. 2.

Однією з функцій програмного комплексу є реалізація комп'ютерного тестування у вигляді вбудовано-автономної програмно-інформаційної підсистеми у складі:

- керуючої програми-оболонки (ядра підсистеми);
- структурованого розгалуженого довідника;
- банка даних тестових питань;
- програми-генератора тестових завдань;
- бази даних результатів тестування (архіву);
- програми автоматизованого опрацювання архіву тестування (підготовка звітів) за заданими критеріями.

Програмно-інформаційне забезпечення, що призначено для реалізації комп'ютерного тестування з навчального курсу "Екологія людини", створено за принципами цілісної відкритої системи. На нашу думку, комп'ютерна програма не повинна одразу за підсумками тестування виставляти конкретну оцінку студентові, а лише фіксувати правильні і неправильні відповіді – загальну їх кількість, що тотожне кількості балів при використанні однорівневих за складністю тестових питань чи стане основою для підрахунку підсумкової кількості балів при різнорівневих (різнонагових) тестових питаннях.

Зважаючи на критерії, які визначають рівень об'єктивності проведеного контролю, багатьма дослідниками встановлено, що однією з основних дидактичних вимог щодо ефективного використання ЕОМ для перевірки знань з урахуванням обсягу, повноти, узагальненості, цілеспрямованості й дієвості є оптимальний рівень складності завдань, вправ і питань, які застосовуються для контролю. Тому за найкраще прийнято застосовувати тестові питання однакового рівня складності, що й було реалізовано під час формування бази тестових питань.

Загалом, результати тестування окремого студента чи цілої групи слід розглядати перш за все як оперативну інформацію для викладача про хід навчально-виховного процесу та поточний рівень його ефективності (визначення кількісних і якісних показників його результативності). За наслідками узагальнення й аналізу цієї інформації викладач має внести відповідні корективи до своєї роботи.

Подальшим розвитком пропонованого програмного забезпечення бачиться розробка повнофункціональної підсистеми діагностування якості й ефективності навчального процесу.

Іншим концептуальним напрямком впровадження інформаційно-комунікаційних технологій у навчальний процес з екології

людини є широке, різнопланове застосування найбільш поширених на цей час програмних засобів, зокрема, програми MS Excel. Нами реалізовано три можливих напрямки застосування програми MS Excel. Вибір означеної програми зумовлений її поширеністю та доступністю використання.

1. Опрацювання результатів тестового контролю рівня навчальних досягнень учнів.
2. Опрацювання результатів педагогічного експерименту.
3. Створення психологічних тестів.

Нами для застосування у лабораторному практикумі з екології людини розроблено кілька програмних тестів, зокрема, “Тест для виявлення темпераменту особистості (за Г. Айзенком)” та низки тестів “Визначення біоритмічного типу працездатності і критичних днів людини”.

На рис. 3 показано робоче поле тесту Ганса Айзенка, рис. 4 демонструє узагальнені результати тестування, на рис. 5 проілюстровано “секторне” подання характеристик темпераментів людини, покладене в основу якісного опрацювання числових результатів тестування. Окрім того, комірки стовпця “Тип темпераменту людини” зв’язані гіперпосиланнями з текстовими файлами, де розміщено розгорнуту інформацію про кожну характеристику й темперамент.

ТЕСТ Ганса АЙЗЕНКА			
№№ пит	ЗАПИТАННЯ		Відповідь 1-мак 0-ні
1	Чи часто Ви прагнете нових вражень, для того щоб розважитися, пережити сильні відчуття?		0
2	Чи часто Ви відчуваєте потребу в друзях, які можуть Вас зрозуміти, підбадьорити, послухувати?		0
3	Чи вважаєте Ви себе безтурботною особистістю?		0
4	Чи вважали Ви когось “ні” (тобто відмовили людині)?		1
5	Ви обмислюєте свої справи не постачаючи і вважаєте за краще почекати, ніж діяти?		1
6	Чи завжди Ви дотримуєтеся своїх обіцянок, навіть якщо це Вам не вигідно?		0
7	Чи часто у Вас бувають сади і піднесення настрою?		1
8	Як звичайно Ви читаете і говорите – швидко, без привального обмислювання?		0
9	Чи бувають випадки, коли у Вас виникають почуття, що Ви неадекватні без певної на це причини?		1
10	Чи люблять Б Ви об заклад майже на все?		0

Рис. 3.

Результати тестування		
Аналіз результатів тестування	ДАНІ ДОСТОВІРНІ	
Темперамент людини	стриманий	МЕЛАНХОЛІК
Показник екстраверсії	2	
Показник невротизму	18	
Індекс відвертості (%)	100	

Рис. 4.

№п/п	Кути сектора		Тип темпераменту людини	
	початковий	кінцевий	Характеристика	Темперамент
1	90	101,25	смилий до розпачу	МЕЛАНХОЛІК
2	101,25	112,5	тристоронній	
3	112,5	123,75	ригідний	
4	123,75	135	вразливий	
5	135	146,25	песимістичний	
6	146,25	157,5	стриманий	
7	157,5	168,75	відлюдкуватий	
8	168,75	180	тихий	
9	180	191,25	пасивний	
10	191,25	202,5	обачливий	ФОЛІГМАТИК
11	202,5	213,75	розсудливий	
12	213,75	225	міролюбивий	
13	225	236,25	контрольований	
14	236,25	247,5	надійний	
15	247,5	258,75	урівноважений	
16	258,75	270	спокійний	
17	270	281,25	смилий до лідерства	САМОВІК
18	281,25	292,5	безтурботний	
19	292,5	303,75	запорожаний	
20	303,75	315	невимушений	
21	315	326,25	чужий	
22	326,25	337,5	балакучий	
23	337,5	348,75	контактний	
24	348,75	0	товариський	ХОЛЕРИК
25	0	11,25	активний	
26	11,25	22,5	оптимістичний	
27	22,5	33,75	імпульсивний	
28	33,75	45	напестийний	
29	45	56,25	збудливий	
30	56,25	67,5	агресивний	
31	67,5	78,75	тристоронній	
32	78,75	90	чутливий	

Рис. 5.

До складу розглядуваного ПМК включено *тлумачно-термінологічний словник (глоссарій)* у двох варіантах подання – друкованому та електронному. Потреба у такому словнику зумовлюється тим, що екологія людини являє собою міждисциплінарний навчальний курс, насичений великою кількістю термінів з різних наук. Успішне опанування цього курсу вимагає знання й однозначне розуміння сутності спеціальної термінології усіма учасниками навчально-виховного процесу.

Електронний словник, який має трирівневу ієрархічну структуру (рис. 6), створено як гіпертекстовий документ засобами програми PowerPoint. Усі складові файли електронного словника записано у форматі “pps”, що забезпечує безпосередню активацію словника без завантаження середовища розробки. Використання програми PowerPoint зумовлено широкими можливостями цього інструментального засобу щодо оформлення інформаційних фрагментів, у яких поєднується мультимедійна (текстова, графічна, звукова, анімаційна) інформація.



Рис. 6.

Подальшим розвитком та урізноманітненням дидактичного забезпечення курсу “Екологія людини” з урахуванням сучасних тенденцій в освіті стало створення інформаційного забезпечення для дистанційного навчання. Першими складниками системи ДН з екології людини стали електронний варіант лекційного курсу та тестуюча система у гіпертекстовому форматі. Зазначене навчально-методичне забезпечення розміщено на сайті за адресою <http://dn.tup.km.ua/>.

Окремого розгляду потребує проблема конструювання (мо-

делювання) навчального процесу з певної дисципліни, особливо, за умови застосування електронних засобів і інформаційних технологій навчання. Провідна роль при цьому має відводитися педагогічному контролю.

До основних складових педагогічного контролю результативності навчального процесу належать моніторинг і діагностика. Сутнісний аналіз термінів “моніторинг” і “діагностика” дозволяє стверджувати, що педагогічна діагностика ґрунтується на матеріалах моніторингу навчально-виховного процесу, виступає інструментальним чи технологічним засобом вироблення корегуючих дій, спрямованих на підвищення ефективності та якості навчання.

Підсумовуючи матеріал цієї статті, вважаємо за доцільне запропонувати структурно-функціональну схему конструювання навчального процесу з екології людини (рис. 7). Ця схема (як модель) є результатом попередніх досліджень й розроблена з позицій технологічного підходу до навчання, використання моніторингу і діагностики навчального процесу для вироблення корегуючих дій (впливів) й широкого впровадження інформаційних технологій.

Визначальною особливістю запропонованої схеми (моделі) є вироблення за результатами діагностики й моніторингу навчального процесу корегуючих впливів та їх спрямування на різні складові методичної системи курсу з метою підвищення його ефективності та якості.

Висновки. Одним із перспективних напрямків розвитку навчально-методичного забезпечення є створення електронних складових програмно-методичних комплектів – окремих програм і комплексів. Авторами статті створено, апробовано та впроваджується на умовах педагогічного експерименту оригінальний програмно-методичний комплекс комп’ютерної підтримки курсу “Екологія людини”. Цей комплекс являє собою спеціалізований пакет навчальних програм, який створений засобами мови програмування Visual FoxPro та інструментальними засобами MS Office і який орієнтовано на роботу у середовищі Windows 9x, 2000, XP. Інтерфейс програмних засобів є інтуїтивно зрозумілим й розрахований на мінімально підготовленого (в галузі комп’ютерних технологій) користувача. У разі появи певних ускладнень, користувач може звернутись до вбудованої допомоги.

Отже, подальшим розвитком й вдосконаленням навчальної

книжки мають стати програмно-методичні комплекти різноманітного наповнення (складу) й багатофункціонального призначення. Обов'язковою функціонально складовою таких ПМК є програмне забезпечення навчального призначення, орієнтоване на автономне використання на окремому комп'ютері чи на віддалений доступ у режимі дистанційної освіти.

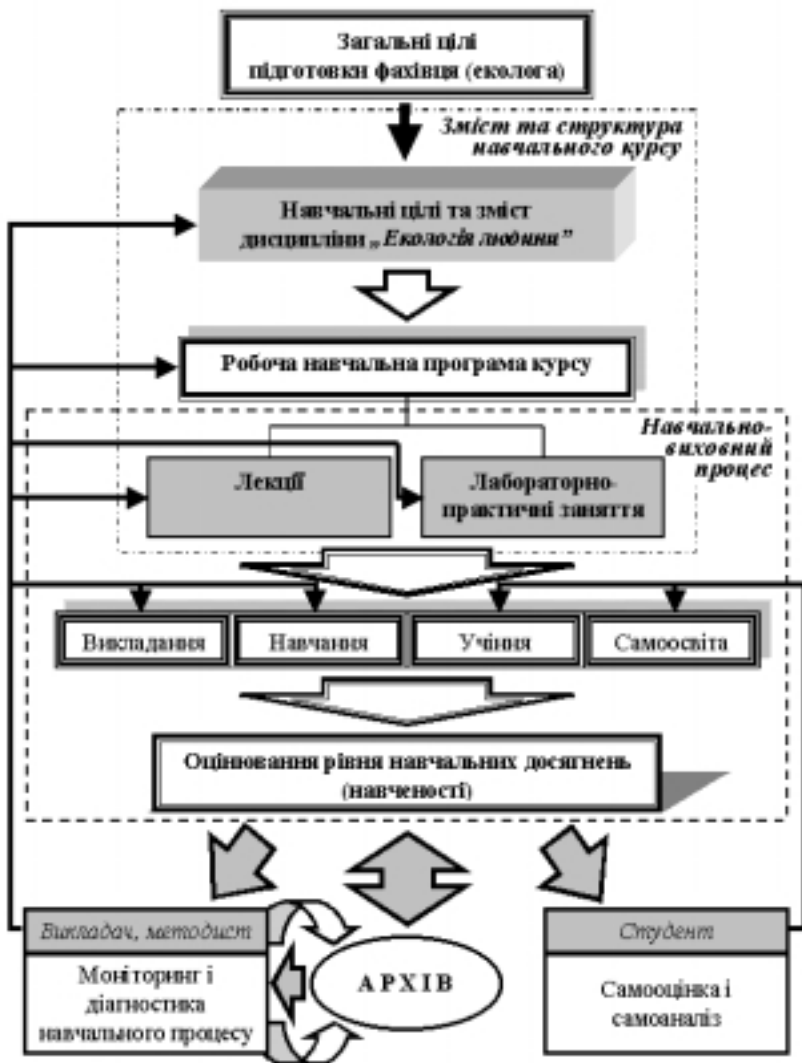


Рис. 7.

Література

1. Гуменюк О.Б., Семенюк Н.В. Соціальна екологія і екологія людини: Програма курсу, метод. вказівки та контрольні завдання для студентів заочної форми навчання спеціальності "Екологія та охоро- на навколишнього середовища". – Хмельницький: ТУП, 2001. – 50 с.
2. Гуржій А.М., Бурда М.І., Волинський В.П. Навчальний підручник для 12-річної загальноосвітньої школи // Освіта України. – №22 (250), 30 травня 2001 р.– С. 10.
3. Дорошенко Ю.О. Інформатизація – пріоритетний напрям реформування освітньої галузі // Педагогічна газета. – №3 (57), березень 1999 р. – С. 4–5.
4. Дорошенко Ю.О., Семенюк Н.В. Застосування професійно-орієнтованих завдань та інформаційних технологій у лабораторно-му практикумі з екології людини // Комп'ютери в навчальному процесі: Матеріали 2-ої Всеукраїнської науково-практичної конфе- ренції 29–30 жовтня 2002 р. – Умань: Алмі, 2002. – С. 20–22.
5. Інформатизація середньої освіти: програмні засоби, техно- логії, досвід, перспективи/ Н.В. Вовковінська, Ю.О. Дорошенко, Л.М. Забродська та ін. За ред. В.М. Мадзігона та Ю.О. Дорошенка. – К.: Педагогічна думка, 2003. – 276 с.
6. Основи нових інформаційних технологій навчання: Посіб- ник для вчителів / Авт. кол.; За ред. Ю.І. Машбиця/ Інститут психо- логії ім. Г.С. Костюка АПН України. – К.: ІЗМН, 1997. – 262 с.
7. Роберт И.В. Современные информационные технологии в образовании: дидактические проблемы, перспективы использова- ния. – М.: Школа-Пресс, 1994 – 205с.
8. Семенюк Н.В. Використання персонального комп'ютера при проведенні лабораторного практикуму з курсу "Екологія лю- дини" // Зміст і технології шкільної освіти: Матеріали звітної нау- кової конференції Інституту педагогіки АПН України 26–28 берез- ня 2002 р. – К.: Пед. думка, 2002. – С. 37–38.
9. Семенюк Н.В. Електронний словник термінів з екології людини // Зміст і технології шкільної освіти: Матеріали звітної нау- кової конференції Інституту педагогіки АПН України 1–2 квітня 2003 р. – Ч. II. – К.: Пед. думка, 2003. – С. 46–48.
10. Семенюк Н.В. Екологія людини: Навчальний посібник. – Хмельницький: ТУП, 2002. – 171 с.
11. Семенюк Н.В., Дорошенко Ю.О. Програмно-методичний

комплект з екології людини // Проблеми сучасного підручника. – К.: Педагогічна думка, 2003. – Вип. 3. – С. 76–90.

12. Смирнов В.А. Научно-методические основы формирования системы обучения биологии в открытом информационном обществе: Дисс. ... докт. пед. наук: 13.00.02. – СПб., 2000. – 292 с.

13. Співаковський О.В. Про вплив інформаційних технологій на технологію освіти // Інформатика. – 2001. – №38 (134). – С. 2–3.

14. Суматохин С.В. Комплексное использование современных технических средств при изучении биологии: Дисс. ... канд. пед. наук: 13.00.02. – М., 1996. – 165с.

15. Филиппов Е.А. Методика использования средств мультимедиа в обучении общей биологии: Дисс. ... канд. пед. наук: 13.00.02. – СПб., 2001. – 149с.

СПЕКТРАЛЬНИЙ АНАЛІЗ МАЙЖЕ ПЕРІОДИЧНИХ СИГНАЛІВ

О.М. Дреєв¹, З.Ю. Філер²

¹ м. Кіровоград, Кіровоградський державний технічний
університет

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

Вступ. Пропонується метод гармонічного аналізу складних майже періодичних сигналів на основі послідовного виділення частот складових коливань з найменшим середньоквадратичним відхиленням методом найменших квадратів. Розроблений алгоритм тестовано на функціях з відомим амплітудно-частотним спектром. Програма застосована для аналізу різних індексів сонячної активності, характеристик погоди та ін. Будуються регресійні рівняння, які дозволяють прогнозувати подальший хід складних випадкових стаціонарних процесів. Для них фази прогнозуються надійно.

1. Апроксимація часових рядів тригонометричними многочленами

1.1. Задача. Маємо табличні значення невідомої в аналітичному вигляді функції на довільній сітці. Потрібно визначити частоти, амплітуди та фази основних складових гармонічних коливань, з яких складається шукана функція.

Дискретне перетворення Фур'є бути застосоване не може, бо вхідна таблична функція, в загальному випадку, має некратні (та, навіть, несумірні) гармоніки і період у такої функції не існує (*він є нескінченно великою величиною*). Ускладнює аналіз також можливе нерівномірне розбиття проміжку, на якому задана функція.

1.2. Результати. В статті висвітлюється алгоритм уточнювання значень найпотужніших гармонік з послідовним відніманням їх від основної таблиці даних значень, доки не дістанемо задовільного середнього квадратичного відхилення від нуля, чи задовільного максимального відхилення. При відомих основних частотах гармонік, процес значно пришвидшується, бо відпадає необхідність пошуку цих частот.

Алгоритми пошуку та визначення складових коливань реалі-

зовані у програмних модулях, написаних мовою C++.

1.3. Опис алгоритму покрокового уточнення

1.3.1. Пошук амплітуди за частотою методом найменших квадратів. Нехай маємо таблично задану функцію $y_i=y(t_i)$, де $i=0, 1, \dots, n$ (тут $n+1$ є кількість точок у розбитті). Необхідно підібрати a_0 та a_1 у функції $g(t)=a_0 \cdot \sin(\omega t)+a_1 \cdot \cos(\omega t)$ таким чином, щоб середнє квадратичне відхилення

$$S(\omega) = \frac{1}{n+1} \sum_{i=0}^n (y(t_i) - g(t_i))^2$$

приймало мінімальне значення.

Через те, що в точці екстремуму значення частинних похідних по коефіцієнтах a_0 та a_1 повинні бути рівні нулю, отримуємо систему рівнянь для пошуку значень цих коефіцієнтів:

$$\begin{cases} a_0 \cdot \overline{\sin^2(\omega t)} + a_1 \cdot \overline{\cos(\omega t) \sin(\omega t)} = \overline{y(t) \sin(\omega t)} \\ a_0 \cdot \overline{\sin(\omega t) \cdot \cos(\omega t)} + a_1 \cdot \overline{\cos^2(\omega t)} = \overline{y(t) \cos(\omega t)} \end{cases}$$

Тут рискою позначено середнє арифметичне значення виразу $\overline{f(t)} = \frac{1}{n+1} \sum_{i=0}^n f(t_i)$. При рівномірному розбитті, коли t_i виражається за законом $t_i=t_0+h \cdot i$, для більш швидкого знаходження сум в системі рівнянь можна скористатися загальними формулами:

$$\begin{aligned} \frac{1}{n+1} \sum_{i=0}^n \sin^2(\omega t_i) &= \frac{1}{2} - \frac{\cos(2\omega t_n) - \cos(2\omega t_{n+1}) - \cos(2\omega t_{-1}) + \cos(2\omega t_0)}{4(n+1)(1 - \cos(2\omega h))}, \\ \frac{1}{n+1} \sum_{i=0}^n \sin(\omega t_i) \cos(\omega t_i) &= \frac{\cos(2\omega t_{-1/2}) - \cos(2\omega t_{n+1/2})}{4(n+1) \sin(\omega h)}, \\ \frac{1}{n+1} \sum_{i=0}^n \cos^2(\omega t_i) &= 1 - \frac{1}{n+1} \sum_{i=0}^n \sin^2(\omega t_i). \end{aligned}$$

Для діапазону значень аргументу t , кратного періоду, середні значення квадратів синуса та косинуса дорівнюють $1/2$, а добутку синуса та косинуса – 0. Це дає простіші рівняння, розв'язані відносно шуканих коефіцієнтів:

$$a_0=2f \sin(\omega t)_c, \quad a_1=2f \cos(\omega t)_c.$$

Індекс c означає середнє значення вказаної функції.

1.3.2. Пошук найпотужнішої частоти. Він проводиться за таким алгоритмом:

Проміжок частот, на якому шукаються значення гармонік, розбивається рівномірною логарифмічною сіткою (тобто $w_i = \sqrt{w_{i+1} \cdot w_{i-1}}$, де w – значення циклічної частоти коливань), у вузлах якої методом найменших квадратів шукаються їх амплітуди. Якщо проміжок частот не обмежений за міркуваннями фізичного змісту задачі, то обмеження можна ввести з огляду на кількість початкових даних у таблиці та частоти дискретизації. Найвища частота повинна давати хоча б декілька точок на періоді, а найменша – давати хоча б півперіод на даному часовому проміжку.

Вибирається найпотужніша w_2 та запам'ятовуються значення амплітуд сусідніх частот w_1 та w_3 . Після цього починається процес уточнення частоти методом половинного ділення.

Для цього рахується амплітуда коливань для частот $w_{1,2} = \sqrt{w_1 w_2}$, $w_{2,3} = \sqrt{w_2 w_3}$. З п'яти отриманих частот вибираємо частоту з найбільшою амплітудою. Приймаємо її за нове значення w_2 , а дві сусідніх частоти позначаємо відповідно w_1 та w_3 (рис. 1). Повторюємо уточнення частоти, доки різниця w_1 та w_3 не стане меншою наперед заданого числа.

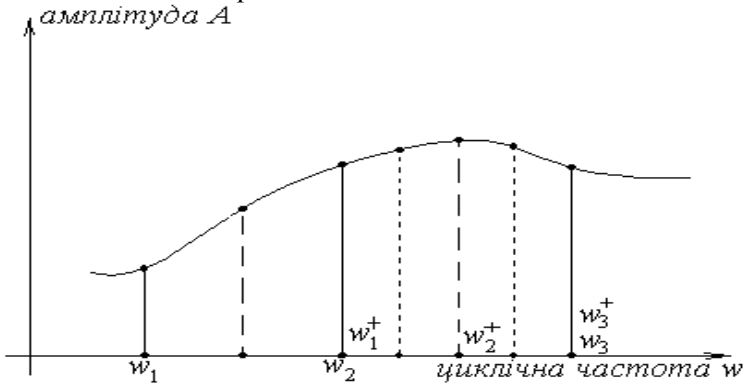


Рис. 1. Пошук значення частоти максимальної потужності

Знайшовши частоту, при якій досягається максимальне наближення $g(t_i) = a_0 \cdot \sin(wt_i) + a_1 \cdot \cos(wt_i)$ до заданої функції, побудуємо нову таблицю:

$$y_i^* = f(t_i) - g(t_i), \quad i=0, 1, \dots, n.$$

Процес пошуку частот з найбільшими амплітудами можна

повторювати доти, доки не буде досягнута достатня точність наближення. В результаті отримаємо представлення у вигляді

$$y(t_i) \approx \sum_{k=0}^m (a_{0,k} \sin(w_k t_i) + a_{1,k} \cos(w_k t_i)),$$

де m – кількість знайдених гармонік, k – номер гармоніки, $i = 0, 1, \dots, n, n+1$ – номери точок в розбитті проміжку.

2. Представлення результатів спектрального аналізу

Отримана спектрограма зображує частоти та відповідні амплітуди $A_k = \sqrt{a_0^2 + a_1^2}$. Візуальна інформація доповнюється кількісною – величина частоти, амплітуди та значення мінімальної суми квадратів по відношенню до дисперсії значень функції. Вона відіграє роль $1-K$, де K – коефіцієнт кореляції між табличними даними та апроксимуючими значеннями функції. Одна з спектрограм зображена на рис. 2.

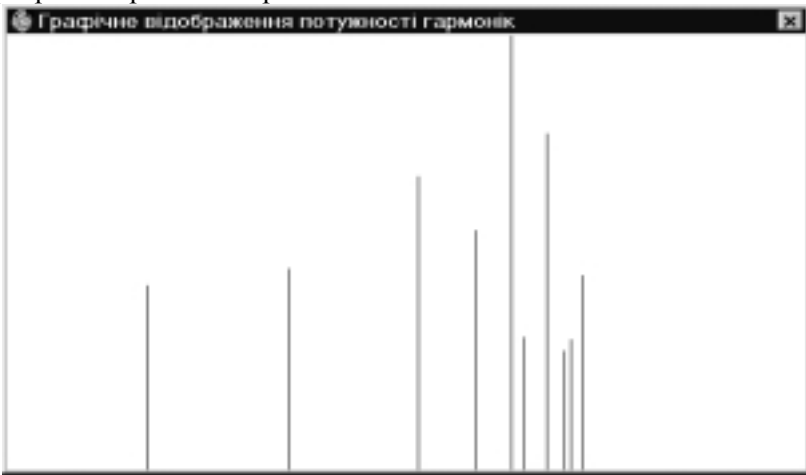


Рис. 2.

При бажанні можна побудувати графіки даної та апроксимуючої функцій, які дають можливість візуально оцінити степінь наближення (рис. 3).

3. Приклади спектрального аналізу числових рядів

3.1. Середньорічні магнітні числа Вольфа. З 1749 р. ведуться регулярні щоденні спостереження сонячної активності. На основі аналізу результатів спостережень знаходяться світові

середньо добові, місячні та річні значення чисел Вольфа та інших індексів сонячної активності. У ХІХ ст. стало зрозуміло, що природа сонячних плям має магнітну основу і орієнтація магнітних полів плям у кожній півкулі Сонця змінюється в середньому через кожні 11 років (після мінімуму кількості плям). На цій основі нами отримані масиви “магнітних” чисел Вольфа. Після кожного проходження мінімуму ми змінювали знак чисел Вольфа W . На рис. 4 зображені ці числа та їх апроксимація тригонометричними поліномами за 302 роки. На основі отриманого найкращого апроксимуючого многочлена побудовано прогноз на наступні 300 років. Він показує можливість високого рівня сонячної активності (СА) у 2240 та 2252 роках. Мабуть, до того часу людство знайде відповідні засоби профілактики цих загроз.

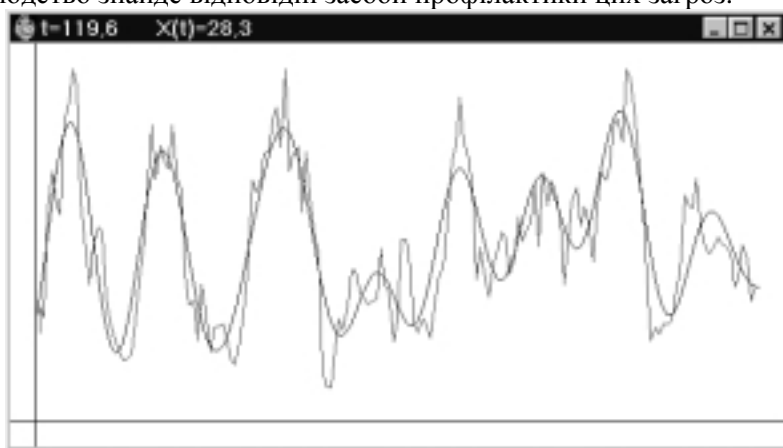


Рис. 3. Апроксимація даних сонячної активності першої половини 2003 р.

3.2. Числа Вольфа за щоденними даними НАСА та їх прогноз на 2004 р. Автори протягом майже 5 років отримують із сайту <http://www.sunspotcycle.com> НАСА значення W . За їх допомогою побудовано апроксимуючий тригонометричний поліном. Основна частота коливань має період 26 діб, що близько до періоду власного обертання Сонця та обертання Місяця навколо Землі. Значення W несуть велику випадкову складову завдяки турбулентним рухам у верхніх шарах Сонця. Тому очікувати надійної екстраполяції чисел Вольфа на довгий термін за допомо-

гою апроксимуючого многочлену не варто. Але фази коливань W цей многочлен передає непогано. На його основі нами побудовано прогноз СА на 2004 р. (Рис. 5).

Среднегодовые магнитные числа Вольфа и их прогноз на 300 лет

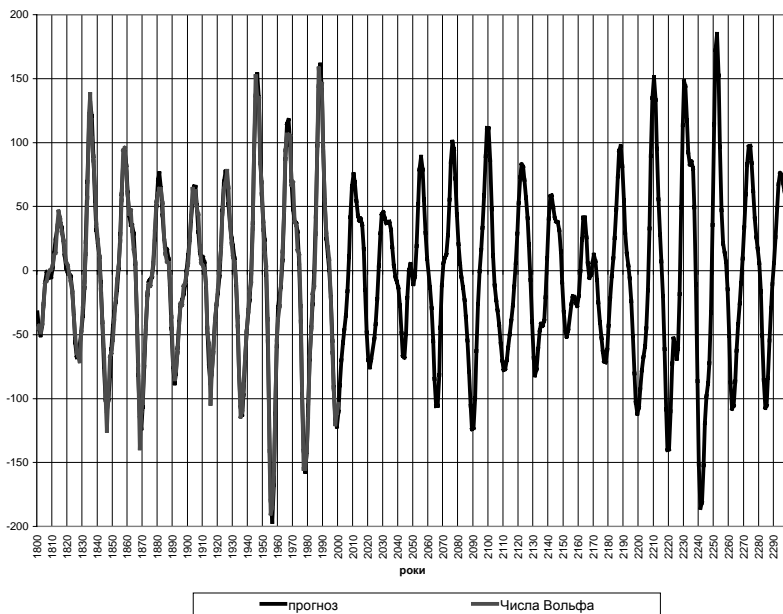


Рис. 4.

Як і у 1993 році, прогноуються локальні максимуми СА у травні та жовтні. Останній сплеск СА припаде на завершення передвборних змагань за президентське крісло, що сприятиме явці виборців на дільниці. А за кого вони голосуватимуть, буде залежати від їх уявлення про здатність претендента на задоволення їх нагальних проблем. І від врожаїв 2004 р., які, на нашу думку, будуть непоганими, бо опадів буде значно більше, ніж у 2003 р., а літньої спеки не очікується. Метеорологічні дані по Кіровоградщині теж були апроксимовані на основі даних спостережень на авіаметеостанції “Кіровоград”, люб’язно наданих її начальником М.М. Тисенюком.

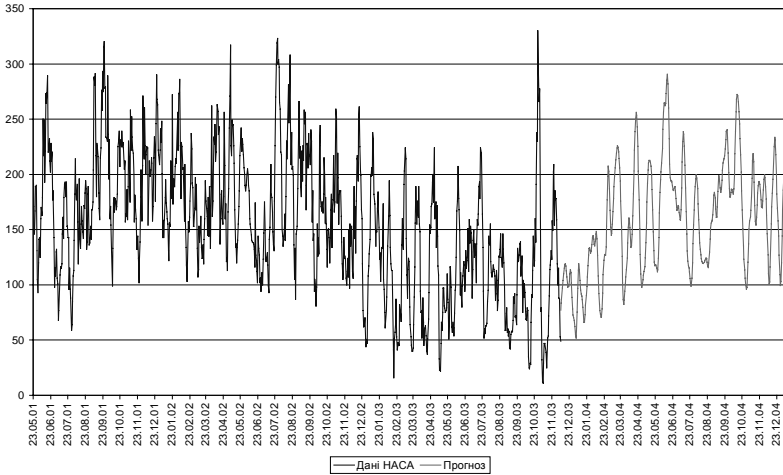


Рис. 5. Числа Вольфа за даними НАСА й прогноз–2004

3.2.1. Вивчення кореляції між майже періодичними часовими рядами за допомогою апроксимуючих многочленів. Зіставлення спектрів (w_k , A_k) двох рядів дає оцінку кореляції між ними. При цьому мінімізується сума квадратів відхилень значень однієї величини від прийнятої функції значень апроксимуючого тригонометричного многочлена другої величини.

Висновки.

1. Розроблений метод апроксимації майже періодичних коливань має достатню точність для інженерних та природознавчих застосувань (на рівні 0,1% для частот).
2. Створена програма тестована на ряді періодичних процесів та майже періодичних коливань суми несумірних складових.
3. За допомогою створеної програми можна аналізувати як детерміновані, так і стаціонарні випадкові процеси.
4. Програма застосована для спектрального аналізу середньорічних, середньомісячних та добових значень чисел Вольфа – характеристики СА та метеорологічних даних.

БЛЕСК И НИЩЕТА ВЕРБАЛЬНОГО ТРЕНИНГА

В.Н. Евтеев

г. Кривой Рог, Криворожский государственный педагогический университет
vne@kpi.dp.ua

Рассмотрим тест на терпимость к формальному изложению. Постараемся избежать чисто “философского” обсуждения, которое часто проявляется в том, что различные вещи обозначаются одним и тем же термином, а затем ведутся бурные дискуссии вокруг этой словесной конструкции. Зафиксируем следующие определения: мудрый – имеющий знания; грамотный – умеющий использовать знания по справочнику; умный – умеющий раздобывать знания. Затем создадим несколько словесных конструкций: мудрый, но не грамотный; грамотный, но не мудрый; мудрый, но не умный; умный, но не мудрый; грамотный, но не умный; и, наконец, умный, но не грамотный. Теперь отложите чтение и, используя принятые определения, проанализируйте на противоречивость указанные словесные конструкции. Если подобное занятие не вызывает у вас протеста, несмотря на его очевидную бесплодность. Или, если вы надеетесь все же получить пользу от таких упражнений – вы человек терпимый к формализациям.

Если же вам неприятны подобные действия, то переходите к следующей статье этого сборника. Если же эта статья расположена в конце сборника, и вы его уже прочитали, то располагайтесь на диване и предайтесь благостным размышлениям. Если же вы любитель просматривать сборники с конца на начало, то переходите к предыдущей статье. Если предыдущей статьи не существует, то воспользуйтесь советом из второго предложения. Если же вы выбрали данную статью из оглавления, то возвращайтесь к оглавлению. Если же вы выбрали данную статью наугад, то бросьте эту вредную привычку. Если вы занимаетесь отбором нужного вам материала, то стисните зубы и читайте дальше. Если вы рецензент, то поставьте отрицательную оценку и руководствуйтесь указаниям, которые перечислены выше. Если вы были невнимательны при чтении, и у вас возникло ощу-

щение, что вы что-то упустили, то вернитесь и прочитайте все еще раз сначала. Если что-то упущено, то прокляните автора за небрежность и продолжайте читать.

Если вы продолжаете читать, и в душе у вас нет ни гнева, ни сарказма, то вы прошли тест. Как вы уже догадались, собственно тестом был второй абзац, в котором подвергалась тестированию ваша способность выдержать нудную процедуру составления полной группы событий при организации ветвления в алгоритме. Известно, что наиболее трудно выявляемые логические ошибки связаны с некорректной организацией ветвлений. Как обучать способности избегать подобных ошибок? Простое препарирование множества типичных ошибок на частные примеры не приведет к желаемому результату. Здесь нужны учебные задания, при выполнении которых, ученик сам будет решать, что является важным, что второстепенным, а что совершенно лишним. Больше всего для этой цели подходят текстовые задания в силу своей гибкости и универсальности. Но в этом случае то, что является мощью и гибкостью изящной словесности может стать серьезной проблемой при построении логических конструкций. Речь идет о многозначности терминов. Хорошо, если наличие контекста снимает такую неопределенность. А если контекст отсутствует? В качестве примера приведем следующую цитату. «В условиях бурного роста человеческих знаний все большее значение приобретает развивающий характер обучения, так как все более необходимым становится принцип *“на немногом учить многому”*» [1]. Из контекста ясно, что термин «принцип» здесь применяется в значении – руководство к действию, но не в значении – нечто фундаментальное и незыблемое. Что получится, если мы просто провозгласим принцип *“на немногом учить многому”*. Попробуйте теперь, не игнорируя упомянутый “принцип”, научить устному счету или правописанию.

В названии статьи применен далеко не оригинальный прием с использованием вычурного термина «вербальный тренинг» вместо простого термина – «словесная тренировка». Это сделано исключительно из боязни использовать многозначные словесные конструкции. Под вербальным тренингом здесь понимаются такие тренажеры, в которых само задание и фрагменты, из которых конструируется ответ, представлены в текстовом виде. Конечно,

реальный тренажер содержит не только подобные упражнения, но и упражнения с формулами и графическими изображениями. Однако текстовые упражнения предназначены, в первую очередь, для тренировки оперирования абстрактными понятиями, не представленными в виде математических формул и иных символов. И как ни странно, некоторая весьма дозированная неоднозначность метаязыка позволяет представить абстрактное понятие в более выпуклом и многогранном виде.

С помощью текстовых заданий удается тренировать умение выполнять логические операции не только на формальном, но и на неформальном уровне. Математическая логика позволяет выполнять логические операции абсолютно “бездумно”, то есть, полностью пренебрегая содержательной стороной высказываний, оперируя этими высказываниями как символами, у которых все содержание сводится к истинностному значению. Однако под логикой мы не всегда понимаем именно формальную логику. Рассмотрим пример. Высказывание А: «Петя любит рок-музыку». Высказывание В: «Маша без ума от Пети». Высказывание С: «Маша терпимо относится к рок музыке». Секвенция – из А и В следует С на бытовом уровне выглядит вполне истинной, но вряд ли ее истинность можно проверить формально с помощью правил логического вывода. Справедливости ради следует отметить, что существует формальная теория нечетких множеств с ее вероятностной трактовкой и возможность создания на основе таких множеств алгебры, которую, в свою очередь, можно положить в основу логического исчисления. Но чем тогда хуже нечеткие понятия метаязыка, использование которых приводит к тем же результатам. Отсюда напрашивается вывод, что вербальные тренажеры являются наиболее подходящими инструментами для тренировки мышления, которое во многом ассоциируется с умением решать задачи и выполнять логические операции. А так ли это? Все ли мы учли? Полную ли группу возможностей мы рассмотрели? Попробуем более подробно поговорить о мышлении.

Первое слово философам. Цитата: «Логические машины являются лишь “...созданными человеческой рукой органами человеческого мозга...” (Маркс К., Энгельс Ф., Соч., 2 изд., т. 46, ч. 2, с. 215), истинным же субъектом мышления остается управ-

ляющий ими человек» [2]. Четкое и однозначное мнение - умение выполнять логические операции еще не мышление. Но это взгляд лишь с общих позиций. Теперь предоставим слово психологам. Вот краткий конспект по этому вопросу.

Мышление – основные формы: понятие, суждение, умозаключение. Рассматриваются формальной логикой.

Виды мышления: наглядно-действенное, наглядно-образное, словесно-логическое. Здесь виды мышления расположены в порядке их поэтапного проявления в процессе эволюции. С точки зрения направленности, цели и особенностей различают: теоретическое и практическое, теоретическое и эмпирическое, реалистическое и аутоистическое (переход во внутренние переживания), творческое и репродуктивное, непровольное и произвольное мышление [3].

Из всего многообразия приведенных классификаций рассмотрим лишь триаду видов мышления. Первые два из них – наглядно-действенное и наглядно-образное – представляют единое целое. Первое – просто-напросто недоразвитое второе. Словесно-логическое мышление стоит особняком от первых двух, поскольку является ярко выраженной знаковой системой. (Слово тоже знак.) А любая знаковая система обладает синтаксисом и правилами оперирования напрямую не связанными с конкретной семантикой знаков. Вспомним древнюю мудрость: “Ясно мыслю – ясно излагаю”. Но, так ли это? Или, если говорить более осторожно, в полной ли мере это так? Ясное изложение это всего лишь удачная словесно-логическая кодировка информации, а не сама информация. Но из факта удачной кодировки на выходе вовсе не следует, что и сама обработка информации осуществлялась с использованием такой кодировки. Говоря другими словами, словесно-логический вид обмена данными не гарантирует существование словесно-логического мышления. Позволим себе высказать гипотезу о том, что словесно-логическая форма не является видом мышлением, а представляет собой лишь интерфейсную часть в процессе общения носителей разума.

Попробуем обосновать эту гипотезу. Поэт сказал: “Победа энтропии черной”. Но, в частности для автора, энтропия всегда была желтой. Попробуйте представить, какого цвета энтропия для вас. И если у вас ничего не получится, то это означает только

одно – вы не привыкли оперировать понятием “энтропия”. Тогда выберите более привычное для вас абстрактное понятие, и у вас сразу все получится. Вы обнаружите, что можете приписать выбранному понятию не только форму и цвет, но и вкус, запах и если постараться, то и тактильные ощущения. Это происходит потому, что абстрактные понятия представлены внутри нашего мыслительного аппарата не в виде словесных символов, а в виде образов, которые, хоть и опосредовано опираются на все основные виды ощущений. Еще один пример: вспомните свои выступления на научных семинарах. Как ведут себя остальные участники? Их интересует в первую очередь постановка задачи. На этом этапе возможны самые неожиданные для вас вопросы такого типа: “А что вы имеете в виду под тем-то и тем-то” и т.п. Идет процесс интенсивного перевода услышанного на свой внутренний язык базовых образов. Причем то, что эти языки сугубо индивидуальны видно вооруженным глазом. Для ответов на вопросы одного участника следует выбирать термины с ярко выраженной эмоциональной окраской. Другой настойчиво требует написать гамильтониан задачи, и, получив соответствующую формулу, перестает слушать, углубившись в анализ. Для него каждый терм формулы – живой объект. В противовес такой индивидуальности следует заметить, что метаязык один для всех. Можно было бы продолжать обсуждение. Но автор надеется, что червь сомнения уже активирован в душе читателя.

Остается много вопросов. Если гипотеза, хотя бы частично, верна, то как ее использовать при составлении текстовых заданий? И все же, кто ты – словесная тренировка? Солидная статс-дама, или легкодоступная куртизанка? Или, возможно, ты вообще не женщина?... И так далее по полной группе событий.

Литература

1. Богословский В.В., Бочкарева Т.И., Великороднова А.И. Практикум по общей психологии: Учебное пособие. – М.: Просвещение, 1979. – 302 с.
2. Б.И. Франкштейн. Мышление // БСЭ – М.: «Советская энциклопедия», 1974. – Т. 17.
3. Психология: Словарь. / под. общ. ред. А.В. Петровского, М.Г. Ярошевского – М.: Политиздат, 1990. – 494 с.

МЕТОДИКА РОЗПІЗНАННЯ PAST INDEFINIT I PARTICIPLE II ДЛЯ СИСТЕМ МАШИННОГО ПЕРЕКЛАДУ

О.В. Ємець¹, К.М. Скиба¹, В.М. Шепеленко², С.В. Марченко²,
Т.О. Рудик¹

¹ м. Хмельницький, Хмельницький державний університет

² м. Київ, Трайдедент Софтвер
arudyk@rambler.ru¹, trident@kagi.com²

Питання про широке використання систем комп'ютерного перекладу в Україні залишається відкритим. З одного боку, зрозуміла доцільність застосування сучасних лінгвістичних програмних інструментів, які забезпечують підвищення ефективності перекладачів, референтів, бізнесменів та інших фахівців, які працюють з актуальною англомовною інформацією. З другого боку, не дивлячись на те, що системи комп'ютерного перекладу з кожним роком стають доступнішими, поки що існує певною мірою недовіра до них.

Окрім причин методичного, психологічного і технологічного характеру, які перешкоджають впровадженню інформаційних і комп'ютерних технологій в учбовий процес, існують причини чисто прагматичні – неякісність машинного перекладу [1]. Редагувати згенеровані комп'ютером тексти на порядок складніше, ніж переклади, виконані живим автором. Якісний комп'ютерний переклад відбирає не менше (якщо не більше) висококваліфікованої лінгвістичної праці, ніж замовлення живому штатному або сторонньому перекладачу [2].

Кращий результат досягається при перекладі технічних текстів завдяки накопиченню термінологічних словників, які дозволяють більш цілеспрямовано і швидко видавати концептуально замовлені значення слів для даної галузі знань. Але обійтися без перевірки і виправлень одержаного перекладу людиною не вдається.

Якість машинного перекладу залежить не тільки від об'єму словника та об'єму інформації, приписуваної лексичним одиницям, але і від ретельності складання й перевірки роботи алгоритмів аналізу і синтезу. Тому використання автоматизованих гра-

матичних довідників не є таким ефективним, що пов'язано із складністю створення алгоритму перекладу граматичних форм, наприклад, з англійської мови на українську. Причиною є омонімія граматичних форм в англійській мові, насамперед, дієслівна форма із закінченням -ed. Так, слово із закінченням -ed (наприклад, translated) може бути:

- дієсловом-присудком у Past Indefinite (Simple Past);
- частиною дієслова-присудка у Passive Voice, або Present Perfect, або Past Perfect;
- дієприкметником минулого часу (Participle II);
- частиною дієприкметника теперішнього часу (Participle I).

Студенти, які вивчають англійську мову, часто не в змозі запам'ятати усі дієслівні часи і безособові форми дієслів і тому часто губляться у перекладі речень з такими омонімічними граматичними формами. Тому виникає потреба створення автоматизованого комплексного граматичного довідника, який би містив алгоритми розпізнавання та перекладу складних дієслівних форм. Один з варіантів – його вбудовування в існуючу систему машинного перекладу (СМП), що одночасно дозволить покращити якість перекладу даною системою.

На даний момент в Україні серед компаній-розробників СМП найкраще зарекомендувала себе Trident Software (<http://www.trident.com.ua>) з системою Pragma 2.0 [3, 4].

Pragma – професійна 32-розрядна система для автоматизованого перекладу документів, яка успадкувала всі лінгвістичні та інтерфейсні розробки СМП L-Master. Вона дозволяє працювати в одній оболонці з кількома мовними напрямками перекладу і забезпечує зручні засоби як для інтерактивного перекладу документів, так і для їхнього наступного редагування. Але проблема омонімії дієслівної форми із закінченням -ed для цієї СМП також до кінця не вирішена.

Представлений на рис. алгоритм (V – verb – дієслово) побудований у послідовності, яка відповідає динаміці становлення граматичних навичок у рамках функціонально направленої граматики. Він дозволяє сформулювати у студентів, які вивчають англійську мову, розуміння значення граматичної структури і засобів її вираження. Впровадження алгоритму у СМП Pragma, у

свою чергу, дозволить покращити якість перекладу дієслівної форми із закінченням -ed.

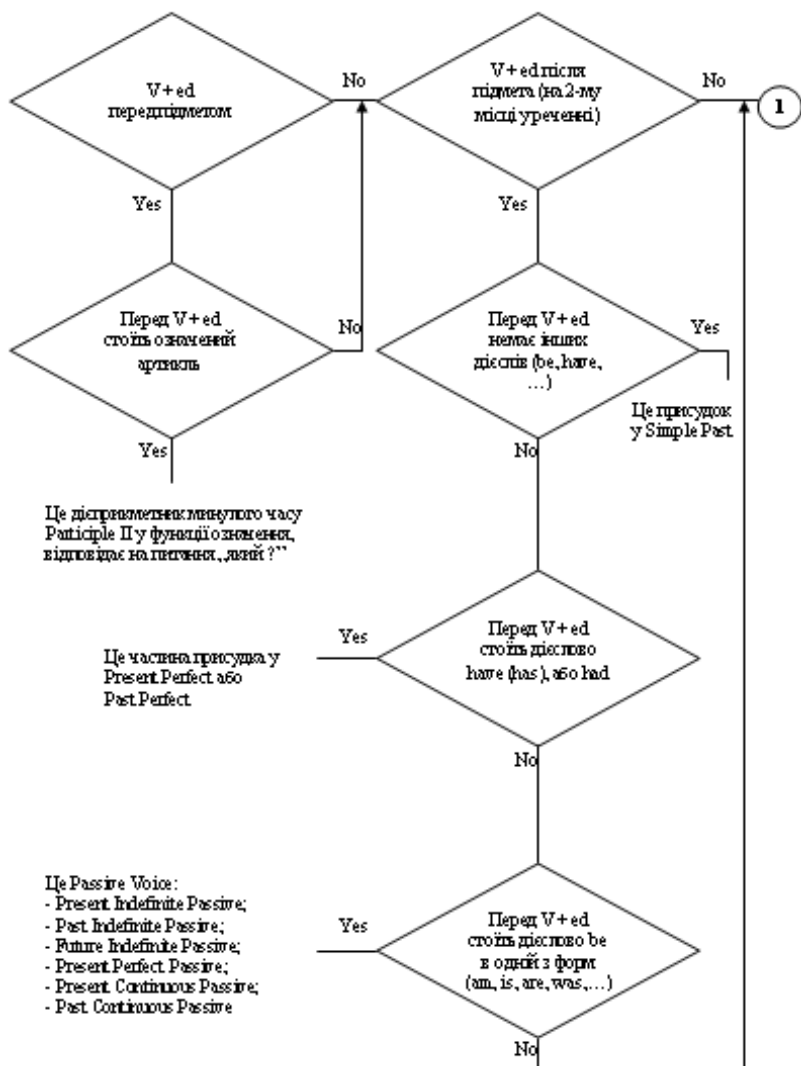


Рис. Блок-схема алгоритму розпізнання Past Indefinite і Participle II для систем машинного перекладу

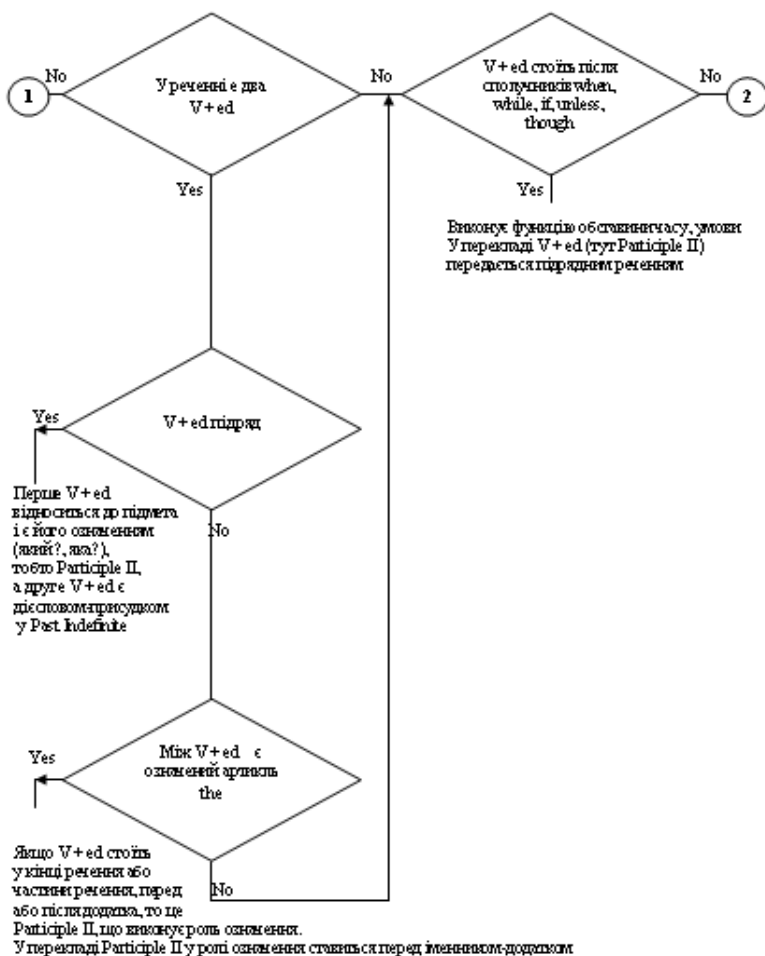


Рис. Блок-схема алгоритму розпізнання Past Indefinite і Participle II для систем машинного перекладу (продовження)

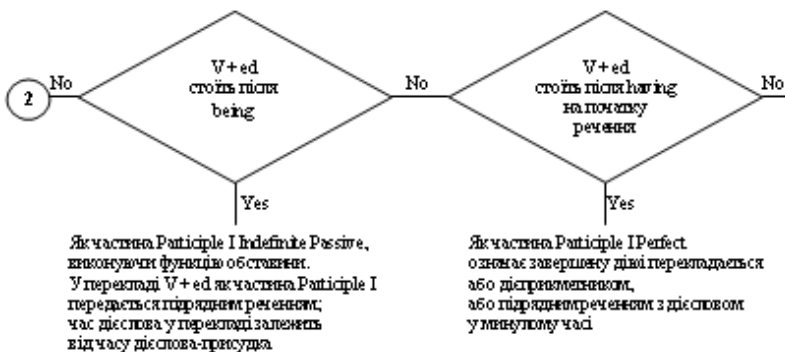


Рис. Блок-схема алгоритму розпізнання Past Indefinite і Participle II для систем машинного перекладу (закінчення)

Література

1. <http://www.strana.kaliningrad.ru/N41/trans.htm>
2. <http://vi.cityline.ru/vi/05oct1997.htm>
3. Рудик О.Ю., Скиба К.М. Рекомендації з вибору ліцензійних СМП (україно-російсько-українська підсистема) / Зб. наук. праць №21, Ч2 (спец. випуск). – Хмельницький: НАПВУ, 2002. – С. 341–345.
4. Рудик О.Ю., Скиба К.М., Сторожук І.Ф. Системи машинного перекладу (українсько- та російсько-англійська підсистема). / Зб. наук. праць №21, Ч2 (спец. випуск), – Хмельницький: НАПВУ, 2002. – С. 346–353.

ІНФОРМАЦІЙНІ СИСТЕМИ ПРИ ВИВЧЕННІ ДИСЦИПЛІН ЕКОНОМІЧНОГО ПРОФІЛЮ

Н.С. Завізна

м. Одеса, Одеський національний університет ім. І.І. Мечнікова
nata@hitfm.odessa.ua

При широкому використанні комп'ютерів в навчальному процесі логічно постає проблема розробки програмних продуктів, якими могли б користуватися студенти з різним досвідом роботи технічними засобами. Фактор зручності та наступності використання розроблених засобів визначається тим, наскільки повно враховані психолого-педагогічні вимоги до них. На сучасному етапі розвитку комп'ютерів та комп'ютерних програм настільки спростився процес роботи на ЕОМ, що для користувачів комп'ютерів вже не йдеться мова про знання комп'ютерних мов.

Маючи сьогодні такі потужні технічні засоби, на жаль, ще не можемо говорити про всебічне застосування і значну роль комп'ютерів у вивченні дисциплін різних циклів.

Необхідною умовою інформатизації навчального процесу є підготовка викладача до використання тих технологій навчання, які називаються новими інформаційними. Таким чином, важливе місце сьогодні відводиться вирішенню проблеми підготовки висококваліфікованих педагогічних кадрів, які в епоху комп'ютеризації та інформатизації володіли б глибокими теоретичними, практичними й професійними знаннями, уміннями та навичками. Тому відчутна потреба у зміні й розвитку системи навчального процесу у вищих економічних навчальних закладах, впровадженні нових підходів і методів, організації навчального процесу з використанням комп'ютерів, що сприятиме гармонійному всебічному розвитку кожного студента, стимулюватиме його творчий пошук, активізуватиме процеси пізнання, прийняття самостійних рішень, моделювання реальних економічних процесів.

Процес професійного становлення майбутнього економіста відстає від вимог, що висуваються суспільством, здатного до використання нових комп'ютерних технологій у своїй галузі діяльності; самостійного та творчого прийняття рішень; самовдоско-

налення та саморозвитку своїх можливостей, здібностей, умінь; який адекватно оцінює свої дії в роботі.

Широке використання комп'ютерів у різноманітних галузях промисловості, науки, освіти вносить кардинальні зміни до цих галузей, стимулює подальший їх розвиток та пропонує нові способи роботи з інформацією, її організацією й новими інформаційними технологіями.

Кожен випускник вищої школи з економічною освітою повинен відповідати сучасним запитам, у тому числі й в плані інформаційної культури. Потрібно відзначити, що термін “інформаційна культура” у педагогіці, в першу чергу, пов'язують з комп'ютеризацією навчання. На наш погляд, це відбувається тому, що комп'ютерні технології займають, разом зі спеціалізацією, провідну позицію на сучасному етапі навчання у вищій школі. Ми говоримо більше про вплив комп'ютерних технологій на формування інформаційної культури студентів, з однієї простої причини: це питання досліджується в контексті навчального процесу вищої школи.

За допомогою комп'ютера створюється можливість розв'язання задач, які не розв'язуються або розв'язуються неефективно без комп'ютера, а саме:

- зміна форм і видів взаємодії студентів та викладачів на основі використання комп'ютерів у навчальному процесі;

- варіювання видів діяльності при роботі з комп'ютером, отримання опису стану індивідуально-психологічних можливостей студентів, їх знань, умінь і навичок та їх коригування з метою підвищення ефективності навчального процесу й контролю з боку викладача;

- комп'ютер у порівнянні з іншими технічними засобами, які використовуються в навчальному процесі, дозволяє повніше враховувати індивідуальні й вікові особливості студента, що створює умови для виведення навчального процесу на новий рівень.

Важливим аспектом є використання інформаційних технологій при вивченні дисциплін економічного профілю у вищій школі. Для вивчення пропонуються комп'ютерні програми для ведення кадрового, бухгалтерського й оперативного обліків – 1С: Підприємство і «Парус».

Програма «Парус» складається з функціонально окремих розділів: «Адміністратор», «Бухгалтерія», «Реалізація і склад».

Розділ «Адміністратор» використовується для настроювання всіх розділів програми. Дії, які реалізуються за допомогою цього розділу:

- створення баз даних;
- відновлення баз даних;
- розподіл прав доступу;
- усунення помилок у роботі програми.

Розділ «Реалізація і склад» разрахований на ведення оперативного обліку. У ньому реалізовані звіти: «Рух товарів», «Товарні запаси», «Резервування товару», «Розрахунки з контрагентами», «Оплата рахунків», «Прайс-лист», «Зміни цін», «Доходи від реалізації», «Продажу», «Структура продажів», «Продажу по товарах», «Заборгованості», «Додаткові звіти».

Розділ «Бухгалтерія» необхідний для ведення обліку господарських операцій, роботи з документами й оформлення бухгалтерської звітності. У цьому розділі існують наступні підрозділи: «Словники», «Шаблони документів», «Найменування і курси валют», «Контрагенти», «Намеклатор», «Зразки», «Константи».

Шаблон документа являє собою бланк, що містить усі необхідні атрибути. Користувач може створювати новий бланк.

Курси валют реєструються і використовуються в операціях. Усі зміни курсу зберігаються.

Словники – своєрідна база даних, в них утримується допоміжна інформація, що використовується при заповненні облікових регістрів.

Намеклатор, дозволяє організувати і зберегти інформацію про запаси, облік рухів, що здійснюється за допомогою програми.

У програму ІС: Підприємство входять:

- режим користувача (призначений для безпосереднього введення, обробки, збереження і видачі інформації про діяльність підприємства);
- режим конфігуратора (призначений для настроювання компонентів системи).

ІС: Підприємство являє собою наступну структуру:

- журнали: документів, операцій, проводок, розрахунків;

- довідники;
- план рахунків;
- розрахунок заробітної платні;
- звіти;
- керування бухгалтерськими підсумками;
- керування оперативними підсумками.

1С: Підприємство і «Парус» – це комплексні програми автоматизованого бухгалтерського обліку підприємств, за допомогою яких швидко і якісно можна вести розрахунки.

Економічні університети мають відігравати провідну роль у вирішенні проблеми підготовки фахівців високої кваліфікації, які в епоху комп'ютеризації та інформатизації всіх сфер людської діяльності володіли б глибокими теоретичними, практичними і професійними знаннями, вміннями й навичками роботи з комп'ютерною технікою, вміли й знали, як ефективно працювати з інформацією. Тому відчутна потреба у зміні й розвитку системи вищого навчання, впровадженні нових підходів і методів.

Досвід показує, що залучення комп'ютерних програмних продуктів до навчального процесу дозволяє інтенсифікувати цей процес для студентів і допомагає їм набути навички самостійної роботи з комп'ютерними програмами економічної спрямованості.

ВНЕДРЕНИЕ ТЕСТОВЫХ ПРОГРАММ В УЧЕБНЫЙ ПРОЦЕСС

Н.С. Завизена, Е.Г. Гуржий
г. Одесса, Одесский национальный университет
им. И.И. Мечникова
nata@hitfm.odessa.ua

Необходимым условием информатизации учебного процесса является подготовка преподавателя к использованию тех технологий обучения, которые называются новыми информационными технологиями. Приведем примеры определений термина «новые информационные технологии» (НИТ) некоторых исследователей. «Нужно отметить, что понятие НИТ понимается широко и неоднозначно, и сводится чаще к описанию средств обучения, доступных для использования в учебном процессе баз данных, электронных таблиц, информационных сетей» [1]. Наиболее широкое определение этого термина приводит В. Пономаренко: «Информационные технологии – средства обработки информации и организационно-управленческие концепции его формирования и применения, а также совокупность всех видов информационной техники; единство процедур относительно собирания, накопления, сохранения, обработки и передачи данных с применением избранного комплекса технических средств» [2]. Успешное применение компьютера в учебном процессе предусматривает рассмотрение его (вместе со всеми составными частями) как неотъемлемую часть этого процесса. Новейшие системы позволяют моделировать учебный процесс так, что не только сохраняются основные преимущества обучения с педагогом, но и прибавляются при этом неисчерпаемые возможности гибкости управления. В этом направлении работали исследователи Г. Бордовский, Т. Носкова, В. Буравихин, В. Жданов, А. Войкунский, В. Гриценко, Б. Паншин, Л. Зазнобина, И. Марусева, И. Роберт.

Новые информационные технологии обучения вносят существенные изменения в процесс обучения. На данный момент создано множество различных обучающих программ, которые позволяют студенту, не обращаясь к преподавателю, получить информацию по тому либо иному вопросу, оценить свои знания и

умения без чьей либо помощи. Таким образом, студент может оценить свои знания без страха сделать ошибку, зная, что об этом неизвестно преподавателю и другим студентам, а учебная задача, тем не менее, будет решена, потому что с помощью компьютера студент получит подсказку по данному вопросу и ошибка будет устранена.

Новые информационные технологии вносят существенные изменения и в деятельность преподавателя. Его роль в учебном процессе зависит от типа обучающей программы и способов ее использования. Чаще всего преподаватель на некоторое время исключается из учебного процесса и тогда студент взаимодействует непосредственно с компьютером. Помощь преподавателя становится необходимой только в том случае, когда возникают определенные трудности с использованием обучающей программы.

Важнейшую роль в обучении студента играет контроль знаний, а также самоконтроль. Объективная оценка собственных возможностей является стимулом для новых достижений. Если преподаватель при разработке структуры учебного процесса большое внимание уделяет самостоятельному закреплению полученных знаний студентами, то его наверняка заинтересует внедрение тестовых программ в учебный процесс. В этом случае компьютер является незаменимым помощником, независимо от того, о каком предмете идет речь: биология, физика, история или любой другой предмет. Программная реализация тестовых заданий позволит преподавателю обработать большой объем данных, создав при этом индивидуальное тестовое задание для каждого студента в зависимости от поставленных целей, степени сложности, а также с учетом способностей тестируемого.

Для создания компьютерной тестовой программы нами была использована среда визуального программирования Delphi, которая позволяет легко создавать структурированные приложения. При запуске тестовой программы на экране появляется вопрос и различные варианты ответа на него. Если студент дает правильный ответ на поставленный перед ним вопрос, то в его актив добавляется один балл и на экране появляется следующий вопрос. Если же ответ был неверным, то на экране появляется еще одно окно, в котором располагается полная информация по

данному вопросу. При этом параллельно происходит подсчет количества заданных вопросов и количество набранных баллов. В любой момент времени студент может узнать о достигнутых им успехах. В то же время преподаватель не тратит времени на обработку полученных результатов, так как они автоматически передаются по сети в базу данных, которая находится на машине преподавателя.

Приложение имеет клиент-серверную архитектуру. В такой архитектуре клиентское приложение, устанавливаемое на машине студента, инициирует запрос, а серверное приложение, которое находится на машине преподавателя, пассивно отвечает на него. Тестовые вопросы, варианты ответов, правильные ответы и подсказки на случай ошибочных ответов хранятся в базе данных на машине преподавателя. Там же хранится еще одна база данных – база данных студентов, в которой находится следующая информация: класс, фамилия, пароль, количество заданных вопросов, количество правильных ответов. При запуске серверного приложения для доступа к базам данных преподаватель вводит пароль, после чего на экране появляется окно, содержащее обе базы данных. При этом преподаватель имеет возможность редактирования баз данных, не выходя из приложения. Для этого у него есть кнопки: добавить запись, удалить запись, очистить все содержимое базы данных, сохранить изменения в базе данных.

Взаимосвязь клиента и сервера происходит с помощью технологии межпроцессного взаимодействия NamedPipe (именованные каналы). Эта технология позволяет создавать приложения с двусторонним каналом связи и гарантией доставки сообщения. Для этого на сервере создается экземпляр именованного канала с помощью функции CreateNamedPipe. Далее созданный канал ожидает подключения к нему от клиента (ConnectNamedPipe). Если клиент подключился, то происходит обмен информацией (Read/Write). При закрытии серверного приложения канал отключается (DisconnectNamedPipe) и экземпляр канала необходимо закрыть (CloseHandle).

При запуске клиентского приложения на экране появляется регистрационное окно, в которое студент вносит свои данные: класс, фамилия, пароль и для соединения с сервером – IP адрес машины, на которой установлено серверное приложение. По за-

вершении этой процедуры происходит прослушивание канала связи (WaitNamedPipe) и если он свободен, то клиент подключится к уже созданному каналу связи (CreateFile) и регистрационные данные поступят в канал связи (Write). Сервер получит эти данные и внесет их в базу данных студентов.

Далее на экране клиентского приложения появится окно с подробным описанием работы данной программы. Как только студент объявляет о готовности начать тест (нажимает на кнопку «Далее»), происходит прослушивание канала связи и, если он свободен, то по нему передается номер вопроса. Сервер, получая номер вопроса, по нему отыскивает нужный вопрос в базе данных и передает по сети клиенту такую структуру: вопрос, ответ 1, ответ 2, ответ 3, ответ 4, правильный ответ, подсказка.

Таким образом, перед студентом появляется следующее окно, в котором содержится вопрос и четыре варианта ответа на него. Определившись с ответом, студент нажимает на кнопку «Далее». По нажатии этой кнопки происходит обработка результата относительно правильного ответа, полученного от сервера. Если ответ был дан верно, то к счетчику правильных ответов добавляется единица, иначе появляется окно подсказки по этому вопросу, в которое загружается подсказка, полученная от сервера. После ознакомления с ней появляется следующий вопрос, а к счетчику общего количества вопросов добавляется единица.

В любой момент обучаемый может получить информацию о количестве набранных им баллов по активизации кнопки «Получить результат», либо автоматически в конце теста. А по окончании теста количество набранных баллов будет отправлено по каналу связи серверу, который занесет их в базу данных студента.

По завершению работы преподавателю достаточно заглянуть в базу данных студента для установления полученных результатов, а не тратить время на ручную обработку данных. Так же и обучаемый знает свои ошибки и достигнутый им результат уже в процессе тестирования.

Такая тестовая программа является многофункциональным помощником как преподавателя, так и студента, играя огромную роль в процессе обучения.

Литература

1. Гребнев И.В. Методическая подготовка студентов университетов // Педагогика. – 1996. – №1. – С. 37-40.

2. Інформаційні системи і технології в економіці: Посібник для студентів вищих навчальних закладів / За редакцією В.С. Пономаренка. – К.: Видавничий центр “Академія”, 2002. – 544 с.

З ДОСВІДУ ВИКЛАДАННЯ КУРСУ ІНФОРМАТИКИ ДЛЯ СТУДЕНТІВ ЕКОНОМІЧНИХ СПЕЦІАЛЬНОСТЕЙ

О.Ф. Клименко¹, Ю.М. Красюк¹, Л.М. Петренко¹,
В.В. Соловйова²

¹ м. Київ, Київський національний економічний університет
² м. Черкаси, Інститут соціального управління, економіки і права
v_n_soloviev@isuep.uch.net

Одним із актуальних завдань, яке стоїть сьогодні перед вищою освітою, є формування основ інформаційної культури студентів, достатніх для впевненого та ефективного використання сучасних інформаційних технологій у їх повсякденному житті та власній професійній діяльності. Значну роль у вирішенні цього завдання відіграє курс інформатики та комп'ютерної техніки.

Нами розроблений робочий зошит з курсу “Інформатика та комп'ютерна техніка” призначений для підготовки бакалаврів за напрямом “Економіка та підприємництво”. Його створено відповідно до вимог проведення практичних занять, лабораторних та самостійних робіт передбачених програмою даного курсу [1]. Провідною ідеєю зошиту є те, що “Інформатика та комп'ютерна техніка” – дисципліна стимулювання активної навчально-пізнавальної діяльності студентів для:

- оволодіння науковими знаннями з:
 - теоретичних основ принципів побудови та функціонування сучасних і перспективних обчислювальних машин;
 - принципів організації обчислювальних процесів на персональних комп'ютерах (ПК) і основ їх алгоритмізації;
 - основ сучасного програмного забезпечення персональних комп'ютерів та комп'ютерних мереж;
- оволодіння навичками та вміннями перспективно використовувати нові інформаційні технології як для вирішення відомих проблем, так і для розв'язування нестандартних, непередбачуваних життєвих та професійних ситуацій.

Формування практичних навичок та умінь роботи з ПК та відповідним програмним забезпеченням досягається в циклі практичних занять та лабораторних робіт, а також під час самостійної пізнавальної діяльності студентів на ПК у позаурочний

час.

Протягом вивчення курсу “Інформатика та комп’ютерна техніка” студент заповнює робочий зошит з дисципліни, відповідно до графіку виконання практичних занять та лабораторних робіт, графіку проведення оперативного контролю, контрольних тестів та контрольних робіт.

Мова заповнення – українська.

Оцінювання знань студентів проводиться у двох напрямках:

1) контроль систематичності та активності роботи студента протягом семестру над вивченням програмного матеріалу дисципліни;

2) контроль за виконанням модульних завдань.

Результати виконання контрольних робіт та контрольних тестів оцінюються в 30, 25, 20, 15, 10, 5 та 0 балів.

Лабораторна робота (розрахована на 2 години) оцінюється в 0, 5 або 10 балів.

Результати практичних занять оцінюється в 0, 5 або 10 балів.

Використання робочого зошита при вивченні курсу інформатики показало його високу практичну ефективність. При цьому реально знизилось як число студентів, недопущених до екзамену, так і число студентів, які отримують допуск до екзамену “в останню мить”.

Крім того, як з’ясувалося, підвищилась успішність студентів по результатах семестру. Робота студентів стала більш ритмічною і плановою.

Література

1. Клименко О.Ф., Головка І.Р., Шарапов О.Д. Інформатика та КТ. – Київ: КНЕУ, 2002. – 456 с.

ЛАБОРАТОРНІ РОБОТИ ЯК МЕТОД АКТИВІЗАЦІЇ ПІЗНАВАЛЬНОЇ ДІЯЛЬНОСТІ СТУДЕНТІВ З ДИСЦИПЛІН КОМП'ЮТЕРНОГО ЦИКЛУ У ВИЩИХ АГРАРНИХ НАВЧАЛЬНИХ ЗАКЛАДАХ

О.В. Ключко

м. Вінниця, Вінницький державний аграрний університет
klochkoob@rambler.ru

Застосування сучасних інформаційних технологій – показник, за допомогою якого можна визначити стан тієї чи іншої галузі знань. Спеціаліст аграрного сектору економіки будь-якої спеціальності повинен володіти навичками застосування персонального комп'ютера для обробки результатів роботи, планування, обліку, аналізу і оперативного управління економічними і технологічними процесами.

Курс інформатики у вищих аграрних навчальних закладах передбачає, окрім вивчення системного програмного забезпечення та основ комп'ютерної техніки і програмування, вивчення пакетів прикладних програм та застосування інформаційних технологій у професійній діяльності. В межах професійного спрямування під час вивчення даного курсу необхідно акцентувати увагу студентів на застосування інформаційних технологій у професійній діяльності. Тому для вищої школи лабораторні роботи стали одним із найважливіших методів самостійної роботи студентів і являють собою практику через дію.

Лабораторні роботи інтегрують теоретико-методологічні знання, практичні навички і уміння в єдиному процесі діяльності навчально-дослідницького характеру. Лабораторні роботи є одним з інтерактивних методів навчання. Найменших результатів можна досягти за умов пасивного навчання (лекція – 5%, читання – 10%), а найбільших – інтерактивного (дискусійні групи – 50%, практика через дію – 75%, навчання інших, або негайне застосування знань – 90%) [1].

На сучасній технічній базі лабораторні роботи не стільки закріплюють теоретичні знання, скільки дозволяють студентові глибше і наочно вивчати механізм застосування теоретичних знань, оволодівати надзвичайно важливим для спеціаліста умін-

ням інтелектуального проникнення у ті природничі, технічні або виробничі процеси, які досліджуються в лабораторному практикумі. Під впливом цієї форми занять (особливо “спеціальних практикумів”) у студентів часто виникають нові ідеї наукового і технічного характеру, які використовуються у курсових кваліфікаційних, дипломних роботах, по лінії НСТ і СКБ. Розвиваються творчі здібності студентів, їхні дослідницькі інтереси. У лабораторіях вищих навчальних закладів формуються навички і уміння майбутніх спеціалістів для роботи у науково-дослідницьких закладах, на виробництві, а у лабораторіях педагогічних навчальних закладів студенти, крім того, вчать проводити лабораторні практикуми у школі.

Сформулюємо основні вимоги до організації і проведення лабораторних занять [2]:

1. Важливою умовою ефективності проведення лабораторних є теоретична підготовка студентів і попередня перевірка викладачами готовності студента (“допуск” до роботи).

2. Необхідною умовою організації лабораторної роботи є підготовка інструкції щодо її проведення, яку студенти мають уважно вивчити на початку заняття. Для контролю готовності використовуються різні методи, в тому числі і програмовані, за допомогою комп’ютерів.

3. Оскільки в лабораторних роботах доводиться, як правило, виконувати вимірювання і наступні обчислення, важливі уміння студента правильно користуватися вимірювальними приладами, а при обробці результатів вимірювання – використовувати наближені обчислення.

4. На лабораторних заняттях однією з ефективних форм навчальної діяльності є спільна групова робота. Важливо так організувати роботу групи, щоб кожний студент у ній знайшов своє місце, відповідну роль, належну частку участі у виконанні роботи і відповідну оцінку викладача.

Так, у Вінницькому державному аграрному університеті з метою надання студентам можливості самостійно опрацьовувати лекційний матеріал, а також набувати навички практичної роботи з комп’ютером, розроблено і впроваджено в навчальний процес системи лабораторних робіт, як друковані, так і у електронному варіанті. З метою диференційованого підходу до навчання

студентів різних спеціальностей створено цикли лабораторних робіт з окремих розділів курсу інформатики, а саме поняття інформації, інформаційні системи, операційні системи, робота з дисками, прикладне програмне забезпечення загального призначення. Крім цього, з предметів інформаційного циклу створено комплекс лабораторних робіт із застосуванням пакетів прикладних програм для розв'язання професійних задач. Система лабораторних робіт розроблена таким чином, щоб за допомогою надлишкової кількості робіт можна було варіювати тематику робіт для кожного студента в залежності від професійної спрямованості і рівня підготовки студента.

Методичні рекомендації для кожної роботи складаються з назви лабораторної роботи, мети, теоретичних відомостей, багатоваріантного практичного завдання, контрольних запитань. Завдяки цьому студенти можуть виконувати кожну роботу не тільки під час аудиторних занять за розкладом, а і під час самостійної роботи в комп'ютерних класах в позаурочний час.

Наведемо план-схеми проведення лабораторних практикумів для студентів різних спеціальностей аграрного вищого навчального закладу.

Лабораторна робота для студентів агрономічного зооінженерного, обліково-фінансового, економічного факультетів, факультету механізації сільського господарства з курсів “Інформатика і комп'ютерна техніка”, “Основи комп'ютерної техніки і програмування”:

Тема: Вивчення основ роботи в Internet.

Мета: Навчитись оперувати із Internet-броузерами (Internet Explorer або Netscape), одержати навички навігації в Internet та пошуку інформації за допомогою пошукових систем.

Теоретичні відомості.

Завдання: Для пошуку необхідної інформації треба за допомогою WEB-броузера зайти на обраний сайт і в вікні для пошуку набрати ключові слова. Ключові слова – це слова або їх сполучення, які потенційно можуть зустрічатись в тих електронних виданнях, що Вас цікавлять. Наприклад, треба знайти інформацію про місто Вінницю. Якщо Ви вкажете тільки слово “Вінниця” або “Винница” або “Vinnitsa” тобто одне слово, то у відповідь пошуковий сайт покаже цілу низку сторінок, де зустрі-

чається таке слово. Якщо, наприклад, Ви захочете подивитись сторінки із словом, що досить часто зустрічається (припустимо “комп`ютер”), то у відповідь Ви одержите повідомлення, що знайдені тисячі або мільйони (залежить від потужності пошукового сайту) сторінок.

Правильніше, коли Ви звузите пошук і запросите сторінки за таким, наприклад, запитом “Винница университет”. Це буде означати, що пошукова система має знайти сторінки, де одночасно зустрічаються ці два слова. Кількість сторінок в цьому випадку буде суттєво меншою.

Контрольні питання:

1. Які назви Internet-мереж Вам відомі?
2. Що таке домен і субдомен?
3. Для чого використовується слеш в Internet-адресах?
4. Які елементи може мати WEB-сторінка та які вони мають властивості?

Для студентів обліково-фінансового і економічного факультетів з предмету “Математичне програмування” з використанням пакету GRAN1 пропонується лабораторна робота “Графічний метод розв’язання задач лінійного програмування за допомогою пакета GRAN1”. Як правило, задача лінійного програмування розв’язується графічним методом вручну, поряд з цим студентам пропонується застосувати прикладні програми для розв’язання задачі лінійного програмування. Одним із таких засобів є педагогічний програмний засіб GRAN1. Наведемо приклад розв’язання задачі графічним методом із застосуванням пакету GRAN1:

Тема: *Графічний метод розв’язання задач лінійного програмування за допомогою пакета GRAN1.*

Мета: Навчитись розв’язувати задачі лінійного програмування графічним методом з використанням пакета GRAN1.

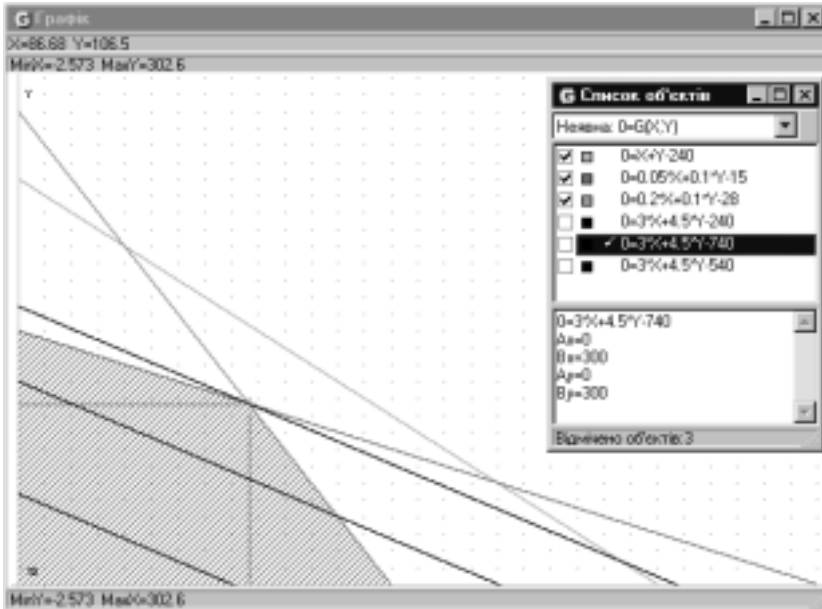
Теоретичні відомості.

Визначити площу саду та винограднику при використанні таких ресурсів: рілля під багаторічні насадження – 240 га, трудові ресурси – 15 тис. люд-діб, грошові ресурси – 28 тис. гр. Відомі витрати виробничих ресурсів на 1 га площі саду та винограднику наведено в таблиці 1.

Таблиця 1.

Ресурси	Сад	Виноградник
.Трудові, тис. люд.-діб	0.05	0.1
.Грошові, тис. гр.	0.2	0.1
.Вартість продукції, тис. гр.	3	4.5

Графічний метод розв'язання задачі за допомогою пакету GRAN1:



За рисунком студенти визначають оптимальне значення функції $Z_{\max} = 3 \cdot X_1 + 4,5 \cdot X_2$, яке досягається в точці $X_1 \approx 86,7$ (га), $X_2 \approx 106,7$ (га).

Завдання. Контрольні питання.

На даній лабораторній роботі студенти швидко опановують навички розв'язання задачі графічним методом із застосуванням пакету GRAN1, можуть за одну лабораторну роботу розв'язати більшу кількість задач лінійного програмування.

Для студентів обліково-фінансового і економічного факультетів з предмету "Математичне програмування" з використанням електронних таблиць Microsoft Excel пропонується лабораторна

робота “Розв’язання задач лінійного програмування з використанням електронної таблиці Excel”. Як правило, задача лінійного програмування розв’язується симплексним методом вручну, поряд з цим студентам пропонується застосувати прикладні програми для отримання розв’язку задачі лінійного програмування. Одним із таких засобів є програмний засіб Microsoft Excel. На даній лабораторній роботі із застосуванням електронних таблиць Microsoft Excel студенти швидко опановують навички розв’язання задач математичного програмування із застосуванням надбудови Microsoft Excel “Поиск решения” і можуть за одну лабораторну роботу розв’язати більшу кількість задач математичного програмування, з використанням матеріалів стандартних звітів “Результаты” “Устойчивость”, “Пределы” зробити ґрунтовний економічний аналіз розв’язку.

Під час виконання лабораторних робіт закріплюється не лише поточний матеріал, а весь матеріал певного розділу або теми. На лабораторних роботах створюються сприятливі умови для розвитку самостійності, ініціативи, творчості, елементів дослідження.

За навчальними цілями можна виділити такі види лабораторних робіт: початкове ознайомлення з новими поняттями з певної теми, створення та дослідження математичних моделей. Завдання лабораторної роботи формулюється з урахуванням принципу рівневої диференціації так, щоб основну частину завдання змогли виконати всі студенти. Метою лабораторних занять є практичне засвоєння студентами науково-теоретичних положень дисципліни.

Таким чином, лабораторні роботи, як практичний метод навчання дають можливість закріпити теоретичні знання, глибше і наочно вивчати механізм застосування теоретичних знань, розвивати творчі здібності студентів, їхні дослідницькі інтереси, активізувати пізнавальну діяльність студентів, інтегрувати теоретико-методологічні знання, практичні навички і уміння в єдиному процесі діяльності навчально-дослідницького характеру.

Література

1. Морзе Н.В. Система методичної підготовки майбутніх вчителів інформатики в педагогічних університетах: Дис. ... докт. пед. наук: 13.00.02 / Національний педагогічний університет імені М.П. Драгоманова. – Київ, 2003. – 600 с.
2. Слєпкань З.І. Наукові засади педагогічного процесу у вищій школі. – К.: НПУ, 2000. – 210 с.
3. Іваськів І.С. Активізація навчально-пізнавальної діяльності учнів на основі систем штучного інтелекту при навчанні інформатики в старшій школі: Автореф. дис. ... канд. пед. наук. – К.: НПУ ім. М.П. Драгоманова, 2000. – 20 с.
4. Тєслєнко Г.С. Інформаційні системи в аграрному менеджменті. – К.: Урожай, 1999. – 256 с.
5. Комп'ютерні технології в управлінні сільськогосподарським підприємством. За ред. П.Т. Саблука, М.Ф. Кропивка, Ю.С. Коваленка. – К.: УААН ІАЕ, 1996. – 310 с.
6. Давыдов В.В. Теория развивающего обучения. – М.: ИНТОР, 1996. – 544 с.

СПЕЦИАЛИЗИРОВАННЫЙ АППАРАТНО-ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ ЧТЕНИЯ ЭЛЕКТРОННЫХ КНИГ

Е.А. Кожаев, Г.Г. Маклакова, Г.Ю. Маклаков
г. Севастополь, Севастопольский национальный технический
университет
gmkromab@sevgtu.sebastopol.ua

В настоящее время для повышения эффективности учебного процесса получают все большее распространение электронные учебники. От обычных учебников их отличает большая наглядность, четкая структурированность, возможность дозирования материала. Эти свойства электронных учебников способствуют успешному их использованию при изучении как специальных, так и общеобразовательных предметов. Электронные книги можно с успехом читать на обычных компьютерах, используя стандартное программное обеспечение Microsoft Office. Внедрение в учебный процесс электронных учебников во многом сдерживается из-за необходимости использовать устройства, способные воспроизводить текст с учетом физиологических и санитарных требований к отображению учебной информации на экране. Таким требованиям отвечают стационарные компьютеры, однако из-за их дороговизны использовать их только для чтения электронных книг является непозволительной роскошью не только в средней, но и высшей школе. Портативные компьютеры класса ноутбук (NoteBook) также достаточно дороги, и поэтому применяются в учебном процессе ограниченно не только в нашей стране, но и за рубежом. Следует заметить, что дороговизна современных компьютеров определяется в основном их универсальностью – они изначально ориентировались на решение самого широкого класса задач. В связи с этим представляется целесообразным разработать специализированный «компьютер», ориентированный только на отображение электронных книг. Идея создания специализированных устройств для чтения электронных книг возникла достаточно давно. В 1968 году студент Алан Кай разработал проект ДунаBook – портативный интерактивный компьютер, выполняющий роль книги [1]. Реализовать этот про-

ект удалось только через 20 лет. Серьезные коммерческие разработки электронных книг начались только с 1998 года. Анализ публикаций показал, что с тех пор созданы только 2 серьезные разработки [1]. Одна из разработок – SoftBook Reader. Устройство весит 1,3 кг, имеет размер чуть более обычной книги. Он оснащен монохромным LCD-монитором (6×8 дюймов), 8 МВ памяти, которой хватает на хранения примерно 5000 страниц. Стоимость устройства \$600. Другая разработка – Rocket-ebook [1]. Устройство весит 630 г, также имеет размер обычной книги. Он оснащен монохромным LCD-монитором (11,4×7,6 см). Стоимость устройства \$200. Особенностью данного устройства является то, что оно ориентировано только на специальный формат текстовых файлов, загружаемых из библиотеки RocketLibrarian через Интернет, другая особенность конструкции – необходимость использования стационарного компьютера, через который и происходит скачивание книг из библиотеки фирмы-разработчика в сети Интернет. В мае 2003 года в Интернет появилось сообщение (<http://news.ferra.ru/2003/05/05/39256/>), что в Китае фирма Matsushita Electric разрабатывает электронную книгу. Устройство имеет две «страницы», каждая из которых в действительности представляет собой 7,2-дюймовый жидкокристаллический дисплей. Разрешение каждого дисплея составляет 1024x768 пикселей. Тексты хранятся на флэш-картах Secure Digital. Устройство будет снабжено средствами защиты от нелегального копирования. Стоимость одной «электронной книги» производства Matsushita Electric составит около 250 долларов.

Общими недостатками описанных выше устройств являются высокая цена и отсутствие русифицированных версий. Для эффективного использования в учебном процессе современных устройств отображения информации они должны иметь высокие эксплуатационные характеристики при минимальной стоимости и высокой надежности.

Проведенный выше обзор показывает актуальность разработки портативных устройств для чтения электронных книг.

Данная работа выполнялась в рамках принятой концепции создания аппаратно-программных комплексов для мультимедийного отображения учебной информации (см. статью в настоящем сборнике: Маклаков Г.Ю., Кожаев Е.А., Маклакова Г.Г.

Перспективы использования современных микроконтроллеров для создания портативных устройств мультимедийного отображения учебной информации). Исходя из выбранной концепции, было принято решение, что устройство должно удовлетворять следующим требованиям:

- для хранения информации должна использоваться энерго-независимая память;
- для управления устройством необходимо использовать специализированный микропроцессор;
- устройство должно обеспечивать простоту обращения с ним (рассчитано на неквалифицированного пользователя);
- для отображения информации надо использовать жидкокристаллический дисплей;
- устройство должно быть простым и дешевым.

Исходя из поставленной задачи была разработана структурная схема устройства (см. рис.1).

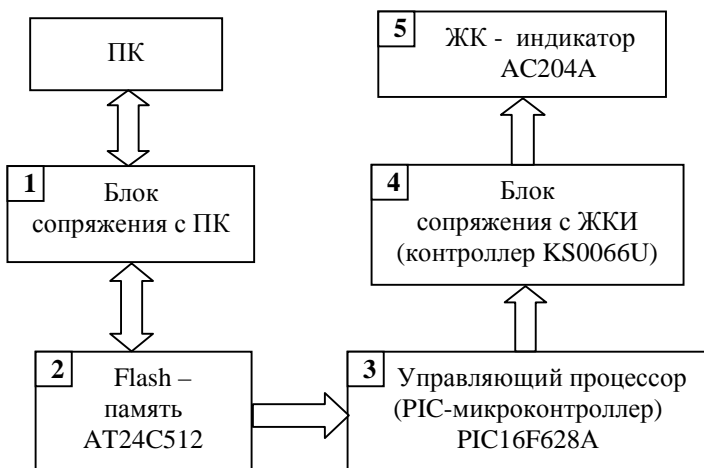


Рис. 1. Функциональная схема аппаратно-программного комплекса «электронная книга»

Блок 1 – блок сопряжения аппаратно-программного ком-

плекса с параллельным портом персонального компьютера. Блок предназначен для подготовки и перезаписи мультимедийной информации из базового компьютера в комплекс «электронная книга». Технически блок реализован на двух транзисторах. В качестве базовой ЭВМ используется персональный компьютер стандартной комплектации, функционирующий в операционной системе Windows 9*/XP.

Блок 2 представляет собой энергонезависимую Flash-память с последовательным интерфейсом I²C [3]. Блок служит для хранения информации, используемой комплексом «электронная книга». В данной конструкции используется микросхема AT24C512, обеспечивающая хранение 400-500 страниц машинописного текста. Предусмотрено использование микросхем Flash-памяти CMOS SEEPROM семейства AT24 с I²C последовательным интерфейсом разработки фирмы Atmel. Конструктивно Flash-память выполнена в виде съемного блока размером 35×15×15 мм.

Блок 3 – управляющий процессор на базе микроконтроллера PIC16F628A [4, 5]. Основное назначение блока – организовать работу «электронной книги» и обеспечить удобный интерфейс работы системы с пользователем. Для управления процессом просмотра книги предусмотрены 3 кнопки. Две из них служат для управления работой устройства (одна из них служит для прокрутки текста, другая – для перемещения курсора меню). Третья кнопка служит для вызова меню и подтверждения выбора его пунктов.

Для управления работой микроконтроллера, а значит и всего устройства в целом, было разработано системное программное обеспечение. Программы написаны на ассемблере MPASM [5]. MPASM используется для двух целей:

- генерации абсолютного кода, который является окончательным и предназначен для записи в микроконтроллер при помощи программатора;
- генерация объектного кода, который впоследствии связывается с другими ассемблированными или компилированными модулями.

Для отладки устройства использовалась интегрированная среда MPLAB фирмы Microchip Technology Incorporated [5].

Блок 4 – блок сопряжения. Обеспечивает вывод информации на дисплей (играет роль видеокарты и драйвера). Блок реализован на микроконтроллере KS0066U фирмы Samsung с русифицированным знакогенератором.

Блок 5 – устройство отображения (дисплей). В качестве устройства отображения используется жидкокристаллический индикатор (ЖКИ) AC204A фирмы Amprе [2]. ЖКИ обеспечивает вывод на экран размером 76×25 мм 4 строки по 20 символов.

Аппаратно-программный комплекс «электронная книга» выполнен в корпусе размером 120×100×25 мм, вес устройства около 150 грамм. Примерная себестоимость устройства – 80 грн. Следует отметить, что цель данной разработки было создать демонстрационный прототип, т.е. устройство, подтверждающее правильность выбранного подхода.

Для расширения функциональных возможностей устройства была создана вторая модификация аппаратно-программного комплекса «электронная книга». Его отличительная особенность – использование графического ЖКИ (индикатор AG240128F фирмы Amprе). Индикатор обеспечивает вывод 240×128 точек размером 0,5×0,5 мм на экран размером 148×75 мм. Функциональная схема почти такая же (отсутствует блок 4), как и в первой модификации, изменено только системное программное обеспечение.

Литература

1. Данилов О. От папирусов до мониторов // Домашний ПК. – 2000. – № 4. – С. 50–51.
2. Самарин А.В. Жидкокристаллические дисплеи. Схемотехника, конструкция и применение. – М.: Солон, 2002. – 304 с.
3. Семенов Б.Ю. Шина I²C в радиотехнических конструкциях. – М.: Солон, 2002.
4. Тавернье К. PIC-микроконтроллеры. Практика применения. – М.: ДМК Пресс, 2003. – 272 с.
5. Яценко В.С. Микроконтроллеры Microchip. Практическое руководство. – М.: Горячая линия-Телеком, 2002. – 296 с.

МЕТОД ТЕСТИРОВАНИЯ ЦИФРОВЫХ СИСТЕМ СО ВСТРОЕННОЙ ПАМЯТЬЮ

И.А. Колпаков, В.Г. Рябцев

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

В настоящее время для создания вычислительных систем широко используются сверхбольшие интегральные схемы (СБИС), содержащие встроенную память [1]. В таких СБИС на одном кристалле интегрированы микроконтроллер, запоминающие устройства и интерфейсы связи с внешними компонентами. Классификация цифровых систем, содержащих встроенную память, приведена на рис. 1.

Конфигурируемые на кристалле цифровые системы развиваются по двум основным направлениям: разработка специализированных микросхем (ASIC) и проектирование программируемых логических схем (ПЛИС).

Современные СБИС запоминающих устройств относятся к цифровым системам, содержащим встроенную память, т.к. содержат основные компоненты компьютера: встроенный микроконтроллер и скоростной интерфейс передачи данных. Например, для получения скорости передачи данных 1,6 Мбайт/с микросхемы памяти Rambus DRAM интегрированы с высокоскоростным логическим интерфейсом в модулях памяти RIMM (Rambus Interface Memory Modules), которые устанавливаются в системную плату так же как обычные SIMM или DIMM модули.

Однако при объединении логического устройства и микросхем памяти возникают новые и неизвестные механизмы ошибок, которые не присущи предыдущим поколения DRAM. Многие ошибки модулей RIMM обнаруживаются только при работе интерфейса на реальной рабочей частоте, а некоторые из них не выявляются даже при использовании режима прямого доступа к памяти. Такие отказы выявляются только при передаче тестовых воздействий для памяти через элементы логического интерфейса.

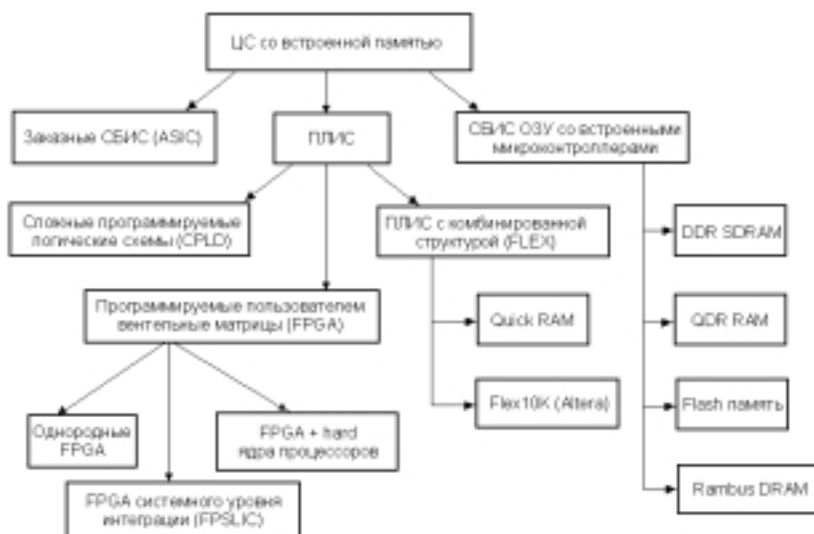


Рис. 1. Классификация цифровых систем со встроенной памятью

Детерминированные тестовые воздействия, сформированные программным способом, проверяют работоспособность только логического интерфейса, но не обеспечивают диагностирование массивов памяти. Чтобы обеспечить высокое качество диагностирования модулей нужно комбинировать тестовые детерминированные воздействия с тестовыми наборами, формируемыми алгоритмическим способом. Переключение направления передачи тестовых наборов от программных средств к аппаратным средствам формирования тестов следует осуществлять на реальной рабочей частоте.

В большинстве случаев объединение векторов, сформированных алгоритмически, с детерминированными тестовыми воздействиями обеспечивается дополнительными коммутационными схемами или за счет применения сложных микропроцессоров для каждого контакта подключения объекта диагноза, что увеличивает трудоемкость синтеза программ тестов и стоимость диагностического оборудования. Для объединения тестовых векторов необходимо создать новый метод, сочетающий высокое быстродействие с небольшими аппаратными затратами.

Для выполнения тестового диагностирования ЦС, которые содержат встроенную память, целесообразно применять диагностический комплекс, содержащий блок связи с ПЭВМ, генератор тактовых импульсов, многофазный генератор синхросигналов, блок микропрограммного управления, алгоритмический генератор (АЛГ), формирователь детерминированных тестов (ФДТ), который состоит из узла идентификации, адресного контроллера, памяти каналов, блока поразрядного сравнения данных, коммутатора выходных данных и адаптера [2].

В каждом канале предусмотрено три ОЗУ: ОЗУС, ОЗУМ, ОЗУВ, которые соответственно хранят воздействие или эталонные реакции обеспечивают управление буферным элементом и разрешают поразрядное сравнение реакции объекта контроля с эталонными сигналами [3].

Тестовые наборы из АЛГ или ФДТ поочередно заносятся в регистр тестовых наборов, а затем записываются в буферный блок памяти, при этом учитываются параметры векторов, которые предварительно заносятся в ОЗУ параметров векторов.

Тестовый набор V_f , который формируется ФДТ, задается в виде кортежа векторов:

$$V_f = \langle L_m^f, D_m^f, M_f, L_b^f, D_b^f, V_f, L_c^f, D_c^f, C_f \rangle,$$

где L_m^f, L_b^f, L_c^f - начальные номера элементов векторов, предназначенных для занесения в ОЗУМ, ОЗУВ, ОЗУС соответственно;

D_m^f, D_b^f, D_c^f - размерности векторов;

M_f, V_f, C_f - вектора тестового набора, формируемого ФДТ.

Тестовый набор V_g , который формируется АЛГ, задается в виде кортежа векторов:

$$V_g = \langle L_m^g, D_m^g, M_g, L_b^g, D_b^g, V_g, L_c^g, D_c^g, C_g \rangle,$$

где L_m^g, L_b^g, L_c^g - начальные номера элементов векторов, предназначенных для занесения в ОЗУМ, ОЗУВ, ОЗУС соответственно;

D_m^g, D_b^g, D_c^g – размерности векторов, формируемых АЛГ;
 M_g, B_g, C_g – вектора тестового набора, формируемого АЛГ.

Содержимое блоков памяти каналов определяется при помощи выражений:

$$\forall k \begin{bmatrix} M_m(A_q + 2k) \\ M_m(A_q + 2k + 1) \end{bmatrix} = M_f^k \oplus M_g^k;$$

$$\forall k \begin{bmatrix} M_b(A_q + 2k) \\ M_b(A_q + 2k + 1) \end{bmatrix} = B_f^k \oplus B_g^k;$$

$$\forall k \begin{bmatrix} M_c(A_q + 2k) \\ M_c(A_q + 2k + 1) \end{bmatrix} = C_f^k \oplus C_g^k;$$

где A_q – адрес начального воздействия;

k – номер текущего такта диагностирования;

$2t$ – общее число тестовых наборов, которые формируются ФДТ и АЛГ;

\oplus – операция присоединения векторов.

Если для встроенной памяти используются две разделенные шины для приема и передачи данных, то АЛГ формирует следующий кортеж векторов:

Операции присоединения векторов выполняются следующим образом:

$$\forall k, j [mm_j(A_q + 2k) := (m_i)_f^k]; i = 0, D_m^f - 1; j = \text{idt}(L_m^f), \text{idt}(L_m^f + D_m^f - 1);$$

$$\forall k, j [mm_j(A_q + 2k + 1) := (m_i)_g^k]; i = 0, D_m^g - 1; j = \text{idt}(L_m^g), \text{idt}(L_m^g + D_m^g - 1);$$

$$\forall k, j [mb_j(A_q + 2k) := (b_i)_f^k], i = 0, D_b^f - 1; j = \text{idt}(L_b^f), \text{idt}(L_b^f + D_b^f - 1);$$

$$\forall k, j [mb_j(A_q + 2k + 1) := (b_i)_g^k], i = 0, D_b^g - 1; j = \text{idt}(L_b^g), \text{idt}(L_b^g + D_b^g - 1);$$

$$\forall k, j [mc_j(A_q + 2k) := (c_i)_f^k], i = 0, D_c^f - 1; j = \text{idt}(L_c^f), \text{idt}(L_c^f + D_c^f - 1);$$

$$\forall k, j [mc_j(A_q + 2k + 1) := (c_i)_g^k], i = 0, D_c^g - 1; j = \text{idt}(L_c^g), \text{idt}(L_c^g + D_c^g - 1);$$

где $(m_i)_f^k \in M_f^k$; $(m_i)_g^k \in M_g^k$; $(b_i)_f^k \in B_f^k$; $(b_i)_g^k \in B_g^k$;
 $(c_i)_f^k \in C_f^k$; $(c_i)_g^k \in C_g^k$;

idt – операция идентификации логического и физического сечений объекта диагностирования, выполняется по таблице соответствий координат векторов и номеров физических контактов диагностируемого изделия.

При реализации данного метода объединения тестовых векторов, сформированных различными устройствами, требуются небольшие дополнительные аппаратные затраты, что уменьшает стоимость диагностируемых систем при одновременном повышении качества диагностируемых изделий.

Литература

1. Таненбаум Э. Архитектура компьютера. – С.-П.: Питер, 2002. – 704 с.
2. Рябцев В.Г. Особенности структуры комплекса диагностирования цифровых систем, содержащих встроенную память // Вісник ТУП. – 2003. – №3. – С. 26-29.
3. Устройство для тестового контроля цифровых блоков. А.с. №1251084 СССР, МКИ G06F 11/26. / А.А. Борисенко, В.Г. Рябцев, А.Д. Стафеев, В.А. Чернышев, А.Ф. Шамарин (СССР). – №3832978/24-24. Заявлено 29.12.84: Оpubл. 15.08.86. Бюл. № 38.– 12 с.

ВИКОРИСТАННЯ РЕЙТИНГОВОЇ СИСТЕМИ КОНТРОЛЮ ЗНАНЬ, УМІНЬ ТА НАВИЧОК СТУДЕНТІВ ПРИ НАВЧАННІ ІНФОРМАТИКИ

Ю.М. Красюк

м. Київ, Київський національний економічний університет
krasyuk_y@ukr.net

Досвід викладання курсу інформатики в вищих закладах освіти економічного профілю вказує на те, що використання традиційної системи контролю знань, умінь та навичок студентів не забезпечує управління мотивацією більшості студентів на належному рівні, а також систематичності стимулювання активної навчально-пізнавальної діяльності студентів та розвиток її саморегуляції. Ми спробували вирішити дану проблему через застосування модульно-рейтингової технології навчання із використанням такої системи контролю, яка стимулює навчально-пізнавальну діяльність студентів, сприяє розвитку їх самостійності та приводить до підвищення рівня якості загальної підготовки.

Аналіз застосування систем рейтингового контролю навчально-пізнавальної діяльності студентів у вітчизняних та зарубіжних педагогічних дослідженнях дає можливість виділити чотири основні моделі таких систем [2, с. 10]. У першій моделі основою підрахунку рейтингу кожного студента є щотижневі заліки, результати яких додаються в кінці семестру. Друга модель використовує систему залікових годин (залікових одиниць), які студенти набирають протягом вивчення навчальної дисципліни. Однак дані моделі не відповідають модульній побудові курсу, тому ми не змогли їх використати. Третя модель системи рейтингового контролю передбачає проведення рейтингових контрольних робіт після вивчення кожного навчального модуля. Сума балів, які студент отримав за відповідні контрольні роботи, визначає його підсумкову рейтингову оцінку. Використання даної моделі в процесі навчання недостатньо активізує навчально-пізнавальну діяльність студента протягом вивчення навчального модуля, не дозволяє стимулювати процес його особистісно-професійного самостворення. Четверта модель полягає у тому, що всі ре-

зультати на етапах поточного, тематичного, підсумкового контролю знань, умінь та навичок студентів оцінюються в рейтингових балах та додаються від етапу до етапу (саме таким чином формується рейтинг студента).

Ми зупинилися на четвертій моделі рейтингової системи контролю, доповнивши її вхідними тестами на початку вивчення навчального модуля, творчими інтегрованими завданнями та призовим фондом, що вводиться викладачем для заохочення активності, самостійності та ініціативності студентів, а також системою вилучення рейтингових балів за несвочасне виконання завдань. Таким чином в основу рейтингового контролю нами була покладена система накопичувального типу, в якій рейтинг студента визначається як сума набраних балів за визначений період за результатами всіх форм занять та всіх видів контролю. Дана сума виступає в ролі числового показника якості роботи студента в порівнянні з максимально можливою кількістю рейтингових балів і з успіхами товаришів по навчанню.

Комплексне впровадження рейтингової системи контролю та оцінювання знань, умінь та навичок студентів ми здійснювали відповідно до представлених у табл. 1 етапів.

Таблиця 1.

Основні етапи та види діяльності	Мета
1	2
<i>Підготовчий етап</i>	
<i>Ранжування навчальних елементів та визначення контрольних точок перевірки знань, умінь та навичок студентів</i>	– визначити освітні вимоги предмету, нормативну основу навчальної дисципліни (освітні стандарти) та спроектувати їх на кожний навчальний елемент, вивчення якого перевіряється та оцінюється.
<i>Вибір методів та засобів контролю знань, умінь та навичок студентів</i>	– вибрати методи та засоби контролю, що забезпечують якісний та об'єктивний контроль результатів навчально-пізнавальної діяльності студентів згідно з її специфікою та відповідно до принципів індивідуалізації та диференціації.

1	2
нтів з особливостями системи контролю та оцінювання	тами щодо доцільності впровадження диференціації навчання, переваг запропонованої модульної системи та рейтингового контролю й оцінювання знань з курсу інформатики; – поінформувати студентів про зміст знань, умінь та навичок, які еквівалентні тій чи іншій якісній та кількісній оцінці.
Перевірка результатів роботи студентів з додатковими навчальними елементами	– визначити рівень готовності студентів, яким потрібно було попрацювати з додатковими навчальними елементами, до учіння.
Проведення поточного та тематичного контролю знань, умінь та навичок студентів	– визначити рівень засвоєння знань та сформованості умінь й навичок студентів з відповідних тем навчального модуля; – проаналізувати отримані результати; – за необхідності визначити навчальні впливи для їх корекції; – при необхідності аргументовано пояснити студентам результати оцінювання за будь-який вид роботи, що була ним виконана (зіставити реальні знання із визначеними стандартами і змістом кожної якісної та кількісної оцінки).
Перевірка розв’язків творчих завдань	– перевірити сформованість умінь використовувати засвоєні знання та набуті уміння й навички до розв’язування творчих завдань та вирішення нестандартних виробничих ситуацій.
<i>Етап оцінювання</i>	
Підведення підсумків навчальної діяльності студентів за семестр	– розрахувати рейтинг кожного студента як суму набраних рейтингових балів за визначений період за результатами всіх форм занять та всіх видів контролю; – проаналізувати отримані результати та зробити висновки щодо можливостей удо-

1	2
	сконалення рейтингової системи контролю та оцінювання.

При цьому ми дотримувалися наступних принципів:

- об’єктивність контролю та висока точність вимірювання результатів навчально-пізнавальної діяльності студентів;
- інформативність та адекватність контролю;
- швидкодія оцінювання знань, умінь та навичок студентів;
- систематичність та всебічність контролю;
- забезпечення зворотного зв’язку у процесі навчання;
- можливість порівняння результатів навчально-пізнавальної діяльності студентів на різних етапах навчання;
- педагогічно доцільна автоматизація контролю знань, умінь та навичок студентів;
- несуттєвий вплив на студентів засобів вимірювання результатів навчально-пізнавальної діяльності;
- умотивованість навчально-пізнавальної діяльності;
- індивідуалізація та диференціація навчання.

Кожне навчальне завдання оцінюється за багатобальною шкалою, що замінює традиційну чотирибальну шкалу. Введення її надало змогу викладачам більш точно оцінювати навчальні досягнення студентів, що сприяло збільшенню вагомості стимулюючої функції контролю. Систематичний розрахунок поточного рейтингу кожного студента та можливість його порівняти з рейтингами своїх однокурсників створює позитивний характер в навчальній змагальності.

Водночас ми звернули увагу на психологічну підготовку студентів до проведення різних видів контролю. Адже не потрібно забувати, що ефективність довільного виду контролю у великій мірі зумовлюється готовністю студента до його проведення. З цією метою ще на початку вивчення курсу студентам пояснюється доцільність, мета та переваги проведення систематичного наскрізного контролю протягом всього семестру. При цьому кожна група отримує інформаційний комплект по організації їх навчальної діяльності протягом семестру (дані матеріали також розміщуються на навчальному Web-сайті). Він включає: тематичний план дисципліни, зміст дисципліни за модулями, практичні заняття, лабораторні роботи, завдання для самостійної роботи,

індивідуальні творчі завдання, особливості системи поточного контролю (форми контролю по навчальних тижнях і критерії оцінювання знань, умінь та навичок студентів) та рекомендовану літературу. Така поінформованість студентів, систематичне надання можливості кожному з них провести самоконтроль і самокорекцію власних знань та навичок за допомогою автоматизованої системи контролю знань ТЕСТ [3] та прозорість системи підсумкового оцінювання сприяє зменшенню рівня тривожності у більшості студентів, розвитку умінь самоконтролю та самокорекції, формуванню адекватності самооцінки [1, с. 12-13].

В кінці семестру підраховується підсумковий рейтинг студента, який служить основою для визначення за встановленою шкалою загальної оцінки з навчального курсу. В цій шкалі бажано передбачити:

- суму балів за роботу по вивченню всього курсу (певної частини курсу), яка дає право автоматичного допуску до екзамену (отримання залікової оцінки) без складання заліку;
- мінімальну суму балів для отримання допуску до складання заліку;
- суму балів, яка надає право на ліквідацію заборгованості;
- суму балів, яка вимагає як мінімум повторного вивчення обов'язкового курсу.

Студент “автоматично” отримує залік (в 1 семестрі) та допуск до іспиту (в 2 семестрі), якщо він досягнув мінімально-базового рівня знань – виконав та захистив усі поставлені перед ним завдання, відвідував лекції, виявляв активність у співпраці з викладачем та іншими студентами і при цьому набрав не менше 200 рейтингових балів (половину від максимально кількості рейтингових балів, яку може набрати студент під час роботи в семестрі, табл. 2).

Таблиця 2.

	<i>Форма контролю</i>	1 семестр	2 семестр
<i>Поточний контроль</i>	Практичні заняття	80	80
	Оперативні тести	90	70
	Творчі завдання	60	55
<i>Тематичний контроль</i>	Лабораторні роботи	60	90
	Контрольні тести	45	60
	Контрольні роботи	60	40

	<i>Форма контролю</i>	1 семестр	2 семестр
	Призовий фонд	5	5
	Всього	400	400

Якщо студент виконав та захистив усі поставлені перед ним в поточному семестрі навчальні завдання, однак набрав від 175 до 199 рейтингових балів, то він отримує можливість скласти залік, за результатами якого буде визначатися його залікова оцінка (зараховано – не зараховано).

Якщо студент на кінець семестру не виконав всіх навчальних завдань (з поважних причин) і при цьому набрав не менше 120 рейтингових балів, то він отримує можливість за встановлений термін часу ліквідувати свою академічну заборгованість та скласти залік.

Якщо студент під час роботи в поточному семестрі виконав всі завдання та набрав менше 120 рейтингових балів, або не виконав всіх навчальних завдань без поважних причин, то йому виставляється залікова оцінка “не зараховано”.

Тепер розглянемо детальніше систему підсумкового оцінювання в другому семестрі. На приклад, студент Іванов під час роботи в семестрі набрав 295 рейтингових балів, які переводяться в 30 балів (табл. 3). Саме з цією кількістю балів студент допускається до екзамену. На іспиті студент отримує білет з шістьма завданнями, за правильне розв’язування яких може заробити 60 балів (правильна відповідь на кожне завдання оцінюється в 10 балів; розв’язування завдання з не досить значними помилками – в 5 балів; розв’язок завдання з значними помилками або його відсутність – в 0 балів). Екзаменаційна робота студента вважається зарахованою, якщо студент отримав не менше 30 балів (половину від максимально можливої кількості балів).

Таблиця 3

Отримана кількість рейтингових балів	Кількість балів після перерахунку
351-400	40
301-350	35
275-300	30
235-274	25
200-234	20
175-199	мінімальна сума балів для отримання до-

Отримана кількість рейтингових балів	Кількість балів після перерахунку
	пуску до складання заліку
120-174	сума балів, яка надає право на ліквідацію заборгованості
менше 120	сума балів, яка вимагає повторного вивчення навчального курсу

Продовжимо розгляд прикладу з студентом Івановим. Припустимо, що він на екзамені отримує 40 балів. Для підсумкового оцінювання навчальної діяльності студента у другому семестрі залишається додати 30 балів за роботу в семестрі та 40 балів за екзаменаційну роботу (максимально він міг би набрати 100 балів: 40 балів за роботу в семестрі та 60 балів за екзаменаційну роботу).

Отриманий результат переводиться в традиційну чотирих-бальну систему за наступною схемою: «2» – набрана студентом сума балів менше 50 рейтингових балів; «3» – студент набрав від 50 до 64 балів; «4» – від 64 до 84 балів; «5» – від 85 до 100 балів. А це означає, що студент Іванов, який загалом за другий семестр набрав 70 балів, отримує четвірку за традиційною системою оцінювання.

Запровадження розглянутої системи рейтингового контролю знань, умінь та навичок студентів при навчанні інформатики створює ситуацію значущості всього процесу перевірки й оцінювання знань. Також змінюється напрямок діяльності студента – із “спрямованості на себе” на “спрямованість на справу” [1, с. 12-13]. Студенти починають сприймати контроль перш за все не як “випробування на підготовленість з тої чи іншої теми” чи “покарання за невивчений матеріал”, а як компетентну допомогу в їх навчально-пізнавальній діяльності, що надасть можливість виявити неточності та прогалини в знаннях та вміннях, вчасно їх виправити, що пізніше надасть можливість стати більш кваліфікованим фахівцем.

Література

1. Бочарнікова В.М. Стимулююча функція контролю знань, умінь і навичок студентів вищої школи: Автореф. дис. ... канд. пед. наук: 13.00.01 / НПУ ім. М.П. Драгоманова. – К., 1999. – 20 с.
2. Шиян Н.І. Технологія модульно-рейтингового навчання у вищій педагогічній школі: Автореф. дис. ... канд. пед. наук: 13.00.01 / Харк. держ. пед. ун-т ім. Г.С. Сковороди. – Х., 1999. – 18 с.
3. Красюк Ю.М., Сидорчук В.В. Загальна характеристика автоматизованої системи контролю знань ТЕСТ // Зб. наук. пр. Міжнар. науково-практичної конф. “Проблеми впровадження інформаційних технологій в економіці та бізнесі”. – Ірпінь: Академія ДПС України. – 2002.

**ТЕОРЕТИЧНІ ТА ПРАКТИЧНІ ЗАСАДИ
ВПРОВАДЖЕННЯ ПЕДАГОГІЧНОЇ ТЕХНОЛОГІЇ
“СТВОРЕННЯ СИТУАЦІЇ УСПІХУ”
В ПРОЦЕС ВИКЛАДАННЯ ДИСЦИПЛІНИ
“ІНФОРМАТИКА ТА КОМП’ЮТЕРНА ТЕХНІКА”
У ВИЩОМУ НАВЧАЛЬНОМУ ЗАКЛАДІ**

І.В. Крашеніннік

м. Мелітополь, Економіко-гуманітарний факультет Таврійського
національного університету ім. В.І. Вернадського
kr_ira@rambler.ru

Сучасний етап розвитку українського суспільства характеризується значним зростанням інформаційного навантаження на людину. Інтенсифікація суспільно-економічних процесів висуває нові вимоги до системи освіти. Освічена людина має володіти не стільки фактологічними знаннями, які в багатьох сферах дуже швидко втрачають актуальність, скільки технологічними вміннями здобувати нові знання, застосовувати їх в стандартних та нестандартних ситуаціях.

Особливі вимоги суспільство ставить до вищої освіти. Кваліфікований спеціаліст повинен вміти самостійно працювати, приймати обґрунтовані рішення, розв’язувати проблеми. Такий підхід створює додаткові труднощі для викладачів вищої школи, пов’язані з тим, що випускники середніх шкіл у переважній більшості своїй психологічно не підготовані та не мають відповідних навичок для самостійної пошукової навчальної праці. Вища школа не може чекати, коли відбудуться докорінні зміни в навчально-виховному процесі школи середньої. Серед стратегічних напрямків подолання цих проблем можна назвати гуманізацію освіти та технологічний підхід, головним завданням яких є розвиток особистості людини: і студента, і викладача.

Гуманізація передбачає розвиток системи особистісно орієнтованої освіти. Основою процесу гуманізації можна вважати гуманістичну психологію (А. Маслоу, Д. Олпорт, К. Роджерс). Проблеми гуманізації розробляли визначні вчені світу (К. Гольдштейн, Р. Мей, Е. Фром, К. Хорні, В. Франкл, Р. Бернс та ін.). Систему особистісно орієнтованої освіти досліджували такі ви-

значні психологи, як К.О. Абульханова-Славська, В.І. Андреев, О.Г. Асмолов, В.В. Давидов, Л.М. Деркач, В.О. Моляко, Л.М. Проколієнко, В.В. Рибалка, М.І. Романенко, В.В. Сериков, В.В. Столін, В.О. Татенко, Т.М. Титаренко, І.С. Якиманська та інші.

Відповідно до сучасного розуміння питання гуманізації освіти “особистісний підхід доцільно розглядати як важливий психолого-педагогічний принцип, як методологічний інструментарій, основу якого становить сукупність вихідних концептуальних уявлень, цільових установок, методико-психодіагностичних та психолого-технологічних засобів, які забезпечують більш глибоке цілісне розуміння, пізнання особистості дитини і на цій основі – її гармонійний розвиток в умовах існуючої освітньої системи” [4, с. 30]. О.М. Пехота зазначає, що є “тільки один спосіб реалізувати особистісний підхід у навчанні – зробити навчання сферою самоствердження особистості” [4, с. 31]. Як резюмує Троїцька Т.С., гуманізація освіти знаходить відображення у максимальній індивідуалізації і диференціації навчання, що передбачає зміну статусу студента: він з об’єкту, який пасивно сприймає інформацію, стає суб’єктом, активним учасником навчального процесу [5, с. 292].

Об’єктивним явищем сьогодення став процес технологізації освіти, який полягає у визначенні напрямків та способів взаємодії суб’єктів навчального процесу. Початок впровадженню педагогічних технологій в освіту було покладено в 20-і – 30-і роки минулого століття видатним вітчизняним педагогом А.С. Макаренком і згодом продовжено зарубіжними та вітчизняними вченими (А.А. Ухтомський, С.Т. Шацький, Т.А. Ільїна, А.І. Космодем’янська, М.В. Кларин, І.Я. Лернер, В.П. Безпалько, М. Кларк, Ф. Персиваль, Г. Еллінгтон, М. Вулман, С. Сполдинг, Д. Фінн та ін). О.М. Пехота визначає чотири етапи в еволюції педагогічної технології, початок якої пов’язаний з активним впровадженням в навчання технічних засобів. Треба відзначити, що це поняття значною мірою є категорією практики, тому не має загальноприйнятого його визначення. П.Д. Мітчелл на основі аналізу багатьох джерел зробив такий висновок: “Педагогічна технологія є галузь досліджень і практики (у межах системи освіти), що має зв’язки (відносини) з усіма аспектами організації педагогічних

систем і процедурою розподілу ресурсів для досягнення специфічних і потенційно відтворюваних педагогічних результатів” [4, с. 22].

О.М. Пехота зазначає, що “педагогічна технологія в загальнопедагогічному розумінні характеризує цілісний освітній процес з його метою, змістом і методами навчання. Окремо предметна педагогічна технологія – сукупність методів і засобів для реалізації визначеного змісту навчання в межах одного предмета (методика викладання предмета). Локальна ж технологія являє собою вирішення окремих дидактичних і виховних завдань” [4, с. 24].

Поєднання гуманістичного та технологічного підходів, яке реалізується через впровадження особистісно орієнтованих педагогічних технологій, є перспективним напрямком подолання недоліків в системі освіти. Ці технології мають відповідати певним вимогам, серед яких І.С. Якиманська називає наступні: виявлення змісту суб’єктного досвіду учня, його перетворення, узгодження з науковим змістом здобутих знань, стимулювання учня до самоцінної освітньої діяльності, контроль і оцінка не тільки результату, а й головним чином процесу учіння [4, с. 36-37]. В.М. Вакуленко вказує такі ознаки технології навчання: мета, організованість і управління процесу навчання, етапність технологізованого процесу, алгоритмічність багатьох процедур, координованість і раціональність виконуваних дій, система параметрів навчального процесу та його результатів, стандартизованість, наявність функціональних характеристик [2, с. 62-63].

Дослідження щодо впровадження педагогічних технологій здебільшого стосуються середньої школи, але відомі українські дослідники (А.М. Алексюк, В.І. Бондар, Я.І. Бурлака, І.А. Зязюн, Н.Г. Ничкало, М.М. Окса, М.Д. Ярмаченко) вказують, що є необхідність технологізації навчального процесу у вищій школі.

Нами було проведено соціологічне дослідження з метою визначення ставлення студентів першого курсу до вищої освіти та власної навчальної діяльності. Аналіз відповідей свідчить, що для студентів досить велике значення має успіх у навчанні. Ми зацікавилися, що є критерієм цього успіху для студентів. Відповіді розподілилися наступним чином: для 58% студентів таким критерієм є оцінка викладача, для 20% – оцінка батьків, для 15%

– оцінка товаришів, і лише для 7% – власна оцінка. Такий результат свідчить про недостатній рівень суб'єктності знань, отриманих у школі і відповідно низький ступінь готовності до самостійної освітньої діяльності.

Слід також відзначити, що більшість студентів першого курсу (60%) не пов'язують успіх у навчанні із подальшою успішною професійною діяльністю. Такий результат, на нашу думку, викликаний тим, що навчання у школі мало здебільшого фактологічний характер. Позитивним моментом є той факт, що 43% опитуваних вважають, що головним завданням вищої освіти є розвиток творчого підходу до діяльності, формування вмінь приймати рішення, а також готові до самостійної праці.

Назвемо головні, на нашу думку, причини адаптації особистісно орієнтованих технологій навчання для процесу підготовки фахівців у вищих навчальних закладах:

- активізація творчої, пошукової роботи студентів;
- переведення нових знань у суб'єктний досвід студентів є основою для розвитку свідомого ставлення до власної освіти і майбутньої професії;
- стимулювання активної діяльності студентів сприяє формуванню спеціаліста з високим професійним потенціалом;
- реалізація гуманістичної парадигми дозволяє розвинути у студентів розуміння особистої гідності, повагу до інших людей.

Однією з перспективних педагогічних технологій, на нашу думку, є технологія “Створення ситуації успіху”, яка передбачає таку організацію навчально-виховного процесу, за якої людина відчуває радість від самого навчання, а також своїх успіхів. Ця технологія бере початок з діяльності видатних педагогів А.С. Макаренка та В.О. Сухомлинського. За А.С. Белкіним ситуацію успіху можна розглядати з різних позицій: педагогічної, психологічної та соціально-психологічної.

Для учня або студента успіх – це переживання стану радості, задоволення від того, що результат, до якого особистість прямувала у своїй діяльності, або збігся з її очікуваннями, сподіваннями (з рівнем домагань), або перевершив їх [1, с. 31].

Для педагога, викладача успіх – це таке цілеспрямоване, організоване поєднання умов, за яких створюється можливість до-

сягти значних результатів у діяльності як окремо взятої особистості, так і колективу в цілому [1, с. 31].

Вказана педагогічна технологія передбачає розподіл учнів на групи: надійні, впевнені, невпевнені, зневірені. О.Є. Олексюк дає таку характеристику цих груп [4, с. 205]:

- надійні: мають добрі здібності, сумлінно ставляться до обов'язків, привчені до самостійності, впевнені в собі, рівень домагань адекватний можливостям;
- впевнені: здібності можуть бути і вищими, ніж у “надійних”, але система роботи не настільки відлагоджена, періоди підйому змінюються розслабленням, можливі збої у роботі, швидке звикання до успіхів, переростання впевненості у самовпевненість;
- невпевнені: мають добрі здібності, відповідально ставляться до справи, але головна риса – невпевненість у своїх силах;
- зневірені: мають непогані здібності, але після відчуття радості сподівань втратили її через різні причини.

В процесі реалізації технології створюються ситуації, у яких учні переживають такі різновиди радості: здійснена, неочікувана, загальна, сімейна, радість пізнання.

Сучасні процеси інформатизації та комп'ютеризації висувають нові вимоги до вищої освіти, а саме: розвиток загальної інформаційної культури майбутніх спеціалістів, а також, як компонент цієї культури, оволодіння навичками використання комп'ютерної техніки в професійній діяльності. Ці вимоги покликана виконати дисципліна “Інформатика та комп'ютерна техніка”, яка включена до навчальних планів усіх спеціальностей.

На основі вказаних вище теоретичних положень, проводиться робота із впровадження педагогічної технології “Створення ситуації успіху” у навчальний процес вищого навчального закладу під час вивчення дисципліни “Інформатика та комп'ютерна техніка”. Основною метою цієї роботи є формування у студентів стійкого відчуття задоволення від своєї успішної навчальної діяльності засобами спеціально створених ситуацій.

Впровадження даної технології проходить такі етапи:

1. Діагностика початкового ступеня впевненості студентів у власних силах, важливості для них навчальних успіхів та розподіл студентів за групами. Відзначимо, що на пер-

шому курсі не виділяється група зневірених.

2. Створення ситуацій для відчуття студентами здійсненої радості через використання ситуацій, запропонованих О.Є. Олексюком [4]:

- “Невтручання” – створення режиму максимальної самостійності для студентів категорій “надійні” та “впевнені”;
- “Холодний душ” – часткове “охолодження” підвищеної радості “впевненого” студента вимогою максимально об’єктивної перевірки результатів своєї роботи та підвищення її якості;
- “Емоційне поглажування” – визнання та заохочення будь-яких, навіть незначних, досягнень “невпевненого” студента;
- “Анонсування” – попереднє визначення діяльності “невпевненого” студента, яка має привести до успіху.

Додатково для “надійних” та “впевнених” студентів введена ситуація “Допоможи іншому”, коли заохочується спільна пізнавальна діяльність. Її можна використати також для “невпевнених” студентів, за умови їх завчасного попередження та підготовки.

3. Формування постійної радості пізнання. Для цього використовувалися наступні ситуації, запропоновані О.Є. Олексюком [4]:

- “Еврика” – активізація студентів до самостійної творчої та пошукової діяльності із заохоченням будь-яких зроблених відкриттів;
- “Навмисна помилка” – виправлення помилок на основі за-своєних знань.

З метою формування цього виду радості також використовуються ситуації, описані О.К. Колеченком [3]:

- “Визначення понять” – застосовується під час вивчення нового матеріалу, студенти повинні за допомогою різних засобів визначити поняття, які є для них невідомими;
- “Мнемотурнір” – використовується під час опитування і передбачає самостійний, взаємний та експертний контроль знань студентів із визначенням оцінки та ступеню впевненості у власних знаннях;
- “Опитування п’ятьох” – передбачає одночасне опитування декількох обраних студентів їхніми товаришами, дозволяє

знизити стресову ситуацію під час контролю знань.

Проведена робота з використання технології “Створення ситуації успіху” під час вивчення дисципліни “Інформатика та комп’ютерна техніка” дозволяє зробити наступні висновки:

1. Вказана педагогічна технологія сприяє розвитку у студентів пізнавальної активності, а також відчуття задоволення від виконаної діяльності.
2. Ситуації, які створюються для забезпечення загального успіху студентів, дозволяють подолати можливі стресові ситуації.
3. Дана технологія відповідає парадигмі гуманізації навчального процесу.

Література:

1. Белкин А.С. Ситуация успеха. Как ее создать: Кн. для учителя. – М.: Просвещение, 1991. – 176 с.
2. Вакуленко В.М. Інноваційні технології навчання як засіб підвищення якості підготовки педагогів // Проблеми сучасної педагогічної освіти. Сер.: Педагогіка і психологія. – Зб. статей: Вип. 5. – К.: Пед. преса. – 2003. – С. 59-65.
3. Колеченко А.К. Энциклопедия педагогических технологий: Пособие для преподавателей. – СПб.: КАРО, 2002. – 368 с.
4. Освітні технології: Навч.-метод. посіб. / О.М. Пехота, А.З. Кіктенко, О.М. Любарська та ін.; За заг. ред. О.М. Пехоти. – К.: А.С.К., 2002. – 255 с.
5. Троїцька Т.С. Методологічні засади гуманістичної освітнянської практики // Проблеми сучасної педагогічної освіти. Сер.: Педагогіка і психологія. – Зб. статей: Вип. 5. – К.: Пед. преса. – 2003. – С. 289-293.

НЕКОТОРЫЕ АСПЕКТЫ ПРЕПОДАВАНИЯ ОСНОВ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

И.А. Кривель¹, А.Н. Моргун²

¹ г. Черкассы, Черкасский национальный университет
имени Богдана Хмельницкого

² г. Черкассы, Черкасский институт пожарной безопасности
имени Героев Чернобыля
an_m@ukr.net

Учитывая современные тенденции и возрастающую роль изучения информатики и информационных технологий компьютерными специальностями ВУЗов, необходимо критически подойти к сложившейся системе преподаваемых знаний. С этих позиций, в первую очередь, требует пересмотра содержание разделов информатики, связанных с алгоритмизацией и программированием.

Не подвергая сомнению необходимость изучения упомянутых разделов, как таковых, следует отметить, однако, наличие в них явно устаревших и исчерпавших себя элементов.

До сих пор, следуя традициям, в качестве средства алгоритмизации решаемых задач предлагается, а точнее сказать, навязывается, использование блок-схем. Справедливости ради отметим, что в своё время их применение было оправдано существующей практикой отделения процессов построения алгоритмов от процессов подготовки программ решения задач [1]. Ведущий разработчик проекта составлял блок-схему программы, в соответствии с которой закреплённый технический работник писал текст программы (например, на Фортране), наносил его на перфокарты и осуществлял связь с вычислительным центром. В современных же условиях применения персональных компьютеров разделять эти процессы стало совершенно бессмысленным. Программист, рисуемый блок-схемы, сидя за клавиатурой (а также и в других местах), выглядит несколько несерьёзно.

С целью предварительной алгоритмизации процессов решения задач известные эффективные методики обучения программированию вместо блок-схем предусматривают применение специальных языковых средств. На их базе удаётся реализовать

практически формальный переход к подпрограммам используемого языка программирования [1]. Благодаря этому учащимся прививаются навыки владения технологией структурного программирования, что немыслимо без широкомасштабного применения подпрограмм. В этом плане авторам очень нравится программа по информатике и информационным технологиям для профессионально-технических учебных заведений [2], с которой каждый может ознакомиться на официальном сайте Министерства образования и науки Украины (www.mon.gov.ua). Можете себе представить, что в этой программе изучение процедур и функций предусмотрено ранее изучения операторов ветвления и цикла. Вместе с тем в практике обучения как в вузах, так и в школах продолжают процветать методы так называемого «хаотического» программирования под неумным девизом «Не важно, как написано, лишь бы работало!». Всё же подпрограммы следует применять не только тогда, когда что-то «повторяется». За соответствующими разъяснениями можно обратиться к главе «Структурная методология разработки программ» из очень популярной у нас книги [3], написанной авторами из Национального технического университета Украины «Киевский политехнический институт».

До сих пор в практике обучения отсутствуют чёткие схемы сопоставления классов задач языкам программирования, с помощью которых реализуется их решение. Многим, наверное, приходилось слышать, что язык Паскаль «лучше» языка Бейсик, но «хуже» языка Си. Каждый язык программирования хорош на своём месте, о чём учащиеся должны иметь полное представление. Каждый язык программирования должен использоваться для решения только «своих» задач. Нет смысла решать, например, чисто вычислительные задачи средствами языка Пролог, или выполнять суммирование элементов целочисленного массива на языке Си. В процессе изучения языки программирования целесообразно не объединять, а разъединять, чётко формулируя назначение и область применения каждого из них. Разумеется, на этот счёт существуют и прямо противоположные точки зрения [4], однако обоснование их не выглядит достаточно убедительным.

Следует признать, что на современном фоне развития при-

кладного программного обеспечения, ориентированного на вычисления, изучение и применение для этой цели языка программирования Паскаль выглядит весьма и весьма устаревшим. Более десяти лет назад почти повсеместный переход от Бейсика к Паскалю был воспринят с энтузиазмом. Однако, за прошедшее время стали очевидными некоторые весьма существенные недостатки систем программирования Turbo и Borland Pascal с точки зрения построения разрабатываемых программ.

Здесь, прежде всего, следует отметить, недостаточную вычислительную надёжность указанных систем программирования. В качестве примера может быть взят классический алгоритм вычисления корней квадратного уравнения [1].

```
{ $B+,D+,E+,I+,L+,N+,Q+,R+,X- }
Program Task_Equation;
Procedure SquEq(a,b,c: Real; Var x1,x2: Real; Var z: Boolean);
Var D: Real;
Begin
  D:=b*b-4*a*c; z:=D>=0;
  If Not z Then Exit;
  x1:=(-b+Sqrt(D))/(2*a); x2:=-b/a-x1
End;
Var a,b,c,x1,x2: Real; z: Boolean;
Begin
  Write('a,b,c: '); ReadLn(a,b,c);
  SquEq(a,b,c,x1,x2,z);
  If z Then WriteLn('x1=',x1,' x2=',x2)
  Else WriteLn('Вещественные корни отсутствуют');
  ReadLn
End.
```

Данная программа приведена специально для того, чтобы каждый желающий мог убедиться в том, что при решении квадратного уравнения $3x^2 - 1.98x + 0.3267 = 0$ с корнями $x_1 = x_2 = 0.33$ будет получен ответ «Вещественные корни отсутствуют». По крайней мере, именно такой ответ был получен авторами на процессоре Pentium. Как известно, программа считается неправильной, если найден хотя бы один пример, подтверждающий это. А как же быть с тем, что именно такое решение квадратного уравнения в наших школах и вузах изучается уже более десяти лет?

С другими, более серьёзными примерами вычислительной ненадёжности систем программирования Turbo и Borland Pascal при решении традиционных задач можно ознакомиться в [1].

В дополнение к этому отметим для Паскаль-программ совершенно неудовлетворительный и несовременный интерфейс пользователя при вводе исходных данных, бедные и абсолютно недостаточные средства представления результатов работы программ, удручающе слабые графические возможности. Можно только сожалеть о том, что в тематических планах некоторых компьютерных дисциплин до сих пор по старинке продолжает существовать изучение графики языка Паскаль.

Что же может и должно заменить Паскаль при изучении алгоритмизации и программирования? Предлагаемое средство лежит на поверхности, широко распространено и остаётся только методически грамотно им воспользоваться. Речь идёт о VBA (Visual Basic for Applications) – объектно-ориентированном языке программирования, особенно удобном для применения в составе табличного процессора Microsoft Excel. Это средство давно уже пробивает себе дорогу, о чём, в частности, свидетельствует [5]. В пользу VBA можно отметить следующее:

1. Характерная для Бейсика простота языковых средств, что особенно привлекает при изучении алгоритмизации и программирования на некомпьютерных специальностях.
2. Использование процедур и функций, в принципе исключаящее «хаотическое» программирование. Для учащихся навыки владения технологией структурного программирования становятся естественными.
3. Простые и доступные средства организации современного интерфейса разрабатываемой программы.
4. Возможность использования всех мощных выразительных средств Excel для представления и оформления результатов работы программы, и в первую очередь, средств графических.
5. Более высокая надёжность в вычислительном отношении. Во всяком случае, вышеприведённое квадратное уравнение Excel решает правильно.
6. Наличие объёмной библиотеки стандартных подпрограмм решения популярных математических, экономических, ста-

тистических и других задач.

Мы не призываем полностью отказаться от применения языка программирования Паскаль. На уровне средней школы Паскаль незаменим при решении олимпиадных задач по информатике, как правило, использующих целочисленные алгоритмы. В последнем случае он может изучаться заинтересованными школьниками факультативно.

Литература

1. Моргун А.Н. Решение задач средствами языка Turbo Pascal 7.0. – К.: Юниор, 2002. – 216 с., ил.

2. Інформатика та інформаційні технології. Програма для професійно-технічних навчальних закладів. Пояснювальна записка. – К.: Міністерство освіти і науки України, Науково-методичний центр професійно-технічної освіти, 2003.

3. Марченко А.И., Марченко Л.А. Программирование в среде Turbo Pascal 7.0. – К., Юниор, 1997. – 496 с., ил.

4. Гришко Л.В. Порівняльний підхід до навчання основам програмування студентів технічних спеціальностей ВНЗ // Теорія та методика навчання математики, фізики, інформатики: Збірник наукових праць. Випуск 3. Т. 3. – Кривий Ріг: Видавничий відділ НМетАУ, 2003. – С. 101-106.

5. Голубев Л.П., Пилипенко Ю.М., Юрачківський Ю.П. Методика навчання основам інформатики і програмування у VBA // Теорія та методика навчання математики, фізики, інформатики : Збірник наукових праць. Випуск 3. Т. 3. – Кривий Ріг: Видавничий відділ НМетАУ, 2003. – С. 80-97.

ЕЛЕКТРОННИЙ ПОСІБНИК З КУРСУ “ОСНОВИ ЗАХИСТУ ІНФОРМАЦІЇ”

С.В. Кукліна¹, Н.В. Моїсеєнко²

¹ м. Кривий Ріг, Середня школа №90

² м. Кривий Ріг, Криворізький державний педагогічний
університет
nvg@kpi.dp.ua

Процес інформатизації освіти і пов'язане з ним використання нових інформаційних технологій у процесі навчання змінюють компоненти теорії навчання і виховання, що впливає на зміни педагогічного процесу. У зв'язку з цим видається доцільним використання можливостей нових інформаційних технологій не стільки для підтримки традиційних форм і методів навчання, скільки для реалізації ідей розвиваючого навчання, інтенсифікації всіх рівнів навчально-виховного процесу, підготовки підростаючого покоління до умов життя в інформатизованому суспільстві.

Поступове зростання кількості обчислювальної техніки у побуті робить перспективною галузь, що пов'язана з розробкою та запровадженням електронних посібників та технологій навчання без допомоги вчителя.

Основна мета таких програм – сприяти активному включенню учнів у навчальний процес, направити їхню увагу, усунути можливі прогалини в знаннях і підготувати до сприйняття нового навчального матеріалу, за допомогою мнемічних засобів поліпшити його запам'ятовування. Багато чого з цього можна досягти і без комп'ютера, однак останній істотно заощаджує час вчителя і розширює дидактичні можливості активізації навчальної діяльності учнів.

Електронний посібник – програмно-методичний комплекс, що забезпечує можливість освоєння навчального курсу або його розділів за допомогою комп'ютера.

На відміну від звичайного, електронний посібник має бути дещо «розумним», оскільки повинен імітувати деяку діяльність вчителя. Крім того, електронний посібник має містити весь матеріал з окремої дисципліни. А наявність «розуму» дозволяє

швидко знаходити потрібну інформацію серед розмаїття термінів та означень.

З одного боку електронний посібник має відповідати за структурою та форматом звичному для нас підручнику. А з іншого – має активізувати пізнавальну діяльність. Для цього кожен розділ можна завершувати тестом.

Комп'ютер дозволяє якісно змінити рівень контролю за діяльністю учнів, забезпечуючи при цьому гнучкість управління учбовим процесом. Комп'ютер дозволяє перевірити всі відповіді, а в багатьох випадках він не тільки фіксує помилку, але достатньо точно визначає її характер, що допомагає вчасно визначити причину, яка зумовила її появу.

Тестова система самоперевірки (самоконтролю) має елементи суперництва з комп'ютером, гри і тому може виявитися найцікавішою частиною підручника. Учні більш розкуто відповідають на запитання комп'ютера. Звісно, що така система повинна використовувати яку-небудь базу тестів, яка може бути використана надалі для проведення заліку або екзамену.

Як дидактичний засіб для активізації навчально-пізнавальної діяльності учнів під час вивчення курсу основ захисту інформації нами було розроблено електронний посібник, орієнтований на роботу на персональних комп'ютерах типу IBM PC (Celeron, Pentium). Основна робота в Windows виконується за допомогою миші, яка використовується нарівні з клавіатурою. Для роботи програми необхідне середовище Internet Explorer. Мовою реалізації посібника є HTML.

Для початку роботи з навчально-контролюючим комплексом треба запустити на виконання файл Start.htm, що знаходиться в каталозі пакету OZII\Posibnyk (рис. 1). Крім головного є ще 10 підкаталогів – Rozdil1, ... , Rozdil5, test1, ... , test5.

Електронний посібник складається з п'ятих розділів, кожен з яких завершується тестом.

Розділ 1 містить вступну частину з початковою термінологією та найпростішими прикладами криптології (зокрема, з художньої літератури).

До Розділу 2 входить перелік основних протоколів та правил роботи з ними. Наявність готових прикладів має підтримати інтерес до роботи та віру у власні сили на початкових етапах на-

вчання.

У Розділі 3 міститься огляд криптографічних ключів. Основна увага пропорційно розподілена між створенням, зберіганням, знищенням та роботою з ключами. Представлені основні прийоми роботи.

У розділі 4 викладені основні криптографічні алгоритми з описом ряду найбільш популярних.

У розділі 5 викладені основи криптоаналізу: загальні методи та їх застосування до деяких найбільш популярних алгоритмів.

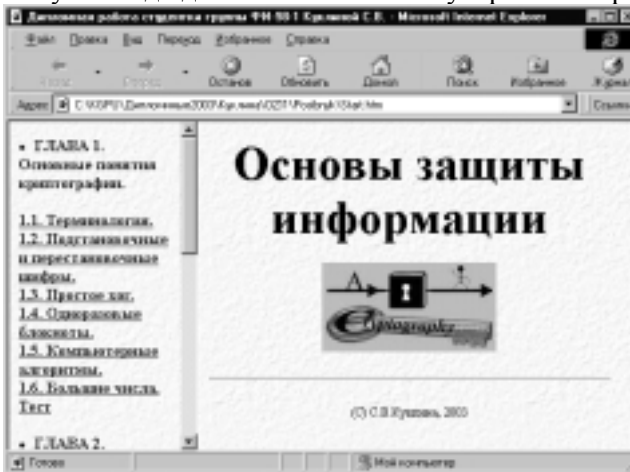


Рис. 1. Вигляд екрану після старту електронного посібника

Дана структура адаптована до кількості годин, що відведені на факультатив. Але глибина вивчення кожної теми, в першу чергу, визначається рівнем забезпечення ЕОМ та індивідуальними здібностями кожного учня.

З метою спрощення роботи при переході використовуються кнопки навігаційного середовища Microsoft Internet Explorer. Такий підхід дозволяє одночасно мати перед очима зміст посібника і переходити від одного підрозділу до іншого. Це можливо тоді, коли посібник виводиться на екран у вікні, розбитому на фрейми, де лівий фрейм (кадр) – меню вибору теми, а правий – зміст обраного розділу.

Вибір одного з пунктів посібника здійснюється за допомогою лівої кнопки миші чи переміщенням за допомогою клавіш управління курсором та натискуванням клавіші Enter для підтве-

рдження вибору.

По суті, всі частини системи – це текстові файли, що містять гіперпосилання, які завантажуються при клацанні на них мишею. В нашому випадку теми в меню вибору організовані у вигляді гіперпосилань. Тобто відбувається завантаження тексту у форматі HTML при клацанні мишею на обраному розділі у лівому фреймі вікна. Якщо повний перегляд матеріалу одночасно не є можливим, то з'являються смуги прокрутки. В такий спосіб відбувається доступ до ключових понять. У будь-який час учень може вибрати режим перегляду поняття і за допомогою кнопок меню здійснити його перегляд.

Кожен розділ містить тест із тем, що в ньому вивчаються. Тестові завдання до посібника виконані мовою JavaScript, яка була впроваджена у виді доповнення HTML.

При роботі з тестом на екран виводиться повідомлення про те, як виконувати тестові завдання (рис. 2).

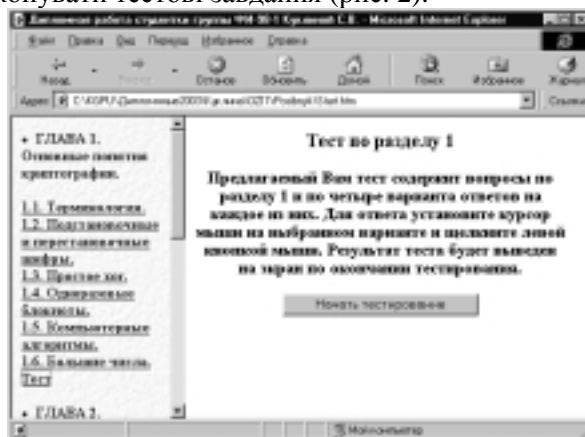


Рис. 2. Вигляд екрану на початку тестування

До кожного питання даються чотири варіанти відповідей, з яких учень має обрати один за допомогою миші. Обраний варіант виділяється іншим кольором, після натискання лівої клавіші миші з'являється коментар, правильна чи неправильна відповідь (рис. 3). Час на відповіді не обмежений, але кількість спроб обмежена двома, після другої спроби система переходить до наступного питання незалежно від результату.

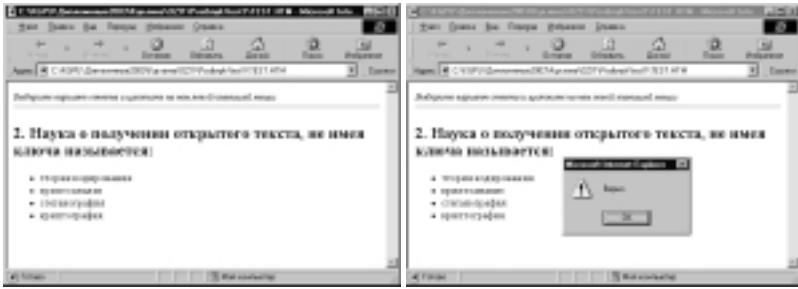


Рис. 3. Работа з тестами

Оцінка за тест виставляється після того, як учень дав відповіді на всі питання. Оцінка знижується за невірні відповіді, або вірні з другої спроби.

Для наповнення теоретичної частини підручника було використано роботу Шеннона [1], а також російські [2–7] та енциклопедичну роботу Шнейера [8].

Література

1. Шеннон К. Работы по теории информации и кибернетике. – М.: ИЛ, 1963. – С.333-369.
2. Анохин М.И., Варновский Н.П., Сидельников В.М., Яценко В.В. Криптография в банковском деле. – М.: МИФИ, 1997.
3. Аграновский А.В., Балакин А.В., Хади Р.А. Классические шифры и методы их криптоанализа. – М: Машиностроение, Информационные технологии, №10, 2001.
4. Введение в криптографию / Под общ. ред. В.В. Яценко. – М.: МЦНМО: «ЧеРо», 2000. – 288 с.
5. Жельников В. Криптография от папируса до компьютера. – М., 1996.
6. Молдовян А.А., Молдовян Н.А., Советов Б.Я., Криптография. – СПб: Лань, 2000.
7. Ростовцев А. Алгебраические основы криптографии. – СПб: Мир и Семья, 2000.
8. Schneier B. Applied Cryptography: Protocols, Algorithms and Source Code in C. – NY.: John Wiley & Sons, 1993.

О ПОВЫШЕНИИ УРОВНЯ КОМПЬЮТЕРНОГО ОБРАЗОВАНИЯ СТУДЕНТОВ ТЕХНИЧЕСКИХ СПЕЦИАЛЬНОСТЕЙ

Г.И. Кулик

г. Днепропетровск, Приднепровская государственная академия
строительства и архитектуры
kulik@pgasa.dp.ua

Сегодня каждый образованный человек стремится повысить свой образовательный уровень. Необходимым компонентом этого процесса является освоение компьютерных технологий и применение их в различных предметных областях.

Если для совершенствования в любой другой области знаний достаточно желания обучаемого и наличия учебников по заданному предмету, то с информатикой все обстоит значительно серьезнее. Требуется наличие дорогостоящей компьютерной техники, что является иногда неразрешимой проблемой как для отдельного студента, так и для учебного заведения. Также известно, что имеющиеся в наличии компьютеры нуждаются в постоянном аппаратном усовершенствовании, что также является дорогостоящим мероприятием. Кроме того, литература в данной предметной области по стоимости одного экземпляра сопоставима с размером стипендии, а иногда и заработной платы. При постоянно поддерживаемых темпах появления технических новшеств и обновления программного обеспечения очень трудно поддерживать уровень преподавания предмета «Информатика» как в средней, так и в высшей школе на современном уровне.

Результатом сложившейся ситуации является, как правило, очень низкий стартовый уровень знаний по информатике у студентов, которые пришли в аудитории технических вузов.

Работа инженера сегодня невозможна без применения информационных технологий. Задачи, которые ставятся перед студентами во время изучения курса информатики в техническом вузе, достаточно разнообразны. Во время изучения курса студент должен изучить аппаратный состав современных компьютеров, основы работы с операционными системами, получить прочные навыки в области алгоритмизации и программирования, освоить

работу с интегрированными пакетами.

Если сопоставлять перечисленные вопросы с часами, выделенными на изучение предмета, то становится ясно, что поставлены труднодостижимые цели.

В настоящее время все больше средств затрачивается на оборудование специализированных компьютерных классов. Это в значительной степени помогает и студентам и преподавателям решить поставленные перед ними задачи и является неоспоримо большим достижением.

Однако, при всем понимании важности курса «Информатика» для инженера в учебных планах не увеличиваются часы на его усвоение, а часто наоборот, уменьшаются. Так, для студентов механических специальностей курс «Информатика» изучается в течение 2-х семестров. В одном из семестров планируется проведение 10 лекционных часов и 26 часов практических и лабораторных занятий. В результате преподаватель может только обозначить темы, входящие в учебный план, а основной объем материала должен быть проработан в часы самостоятельной работы студентов.

При организации самостоятельной работы студентов также мы сталкиваемся с рядом серьезных проблем. Одной из них является дефицит рабочих мест при проведении самостоятельной работы студентов. Пропускная способность вычислительного центра значительно ниже, чем требуется. Часть студентов, которые располагают компьютерами, работают дома, часть студентов выполняют свои задания в компьютерных клубах, что является не для всех доступным по материальным соображениям. Перед студентом поставлена задача освоить материал курса при ограниченном количестве машинного времени, недостаточном количестве учебного и методического материала при перераспределении часов в пользу самоподготовки.

В таких условиях необходимо пересматривать вопросы подготовки материалов, выносимых на изучение. Необходимо обеспечить студентов доступными и емкими методическими разработками, которые позволят усвоить изучаемый предмет и применять полученные знания во время учебы на старших курсах, а в дальнейшем и во время производственной деятельности. Содержание таких методических разработок должно постоянно об-

новляться и отражать современный уровень компьютерной техники и программного обеспечения. По форме эти методические разработки также должны соответствовать современным требованиям и использовать методы представления материалов, позволяющих повысить эффективность учебного процесса.

Наиболее результативным с точки зрения автора является органичное сочетание традиционных форм представления методических материалов, так и форм, использующих современные компьютерные технологии.

Для самостоятельной работы студентов при изучении курса «Информатика» готовятся методические материалы в виде отдельных модулей, соответствующих различным темам курса. Каждая изучаемая тема готовится в виде текстовых документов, схем, слайдов, презентаций и располагается на компьютерах учебного вычислительного центра. Эти материалы могут быть использованы как при самостоятельной работе студентов, так и во время аудиторных занятий.

Такой способ подготовки и использования методических материалов обладает рядом преимуществ. Компьютерный вариант представления материалов позволяет студенту в кратчайшие сроки получать справочную информацию в часы самостоятельной работы на вычислительном центре. Предлагаемая методика позволяет активизировать аудиторные занятия и внеаудиторную работу студентов. Компьютерный вариант методических разработок легко модифицируется при изменениях в изучаемом курсе.

Предлагаемый способ позволит в сжатые сроки освоить изучаемый материал, улучшить качество преподавания и повысить уровень компьютерной подготовки студентов.

ОЦІНЮВАННЯ ВІДПОВІДЕЙ “ВІДКРИТОГО” ТИПУ ЗА УМОВ КОМП’ЮТЕРНОГО ТЕСТУВАННЯ ЗНАНЬ

Л.Л. Леоненко

м. Одеса, Одеська національна академія зв’язку ім. О.С. Попова
domos@te.net.ua

Найбільш простим і поширеним засобом автоматизованої перевірки відповідей, що подаються натуральною мовою і мають відкриту форму, є метод “ключових слів”. Добре відомо, що цьому методу притаманні важливі недоліки. Деякі з них усуваються різноманітними “вдосконаленнями” метода. Наприклад, серед ключових слів можна розрізнити “обов’язкові” (відсутність яких веде до оцінки відповіді як невірної) та “необов’язкові” (відсутність яких допускається, хоча й знижує оцінку). Можна передбачити можливість синонімічних ключових слів чи виразів. Можна також враховувати, крім ключових, “заборонені” слова, наявність яких у відповіді робить її невірною, тощо.

Та все ж подібні удосконалення не усувають наступні суттєві вади метода ключових слів:

1) якщо у правильній відповіді студента наявні всі ключові слова, проте у якихось з них допущено орфографічні помилки (притому несуттєві відносно теми опитування), відповідь може бути оцінена як невірна;

2) якщо неправильна відповідь студента містить всі ключові слова (тобто у ній є, крім ключових, також слова, що спричиняють її хибність), відповідь оцінюється як правильна;

3) метод ключових слів повністю ігнорує структуру речення, отже беззмисловна (з погляду людини) послідовність, що містить всі ключові слова, вважатиметься правильною відповіддю.

Започатковану проф. Г.В. Піддубним *теорію подібності текстів* [1] можна розглядати як узагальнення метода ключових слів, яке дозволяє “перебороти” недоліки 1)–3). Та правильніше, на мою думку, ставитись до теорії подібностей як до альтернативи згаданого метода, оскільки вона ґрунтується на деяких принципово відмінних поняттях – а саме: *a)* понятті *подібності* двох текстів; *b)* понятті *відносної значущості* різних елементів тексту; *i c)* понятті *зв’язності* груп слів у тексті.

Слід зауважити, що “тексти”, про які йде мова, можуть бути текстами і етнічних, і штучних мов. “Текст” має ієрархічну структуру: розрізняються його елементи, що належать до різних “рівнів” (наприклад, для натуральної мови до нульового рівня відносяться символи, до першого рівня – слова, до наступного – речення).

Степінь подібності двох елементів тексту S_1 та S_2 одного і того ж рівня залежить від: 1) множини спільних для S_1 та S_2 символів; 2) врахування (чи неврахування) порядку, у якому ці спільні символи розміщені у S_1 та S_2 . Запропоновано різні чисельні міри для степенів подібності [1]. Задавши для кожного ключового слова певний рівень подібності, можна забезпечити розпізнавання у відповіді студента ключових слів з допущеними у них несуттєвими помилками. Наприклад, степінь подібності слів “паралелограм” та “паарлеограм” дорівнює 0.92; отже, якщо для слова “паралелограм” задано рівень подібності 0.9, ці два слова будуть визнані еквівалентними, незважаючи на переставлення літер p і a та пропуск літери l .

У випадку речень природних мов “символами” будуть слова цих речень; і два речення матимуть високий степінь подібності, якщо для більшості слів одного з них у другому знайдуться *подібні* слова. Розглянемо приклад тестового питання з шкільного підручника з інформатики [2]:

Що називають файлом?

Застосування теорії подібності дає можливість оцінити відповідь

Поіменованую купность данных

як правильну, незважаючи на те, що у кожному її слові допущено граматичні помилки. Автор відповіді мав на увазі *Поіменовану сукупність даних* – і, очевидно, він розуміє, що таке файл, хоча й має проблеми з українською мовою. У той же час відповіді, що не містять слів, подібних до слова “ім’я” чи його синонімів (як от *Дані на магнітному диску* або *Запис у каталозі*) будуть оцінені як невірні. (Зрозуміло, що можна вибрати рівні подібності для слів і так, щоб орфографічні помилки у словах вважалися суттєвими. Наприклад слова *даних* і *данных* будуть уже *неподібними* на рівні 0.7).

Відносна значущість двох елементів тексту S_1 та S_2 одного і

того ж рівня визначається більшою або меншою важливістю присутності подібних до них елементів $p(S_1)$ та $p(S_2)$ у тексті-відповіді. Наприклад, слова “інтеграл” та “інтервал” мають високий степінь подібності (0.88). Якщо очікується поява у відповіді обох цих слів, то, щоб розрізнити їх, можна надати літері z у першому слові відносну “вагу” 3.0 (а іншим літерам – вагу 1.0). Внаслідок степінь подібності даних слів знизиться до величини 0.7 (алгоритми для обчислення таких “зважених” подібностей запропоновано у [3]). У реченні високі відносні значущості надаються найбільш важливим – для правильності відповіді – словам (а, скажімо, прийменники найчастіше дістають нульову значущість).

Наступний приклад тестового питання узятو з затверджених міністерством освіти України шкільних тестів з історії:

Де і коли відбулася перша велика битва козаків під проводом Богдана Хмельницького з польським військом?

Розподіливши відносну вагу слів “зразкової” відповіді “*Під Жовтими Водами 5-6 травня 1648 року*” наступним способом: 0, 3, 3, 1, 1, 2, 7, 0, – отримаємо такі оцінки можливих учнівських відповідей (у дужках наведено степені їх “зваженої” подібності до “зразкової” відповіді):

Жовті Води, 1648, 5 та 6 травня – відмінно (1.00)

Травень 1648, Жовті Води – добре (0.81)

Поблизу Жовтих Вод у 1648 році – задовільно (0.51)

Світлі Води, 1648 – незадовільно (0.39)

Жовті Води, 1649 – незадовільно (0.23)

Такі оцінки здаються цілком прийнятними. Водночас для відповіді *Світлі Води, 1648, 5-6 травня* отримаємо при тих самих відносних вагах слів оцінку “задовільно” (0.66), з якою можна не погодитись. Якщо викладач вважає подібні відповіді незадовільними, він може надати слову *Жовті* параметр “обов’язковості” – так, щоб будь-яка відповідь, що не містить слова, подібного на заданому рівні до *Жовті*, вважалася б неправильною.

Степінь “зваженої” подібності спадає у випадку, коли відповідь студента містить “сторонні” слова; і так само тоді, коли необхідні для правильної відповіді слова без потреби повторюються. Наприклад, відповідь

1648, 1649, Корсунь, Жовті Води, Берестечко
отримає від комп'ютера оцінку незадовільно (0.40), а відповідь
Жовті-Жовті Води, 1648, 5 та 6 травня
буде оцінено як задовільну (0.74).

Розглянемо тепер один з методів, що оцінюють *подібність структури* двох тестів ([4], див. також [5]).

Більшість речень природних та штучних мов складається з певних *груп* імен (наприклад, просте натуральне речення містить групу підмета, групу присудка, які, у свою чергу, можуть містити інші групи). *Степінь зв'язності* групи G, яка входить до деякого “тексту” T, визначається як функція, що залежить від структури розміщення між елементами групи G тих елементів T, які належать до *інших* груп T (зокрема, якщо між елементами G нема ніяких елементів, що не належать G, зв'язність дорівнює 1.0). При оцінюванні відповіді у ній знаходяться групи імен, що відповідають групам “зразка”, і порівнюються степені зв'язності цих груп. У випадку суттєвої розбіжності зв'язностей оцінка за відповідь знижується.

Наприклад, у реченні

Ньютон та Лейбніц винайшли математичний аналіз

можна виділити такі групи імен: $G_1 = \{\text{Ньютон, Лейбніц}\}$, $G_2 = \{\text{винайшли}\}$ і $G_3 = \{\text{математичний, аналіз}\}$. Зв'язність кожної з цих груп у цьому реченні дорівнює одиниці. (Взагалі, на численних прикладах речень української, російської та англійської мови можна пересвідчитись, що зв'язність “природних” груп імен, як правило, є близькою до одиниці). Для порівняння даного речення з реченнями, що виражають ту саму чи близьку думку, слід доповнити кожну з цих груп словами-синонімами (наприклад, G_2 поповнюється словами *відкрили, придумали* і т.ін., G_3 – словом *матаналіз*, тощо). Далі словам кожної групи надається відносна вага (не наводжу подробиць – ясно, наприклад, що вага групи G_2 може бути суттєво меншою за вагу інших груп). Алгоритм врахування степенів зв'язності груп у оцінці подібності структур дає такі результати для наступних прикладів речень (у дужках – степені їх “зваженої” подібності до наведеного вище речення):

Лейбніц і Ньютон – винахідники матаналізу (1.00)

Мат. аналіз був створений Ньютоном і Лейбніцем (1.00)

Лейбніц, а також Ньютон, винайшли аналіз (0.85)
Ньютон відкрив матаналіз, і Лейбніц також (0.73)
Ньютон придумав Лейбніца и створив аналіз (0.63)
Аналіз Ньютонa математичний Лейбніц придумав (0.53)
Математика Ньютонa відкрила Лейбніца аналіз (0.39)

Лексичний склад всіх цих речень практично тотожний (якщо брати до уваги подібність слів і синонімію). Як бачимо, врахування зв'язності груп імен у речення дозволяє комп'ютеру розрізнити прийнятні (перші чотири) і неприйнятні (останні три) варіанти.

Наостанок зазначимо, що обговорені вище методи є одними з складових засобів оцінювання відповідей у системі комп'ютерного тестування знань CONTROL, розробленій на кафедрі інформаційних технологій Одеської національної академії зв'язку ім. О.С. Попова. Система CONTROL використовувалась у ОНАЗ під час проведення заліків, іспитів, а також для поточного контролю знань з дисциплін “Інформатика”, “Чисельні методи розв'язування інженерних задач”, “Теорія лінійних електричних ланцюгів”, “Обчислювальна техніка та мікропроцесори”, “Технічна електродинаміка”, “Основи охорони праці”.

Ось два приклади питань з курсу інформатики:

1) *Яку задачу виконує наступний фрагмент DELPHI-програми?*

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

```

Система CONTROL оцінює як правильні, зокрема, наступні відповіді на це питання:

Викреслюються всі пробіли на початку рядка

Вилучення початкових пробілів

Тут стираються крайні зліва пробіли

А до неправильних відповідей система відносить усі ті, у

яких не сказано про вилучення пробілів і/або про розміщення цих пробілів у рядку.

2) Що таке **вказівник (pointer)**? Дайте словесне визначення цього поняття.

Правильними відповідями на це питання система CONTROL визнає, зокрема, такі:

Тип даних, що являють собою адреси інших даних у пам'яті комп'ютера;

Змінна, що містить адресу пам'яті;

Це величина, значенням якої є адреса.

Водночас відповіді на зразок

Це величина, що вказує на іншу величину

будуть оцінені системою як неправильні.

Досвід використання системи CONTROL свідчить про її ефективність, що підтвердив також педагогічний експеримент, проведений у академії з залученням усіх студентів першого курсу (з дисципліни “Інформатика”).

Література

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

2. Жалдак М.І., Морзе Н.В. Інформатика–7. Експериментальний навчальний посібник для учнів 7 класу загальноосвітньої школи. – К.: ДіаСофт, 2000. – 208 с.

3. Leonenko L. Analogical inferences in computer assisted knowledge testing systems // 6-th World Multiconf. on Systemics, Cybernetics and Informatics. Vol. XVIII – 2002. – pp. 371–376.

4. Leonenko L. Analogies between Texts: How to Appreciate the Structures // 12th Intern. Congress of LMPS: Volume of Abstracts. – Oviedo: Universidad de Oviedo, 2003. – pp. 267–268.

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

КУРС “ЦИФРОВІ ЕОМ” В ПЕДАГОГІЧНОМУ ВНЗ

Н.А. Леонова, О.А. Хараджян
м. Кривий Ріг, Криворізький державний педагогічний
університет

На початку третього тисячоліття все більшою стає потреба у використанні засобів цифрової обробки інформації: персональні комп'ютери, цифрові фотоапарати, цифрові відеокамери, цифрові засоби зв'язку і т.ін.

Для професійного розуміння принципів побудови та роботи цифрової техніки необхідно, на наш погляд, до навчального плану спеціальностей фізико-математичного факультету педагогічного університету ввести курс “Цифрові ЕОМ”, який би дав можливість студентам ознайомитись із основними пристроями, що забезпечують роботу цифрової техніки – шифратори, дешифратори, мультиплексори, демультиплексори, тригери, лічильники, цифро-аналогові та аналого-цифрові перетворювачі, запам'ятовуючи пристрої і т.п. Необхідною основою для вивчення курсу є знання електродинаміки, аналогової електроніки, алгебри логіки, теорії графів. Продовженням курсу “Цифрові ЕОМ” є курс “Архітектура ЕОМ”. Такий курс вже вивчається студентами технічних спеціальностей, зокрема, “Комп'ютерні мережі та системи”.

Мета пропонованого курсу – забезпечити теоретичну підготовку в галузі методів проектування елементів цифрових систем та електронно-обчислювальних машин. Задачі курсу – дати студентам поняття, необхідні для аналізу роботи цифрових схем, синтезу алгоритмів роботи цифрових схем, розуміння принципів побудови типових функціональних вузлів, аналізу роботи мікропроцесорних систем; синтезу роботи мікропроцесорних систем, розуміння роботи мікроконтролерів.

В результаті вивчення курсу студент повинен

Знати:

– основні положення алгебри логіки, умовні позначення логічних функцій на схемах, поняття логічного базису, методи мінімізації логічних функцій за допомогою таблиць Карно, перетворення логічних функцій до базисів “І-НЕ” та “І-АБО-НЕ”;

– принципи побудови та схеми дешифраторів, демультіплексорів, мультіплексорів, шифраторів, перетворювачів коду, суматорів, порівняння кодів, тригерів, асинхронних та синхронних RS-тригерів, D-тригерів зі статичним керуванням та динамічним керуванням, JK-тригерів, T-тригерів, паралельних та послідовних регістрів, реверсивних регістрів зсуву, синхронний та асинхронний способи завантаження паралельного коду, асинхронних та синхронних лічильників з послідовним переносом, реверсивних лічильників, цифро-аналогових перетворювачів, аналого-цифрових перетворювачів порозрядного зрівноваження (послідовних наближень), оперативних запам'ятовуючих пристроїв статичного типу, оперативних запам'ятовуючих пристроїв динамічного типу, постійних запам'ятовуючих пристроїв, які перепрограмуються та які однократно програмуються, мікроконтролерів типу PIC, Atmel, мікропроцесорів типу 80X86.

Вміти:

- описувати логічні функції у вигляді таблиці істинності;
- описувати логічні функції у вигляді алгебраїчного виразу;
- перетворювати логічні функції з одного базису в інший;
- перетворювати числа довільної системи числення в код Грея;
- мінімізувати логічні функції за допомогою карт Карно;
- синтезувати логічні пристрої в заданому базисі логічних елементів, схеми дешифраторів, схеми перетворювачів кодів, схеми мультіплексорів, схеми тригерів з заданими входами, схеми регістрів, схему цифрового автомата, схеми лічильників, схеми підвищення розрядності запам'ятовуючих пристроїв.

Нижче наводимо тематика курсу лекцій.

№	Найменування теми. Основні питання лекції та її зміст.
1.	Логічні функції та їх мінімізація. Основні положення алгебри логіки. Умовні позначення логічних функцій на схемах. Поняття логічного базису. Методи мінімізації логічних функцій за допомогою таблиць Карно
2.	Цифрова схемотехніка. Частина 1. Принципи побудови та основні схеми комбінаційних (дешифратори, шифратори, демультіплексори, мультіплексори, перетворювачі коду, суматори) схем.
3.	Цифрова схемотехніка. Частина 2. Принципи побудови та основні схеми послідовнісних схем (тригери RS, T, D, JK,

	реєстри, лічильники)
4.	Аналого-цифрові пристрої. Частина 1. Будова та принцип роботи цифро-аналогових перетворювачів (R-2R, біполярні, чотирьохквadrантні)
5.	Аналого-цифрові пристрої. Частина 2. Принцип роботи аналого-цифрових перетворювачів (перетворювачі порозрядного зрівноваження, паралельні перетворювачі)
6.	Запам'ятовуючі пристрої. Основні характеристики запам'ятовуючих пристроїв. Будова оперативних запам'ятовуючих пристроїв (статичні, динамічні). Будова постійних запам'ятовуючих пристроїв (однократні, масочні, перепрограмуємі)
7.	Мікроконтролери типу PIC. Структура, організація пам'яті, порти вводу-виводу, таймери, послідовні порти, аналого-цифрові перетворювачі
8.	Мікроконтролери типу Atmel. Структура, організація пам'яті, порти вводу-виводу, таймери, послідовні порти, аналого-цифрові перетворювачі

Перелік тем лабораторних занять

№	Найменування теми
1.	Мінімізація логічних функцій.
2.	Комбінаційні схеми.
3.	Комбінаційні схеми
4.	Послідовнісні схеми.
5.	Програмування контролерів PIC.
6.	Програмування контролерів Atmel..

Лабораторні роботи виконуються в середовищі симулятора електронних схем Electronics WorkBench.

У бібліотеки елементів програми Electronics Workbench входять аналогові, цифрові і цифро-аналогові компоненти.

Усі компоненти можна умовно розбити на наступні групи: базові компоненти, джерела, лінійні компоненти, ключі, нелінійні компоненти, індикатори, логічні компоненти, вузли комбінаційного типу, вузли послідовного типу, гібридні компоненти.

Цифрові елементи програми представлені наступними групами: індикатори, логічні елементи, вузли комбінаційного типу, вузли послідовнісного типу, гібридні елементи, пробник логічного рівня.

Пробник визначає логічний рівень (0 чи 1) у конкретній точці схеми. Якщо досліджувана точка має рівень логічної 1, індикатор загоряється червоним кольором. Рівень логічного нуля світінням не відзначається. Колір світіння пробника можна змінити.

Electronics Workbench містить повний набір логічних елементів і дозволяє задавати їхні основні характеристики, у тому числі, тип елемента: ТТЛ чи КМОП. Число входів логічних елементів схем можна установити в межах від 2 до 8, але вихід елемента може бути тільки один.

Напісуматор виконує додавання двох однорозрядних двійкових чисел. Він має два входи доданків: А, В та два виходи: суми (Sum) і переносу (Carry). Підсумовування виробляється елементом “виключне або”, а перенос – елементом “Г”.

Повний двійковий суматор виконує додавання трьох однорозрядних двійкових чисел. Результатом є двохрозрядне двійкове число, молодший розряд якого названий сумою, а старший розряд – переносом.

Дешифратор – логічний пристрій, що має n входів і 2^n виходів. Кожної комбінації вхідного коду відповідає активний рівень на одному з 2^n виходів. Даний дешифратор має три входи адреси (А, В, С), два входи (G1, G2), які дозволяють роботу дешифратора, і 8 виходів (Y0...Y7). Номер виходу, що має активний стан, дорівнює числу N, обумовленому станом адресних входів: $N = 2^2C + 2^1B + 2^0A$.

Активним рівнем є рівень логічного нуля. Дешифратор працює, якщо на вході G1 високий потенціал, а на G2 – низький. У решті випадків усі виходи пасивні і мають рівень логічної 1.

Мультиплексор (селектор даних) здійснює операцію передачі сигналу з обраного входу на вихід. Номер входу дорівнює адресі – двійковому числу, обумовленому станом адресних входів.

Демультиплексор виконує операцію, зворотну мультиплексору. Він передає дані із входу на той вихід, номер якого дорівнює адресі.

Тригер – найпростіший послідовнісний елемент із двома станами, що містить елементарну запам'ятовуючу комірку і схему керування, що змінює стан елементарної комірки. Стан тригера залежить як від комбінації на входах, так і від попереднього стану. Тригерні пристрої лежать в основі комп'ютерної операти-

вної пам'яті і використовуються в безлічі послідовнісних схем. Тригер можна скласти з простих логічних елементів.

Лічильник – елемент, що здійснює рахунок імпульсів, які подаються на його вхід. Двійкове число, що представляється станом його виходів, по фронту імпульсу на рахунковому вході збільшується на одиницю. Описуваний пристрій являє собою чотирирохрозрядний лічильник із двома входами синхронізації і чотирма виходами.

Також в Electronics Workbench є прилади для проведення вимірювань.

Найпростішими приладами є вольтметр і амперметр. Вони не вимагають настроювання, автоматично змінюючи діапазон вимірювань. В одній схемі можна застосовувати декілька таких приладів одночасно, спостерігаючи струми в різних ділянках і напруги на різних елементах.

Крім амперметра і вольтметра, у Electronics Workbench є сім приладів, з численними режимами роботи, кожний з яких можна використовувати в схемі тільки один раз: прилади для формування і спостереження аналогових величин – мультиметр, функціональний генератор, осцилограф, Бодє-плотер та прилади для формування і спостереження логічних величин: генератор слів, логічний аналізатор, логічний перетворювач.

Програма курсу є орієнтовною і може змінюватись у залежності від спеціальності та рівня підготовки контингенту.

Література

1. Завадский В.А. Компьютерная электроника. – К.: ВЕК, 1996. – 368 с.
2. Королев Г.В. Электронные устройства автоматики. – М.: Высш. шк., 1991. – 256 с.
3. Хоровиц П., Хилл У. Искусство схемотехники: В 3-х томах. Пер. с англ. – М.: Мир, 1993.
4. Аналоговая и цифровая электроника (Полный курс): Учебник для вузов / Ю.Ф. Опадчий, О.П. Глудкин, А.И. Гуров. – М.: Горячая линия–Телеком, 2002. – 768 с.
5. Ульрих В.А. Микроконтроллеры PIC16C7X. – С.-Пб.: Наука и техника, 2000. – 253 с.

ДЕКІЛЬКА ПИТАНЬ ПРО МЕТОДИ ВИКЛАДАННЯ ОЗНАЙОМЧИХ КУРСІВ ІНФОРМАТИКИ У ВНЗ (ЧАСТИНА ДРУГА)

Р.В. Ліхачов

м. Кривий Ріг, Криворізький металургійний факультет
Національної металургійної академії України

Про молоде вино та старі міхи

Якщо вдатись до припущення, що діючу систему викладання інформатики обов'язково потрібно змінювати, то неможливо залишити осторонь той факт, що зміні підлягатиме і сама методика навчання. Про які зміни йде мова?

Неможливо, в умовах стрімкого розвитку новітніх технологій, залишатись запеклими консерваторами щодо перегляду навчальної програми ознайомчого курсу інформатики. Сьогодні для формування навичок раціональної роботи з комп'ютерною технікою необхідна така система навчання, що передбачає постійне врахування вимог часу.

На нашу думку, більше уваги повинно приділятися основним засобам збереження та обробки інформації, інтерфейсам сучасних операційних систем, детальному тлумаченню понять “обчислювальна система”, “операційна система”, “програмний інтерфейс”, “інтерфейс користувача” тощо, класифікації прикладного ПЗ, питанням інформаційної безпеки та принципам побудови комп'ютерних мереж.

У відповідності до цього, необхідна перебудова усієї методичної системи навчання інформатики. Додатковою перешкодою при цьому є різні тлумачення цього терміну – від математичної теорії інформації до набору вмінь роботи з засобами ОТ. Для уникнення термінологічної плутанини доцільно у подальшому відокремити від інформатики “Курс користувача” (ознайомчий курс інформатики).

Деякі ВНЗ переглянули навчальну програму з курсу інформатики та дійшли висновку – або вивчати предмет досить глибоко (та випускати в світ фахівців у галузі комп'ютерних технологій), або не вивчати зовсім, мотивуючи відмову тим, що в сучасній школі предмет дається слухачам на достатньо високому рів-

ні.

Студенти, що навчаються сьогодні, діляться на дві категорії: ті, що володіють комп'ютерною технікою, та ті, що зіткнулися з проблемою нагальної потреби використання комп'ютера в деякій специфічній галузі, згідно з обраним фахом, та абсолютним незрозумінням того, як це зробити.

Тому необхідно приділити велику увагу побудові компактних та прозорих ознайомчих курсів, метою яких є вирівнювання базового рівня знань та вмінь з предмету. Саме такий курс має створювати можливості ефективного застосування сучасних технологій в обраній галузі, для якої комп'ютер є лише одним з інструментів для розв'язання виробничих задач.

Спробуйте відповісти на одне просте запитання – чи замислюєтесь ви, коли вмикаєте світло в кімнаті? Ні? Як це так ні? Ви що, взагалі не замислюєтесь?

А як тоді вмикаєте – користуючись алгоритмом, чи блок-схемою, що її вивішено біля вимикача? Ні! Кожен згадає, що точнісінько так, не замислюючись, він пише, не згадуючи кожного разу абетку, та норм каліграфії.

Саме рефлекторні дії дозволяють нам шляхом інтеріоризації спеціально підібраної системи вправ досягти “бездумності” у використанні комп'ютера як інструмента, перевівши ці навички зі “сфери мистецтва” у “сферу ремесла”.

Тоді методика навчання ознайомчого курсу інформатики являтиме собою лише правильне формування та закріплення інтеріоризованих дій майбутнього користувача.

Юзабіліті – нова мода чи вимога часу?

В сучасному світі великі програмні гіганти і невеличкі компанії створюють програмне забезпечення, що дозволяє користувачу поступово самовдосконалюватись і працювати з більшою віддачею. На дослідження, що сприяють створенню таких продуктів, витрачаються мільйони. Ергономіка прораховується більш ретельно, ніж параметри космічних станцій.

Юзабіліті є комплексною дисципліною, що об'єднує ергономічні вимоги до програмного забезпечення з медико-біологічними та психічними особливостями сприйняття ін формації у комп'ютерній формі з метою її найкращого засвоєння.

Подивіться на те, як користувачі з часом вдосконалюють свої методи роботи на комп'ютері. Кожен починає з простих речей, тих що на поверхні, використання маніпулятора “миша” для спрощення деяких дій тощо. Потім людина розуміє, що кожного разу відриватися від клавіатури у намаганні виконати прості дії вже занадто довго і помічає, що система дозволяє використати деякі клавіатурні аналогії. Процес роботи швидко починає прискорюватись, виникає жвавий інтерес до пошуку більш корисних методів. Врешті-решт людина працює з програмним продуктом, не помічаючи своїх дій. Швидкість самозростання користувача при роботі з програмою і визначається її юзабіліті.

Руда лисичка та ледачий пес

Створити щось по-справжньому нове важко, проте ніхто не забороняє використати те, що вже підзабулося (воно інколи має достатньо сучасний вигляд). Коли в процесі опанування матеріалу є можливість, зробивши щось не так, проконсультуватись та отримати кваліфіковане роз'яснення, ученя засвоює матеріал набагато краще.

Проблема полягає в небажанні чималої частини викладачів зацікавити студента, сприяти його бажанню вдосконалитись. Більшість з них махнула рукою на будь-які спроби нормально викласти курс, обмежуючись диктовкою лекцій та кількома типовими завданнями.

Проте давайте поглянемо, чи й справді все так важко і невірно.

Ось приклад, що покаже, як можна працювати, та надасть можливості порівняти власний підхід до проблеми.

Запропонуйте комусь вийти до дошки і написати латинську абетку для того, щоб аудиторія мала змогу побачити правильне написання окремих символів. Кожен, хто вийде, почне нервуватись перед десятком-другим очей, зробить декілька помилок, та напише символи так, що не може йти мови про їх написання згідно правилам.

Що трапилось? Невже така проста задача викликати такі непрості проблеми? Так!

А тепер запропонуйте будь-кому написати на дошці таку англійську фразу “A quick brown fox jumps over the lazy dog”, задля

того, щоб зрозуміти, про що там йде мова. Завдання буде виконано бездоганно. Що таке? До чого тут руда лисичка, до чого тут ледачий пес? Про що йде мова?

Мова йде про те, що, змінивши акцент, ви досягли мети в написанні англійської абетки, до того ж бездоганно і достатньо каліграфічно правильно (ця кумедна фраза несе в собі весь запас латинських символів, з яких складена абетка).

Не можна акцентувати увагу на засобах – потрібно намагатись досягти мети. Якщо ви замислитесь над тим, як вмикаєте світло в кімнаті, то можете і не потрапити у вмикач. Якщо ж спробуєте побачити в темній кімнаті кішку, то рука сама ввімкне лампу!

Могутня рука Фортунати, або “час сміливих”

Ті, хто не бажає змін, можуть зауважити на те, що кількість студентів в групах не дає можливості нормального проведення занять та спілкування з кожним окремо, як підсумок – треба все робити, як раніше.

У багатьох закарбувався в пам’яті 1986 рік. Трагедія, що, так чи інакше, увійшла в кожен дім. Про що мова? Про те, що за офіційними даними кількість студентів, що вступатимуть до ВНЗ України в майбутні 4–5 років, різко зменшиться. А це шанс проводити більше часу з кожним з них. Трагедія, що дала не лише біль, а і можливість змінити напрямки викладання безболісно, та корисно.

ОСОБЕННОСТИ ПРЕПОДАВАНИЯ ДИСЦИПЛИНЫ «МИКРОПРОЦЕССОРНЫЕ УСТРОЙСТВА»

С.С. Лихман

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

Дисциплина «Микропроцессорные устройства» относится к группе специальных дисциплин, изучаемых студентами на кафедре «Компьютерные системы автоматизации и управления электроприводом» Криворожского металлургического факультета Национальной металлургической академии Украины.

Изучение ее невозможно без наличия базовых знаний в следующих областях: дискретные устройства, электроника, информатика, алгоритмы и программирование, двоичная арифметика, булева алгебра, логика.

В соответствии с учебным планом одной из целей изучения дисциплины студентами является получение практических навыков программирования микроконтроллеров, применяемых в системах управления электроприводом. Поэтому академией был приобретен лабораторный стенд, содержащий микроконтроллер (семейства 8031) – как основное управляющее устройство и функциональные дополнительные устройства: отображения информации, преобразования информации из аналогового типа в цифровую (дискретную) и обратно, источники постоянной и переменной частоты. Этот набор устройств наиболее активно применяется при программировании задач управления.

Лабораторные работы составлены по принципу: «делаешь и видишь результат». Такой подход, как показал опыт, вызывает повышенную заинтересованность студентов. На первой лабораторной работе, цель которой – ознакомление с лабораторным стендом, демонстрируются все возможности стенда. При этом студент не разрабатывает программу, а только вводит и наблюдает ее работу.

Создание заинтересованности при изучении дисциплины является одним из условий качественного восприятия материала студентом. Поэтому «интересные» элементы нужно вносить и в лекционный материал. Это могут быть просто интересные при-

меры использования обсуждаемых предметов или даже необычные аналогии, использующие обычные термины, с которыми студенты часто встречаются в повседневной жизни. Скорее всего, студент и запомнит именно этот, интересный ему элемент, а затем уже по аналогии восстановит то, что старался донести до него преподаватель. Например, при изучении механизма прерываний микроконтроллера довольно удобно провести аналогию между системой прерываний и человеком, которого застал звонок телефона во время чтения книги.

Кроме того, заинтересованность студента могут вызвать изделия, демонстрирующие работу микроконтроллера, применение микроконтроллеров в необычных условиях и высоких технологиях, последние научные разработки в этой области. Поэтому современный преподаватель должен не только хорошо знать свой предмет, но и ориентироваться в направлениях развития предметной области. Естественно, обеспечить это можно только при наличии свежей информации, источниками которой могут служить периодические издания, СМИ и всемирная сеть Internet. К счастью, у преподавателей нашей кафедры имеется доступ к сети Internet и некоторым периодическим изданиям.

Необходимо отметить, что на данный момент недостаточно прослеживаются межпредметные связи: хотелось бы, чтобы знания и практические навыки, приобретенные студентами при изучении предмета, нашли применение в процессе изучения других дисциплин. Примером может служить взаимосвязанное задание для курсового проектирования, являющееся продолжением курсового проекта по теории управления. Разработанная студентом цифровая система управления может быть реализована с помощью микропроцессорной системы. Это даст возможность увидеть практическое применение теоретических знаний и связь между предметами. Такой подход даст в руки студентов серьезный инструмент, с помощью которого они смогут успешно решать задачи управления.

МЕТОД АПРИОРНОЙ ОЦЕНКИ ИНФОРМАТИВНОСТИ ФАКТОРОВ, ФОРМИРУЮЩИХ ПОЗНАВАТЕЛЬНУЮ САМОСТОЯТЕЛЬНОСТЬ СТУДЕНТОВ ПРИ ИЗУЧЕНИИ ДИСЦИПЛИН КОМПЬЮТЕРНОГО ЦИКЛА

В.Г. Логвиненко

г. Сумы, Сумской национальный аграрный университет
vs@sau.sumy.ua

Познавательная самостоятельность (ПС) студентов была определена [1, 2] как интегративное качество личности, имеющее в основе познавательную активность, связанное с инициативой, с поиском различных путей решения учебно-познавательных задач без участия преподавателя (преподаватель подготавливает систему заданий), и обеспечивающее саморазвитие личности. Была предложена структура понятия, которая включала пять компонент, и на основе анализа литературных данных были выделены свойства личности, которые, по мнению большинства исследователей, формируют ПС. Эти свойства с позиции исследования ПС мы будем называть факторами ПС.

Задача ПС относится к числу многофакторной и многомерной, поэтому необходимо выделить те факторы ПС, на которые необходимо воздействовать в первую очередь и которые приводят к повышению эффективности учебно-познавательной деятельности.

Разные исследователи выявили значительное число факторов ПС. Но вопрос о степени проявления каждого фактора в общей структуре познавательной самостоятельности студента, не был даже поставлен, не говоря об его решении. Поэтому задача сформулируем следующим образом: используя структуру ПС как эксплицированного понятия, выбрать и обосновать метод оценки информативности факторов ПС и их ранжирования в аспекте педагогической ценности для управления ПС.

Понятие информативности факторов. В педагогической литературе формализованное представление информативности факторов было введено, по-видимому, впервые в [3] для задачи оценки качества учебно-познавательной деятельности на основе методов нечёткой логики и группового учёта аргументов. В ас-

пекте нашего исследования *информативность фактора* – способность этого фактора содержать информацию о степени его влияния на ПС.

Метод априорной оценки. Для выявления степени влияния какого-либо фактора на развитие ПС наиболее подходящими методами остаются экспертные методы, основанные на субъективных оценках экспертов.

Для нашей цели наиболее приемлемым является метод парных сравнений. Его суть состоит в следующем.

Формирование анкеты. Для определения степени влияния факторов на развитие ПС студента формируем специальную анкету, в которой педагогам-экспертам предлагается оценить степень совместного проявления основных характеристик (факторов) ПС. Для этого им необходимо заполнить таблицу, в которой характеристики (факторы) познавательной самостоятельности записаны: а) сверху вниз в первом столбце таблицы; б) справа налево в том же порядке в шапке таблицы 1.

Таблица 1. Определение степени влияния факторов на развитие познавательной самостоятельности

Номер фактора	20	...	3	2	1	Ранг, R	Степень влияния, P
1					–		
2				–			
3			–				
...		–					
20	–						

Исходя из [1], познавательная самостоятельность при изучении компьютерных дисциплин включает следующие факторы: 1) потребность и желание овладеть знаниями и способами деятельности; 2) познавательный мотив и интерес; 3) интерес к результатам своей самостоятельной познавательной деятельности; 4) интерес к будущей профессии; 5) инициативность; 6) опорные знания (которыми владеет личность); 7) опорные умения и навыки, владение ПК и изученными ранее программными средствами; 8) приобретенные знания по изучаемой дисциплине компьютерного цикла; 9) приобретенные умения и навыки по изучаемой

дисциплине компьютерного цикла, владение ПК и изученными программными средствами; 10) использование научно-методической литературы, средств коммуникаций, Интернета; 11) внимательность; 12) волевые усилия; 13) целеустремленность; 14) настойчивость; 15) контактность с преподавателем во время выполнения самостоятельной познавательной деятельности с целью получения информации; 16) контактность с другими студентами во время выполнения самостоятельной познавательной деятельности с целью получения информации; 17) умение ставить и достигать цели познавательной деятельности; 18) умение планировать свою познавательную деятельность; 19) умение оценить свои потенциальные возможности при выполнении познавательной деятельности; 20) умение оценить результаты своей познавательной деятельности.

Составленная таблица просматривается слева направо: каждый элемент столбца сравнивается с каждым элементом строки. Если, по мнению эксперта, элемент, находящийся в левом столбце таблицы, встречается реже, чем элемент, помещенный в верхней строке, то в клетке на пересечении столбца и строки записывается единица. Если элемент, находящийся в верхней строке таблицы, встречается чаще, чем в левом столбце, то ставится ноль. В случае равных частот проявлений в ячейке ставится значение 0,5.

Например, если сопоставляются элементы 3 и 12, то можно это описать следующим образом: «проявляется ли в структуре познавательной самостоятельности, по Вашему мнению, имеет интерес к результатам своей познавательной деятельности при изучении дисциплин компьютерного цикла чаще, чем проявление волевых усилий, которые прилагает студент при выполнении познавательной деятельности». Заполняются все клетки таблицы выше диагонали. Клетки таблицы, находящиеся внизу диагонали, заполняются симметрично этой диагонали. Например, если в клетке на пересечении столбца 3 (фактор 3) и строки 12 (фактор 12) эксперт вписал 1 (0,5), то в клетке на пересечении столбца 12 (фактор 12) и строки 3 (фактор 3) эксперт должен вписать 0 (0,5). Последние два столбца для *R* и *P* эксперты не заполняют.

В качестве экспертов приглашаются преподаватели компьютерных дисциплин с опытом работы в вузе.

Обработка анкет. В результате заполнения экспертами анкеты можно рассчитать степень проявления составляющих ПС. Допустим, что эксперты отвечали на вопросы анкеты, т.е. заполняли все столбцы таблицы 1, кроме двух последних. Далее полученная информация обрабатывается следующим образом:

1. Выполняется построчное суммирование содержимого клеток. Результатом такого суммирования является ранг R каждого фактора в структуре познавательной самостоятельности. Он записывается в соответствующий столбец «ранг R » таблиц.

2. Фактору с наивысшим рангом присваивается степень, равная 1. Например, в табл. 2 наивысший ранг имеет фактор 5; в табл. 3 наивысший ранг имеют факторы 8 и 9; в табл. 4 наивысший ранг имеет фактор 1.

3. Рассчитывается степень проявления P_i каждого фактора по формуле:

$$P_i = \frac{P_{\max} \cdot R_i}{R_{\max}},$$

где $P_{\max}=1$ – степень проявления фактора, который имеет наивысший ранг; R_i – ранг фактора; R_{\max} – наивысший ранг.

Результаты заносятся в столбцы «Степень проявления P » согласно таблице 1. Так обрабатываются все данные.

4. Заполняется таблица 2, в которую сводятся результаты анкет.

Таблица 2. Сводная форма для заполнения результатами расчетов

№ элемента	P_1	P_2	P_3	...	P_n	P_{cp}	S_i	v_i	P_i^g	P_i^H
1										
2										
3										
...										
20										

В эту таблицу переносятся результаты вычисления степеней P_{ni} из каждой анкеты n -го эксперта по каждому i -му фактору, т.е. для 1-го эксперта заполняется столбец P_1 , для 2-го – P_2 и т.д., для n -го эксперта – P_n .

5. На основании полученных данных P_i рассчитываются усредненные значения степеней проявления факторов в структуре

познавательной самостоятельности:

$$P_{i,cp} = \frac{\sum_{j=1}^n P_{ij}}{n},$$

где n – количество экспертов, j – номер эксперта; i – номер фактора. Эти величины уже могут служить мерой информативности фактора.

6. Определяется дисперсия оценок экспертов S_i по каждому фактору ПС, которая вычисляется по формуле (для малой выборки используется несмещенная оценка дисперсии):

$$S_i = \sqrt{\frac{\sum_{j=1}^n (P_{i,j} - P_{i,cp})^2}{n-1}},$$

где n – количество экспертов, $P_{i,j}$ – степень проявления в структуре познавательной самостоятельности i -го фактора по мнению j -го эксперта. Величина $n-1$ указывает на количество степеней свободы.

7. Определяется доверительный интервал значений для каждой составляющей познавательной самостоятельности по формулам:

$$v_i = t_\varphi * \frac{S_i}{\sqrt{n}}; \quad P_i^g = P_{i,cp} + v_i; \quad P_i^n = P_{i,cp} - v_i,$$

где v_i – доверительный интервал; t_φ – доверительная вероятность; P_i^g (P_i^n) – верхняя (нижняя) доверительная граница значений информативности факторов познавательной самостоятельности.

Результаты и их анализ. Расчеты показывают, что с определенной надежностью (уровнем значимости), средние значения информативности для каждого из факторов познавательной самостоятельности находятся в рассчитанных границах. Все значения находятся в диапазоне [0;1].

Ранжирование значений верхних границ доверительных интервалов позволит говорить о наиболее информативных факторах познавательной самостоятельности.

Выводы

1. Для выявления степени влияния какого-либо фактора на развитие ПС наиболее подходящим методом является метод парных сравнений.
2. Усредненная экспертная оценка степени влияния какого-либо фактора может рассматриваться как мера информативности этого фактора в структуре познавательной самостоятельности.
3. Ранжирование факторов по их информативности позволяет воздействовать в первую очередь на те факторы ПС, которые приводят к повышению эффективности учебно-познавательной деятельности.

Литература

1. Логвиненко В.Г. Познавательная самостоятельность студентов в условиях информатизации образования как объект педагогических исследований // Проблемы інженерно-педагогічної освіти. Збірник наукових праць. Випуск 4. – Харків, УПА, 2003. – С. 30–35.
2. Логвиненко В.Г. Познавательная самостоятельность студентов вузов: состояние проблемы. // Проблемы инженерно-педагогічної освіти. Збірник наукових праць. Випуск 5.– Харків, УПА, 2003. – С. 347–356.
3. Ашерев А.Т., Ящун Т.В. Аналіз інформативності факторів навчально-пізнавальної діяльності в системі «людина-комп'ютер». І. Теоретичні основи // Вісник Сумського державного аграрного університету: Науково-методичний журнал “Механізація та автоматизація виробничих процесів”. – Сумы. – 1999. – №4. – С.166-170.

ФОРМУВАННЯ ДОСЛІДНИЦЬКИХ ВМІНЬ І НАВИЧОК У СТУДЕНТІВ НА ОСНОВІ ВИВЧЕННЯ МОДЕЛЕЙ МЕХАНІЧНИХ СИСТЕМ У СЕРЕДОВИЩІ SIMULINK

Ю.Г. Лотюк^α, О.Є. Щодро^β

м. Рівне, Міжнародний університет “Рівненський економіко-гуманітарний інститут” імені академіка Степана Дем’янука

^α lotyuk@rivne.com

^β aleug@ukrwest.net

При проведенні практично-лабораторних занять з курсів “Теорія систем” та “Моделювання складних систем” зі студентами 4-х і 5-х курсів спеціальностей “Інформатика” та “Прикладна математика” розглядаються прості математичні моделі, побудовані на основі звичайних диференційних рівнянь при завданні початкових умов. Безпосередньо перед розв’язанням задачі проводиться аналіз її фізичних властивостей та виконується декомпозиція за силами, що діють на її елементи. Внаслідок проведеного аналізу вдається записати диференційне рівняння, що лежить в основі моделі, а також побудувати відповідні алгоритмічні та структурні схеми, які відображають всебічні зв’язки між її елементами. Такий підхід властивий аналітичному моделюванню, коли після отримання диференціального рівняння аналізується його розв’язок. Для вивчення властивостей моделі за отриманим розв’язком диференціального рівняння застосовується потужний апарат сучасного математичного аналізу.

Проте, не завжди вдається скласти диференціальне рівняння, яке б з достатньою точністю описувало фізичне явище, а тим більше знайти його прийнятний розв’язок. У цьому разі застосовується імітаційна модель фізичного явища. Хоча вона менш точна порівняно з аналітичною моделлю, проте вона простіша, більш наочна та сприятлива для розуміння студентів. Єдиним недоліком імітаційної моделі є велика кількість обчислень, які необхідно провести для її побудови та дослідження. Але при застосуванні Simulink, що є додатком до комп’ютерного середовища MatLab, що призначене для професійної роботи у математичних, інженерних, та інших застосуваннях, це обмеження знімається. Використовуючи дану комп’ютерну інтерактивну систему, мож-

на у діалоговому режимі створювати та налагоджувати програми довільної складності, причому запис виразів наближений до математичних форми.

Додаток Simulink [1] призначений для створення динамічних моделей: дискретних, неперервних та гібридних. Такі моделі можуть відображати основні предмети дослідження – системи управління, обробка сигналів та зображень, аеродинамічні моделі, моделі електронних схем тощо.

Користувач конструює досліджувану схему з блоків графічної бібліотеки, користуючись технологією “Drag and Drop”. Після побудови схеми використовуються підпрограми, що реалізують у чисельному або графічному вигляді алгоритми обробки або аналізу заданої схеми.

Відкрита архітектура дозволяє розширювати засоби моделювання, тобто створювати бібліотеки блоків користувача з іконками, створювати інтерфейс користувача з MatLab, Fortran або C-програмами.

Студенти на стадії аналізу окремих блоків моделі застосовують Simulink, будуючи графічні зображення вихідних процесів системи при заданому взаємозв’язку між елементами системи та з врахуванням вхідних сигналів.

Розглянемо математичну модель та її реалізацію за допомогою додатку Simulink.

На прикладі динамічної моделі з механіки покажемо процес створення та дослідження моделі засобами додатку Simulink, як це робиться на лабораторних роботах з “Моделювання складних систем” зі студентами 4-го курсу спеціальностей “Інформатика” Міжнародного університету “РЕГГ” ім. акад. С. Дем’янчука.

Задача. Знайти закон зміни координати x візка від часу [2]. Візок рухається під дією сили пружності (рис. 1). Руху візка протидіє сила опору rV , яка залежить від швидкості візка. Відомі початковий розтяг пружини x_0 , маса візка m , коефіцієнт опору руху r , жорсткість пружини k . Дослідити, як змінюється характер коливань в залежності від маси візка, жорсткості пружини, коефіцієнту опору руху.

Побудуємо імітаційну модель системи, описаної у задачі. На візок діє сила $F_1=kx-Vr$, яка складається із сили пружності та протидіючої їй сили опору. Знаючи цю силу та масу візка, визна-

чимо миттєве прискорення візка $F_2=ma$, $F_1=F_2$, $a = \frac{kx - Vr}{m}$, де k

– коефіцієнт пружності пружини, r – коефіцієнт опору руху, V – миттєва швидкість візка, x – розтяг пружини, m – маса візка. Ви- значивши миттєве прискорення, знайдемо миттєве переміщення

візка $\Delta x = \frac{a\Delta t^2}{2} + \Delta Vt$. На кожній ітерації будемо визначати ко-

ординату X візка $x_n=x_{n-1}+\Delta x$ та миттєву швидкість $V_n=V_{n-1}+a\Delta t$, де n – номер ітерації.

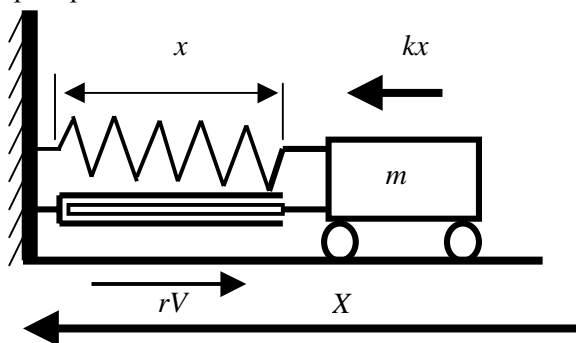


Рис.1. Механічна схема задачі.

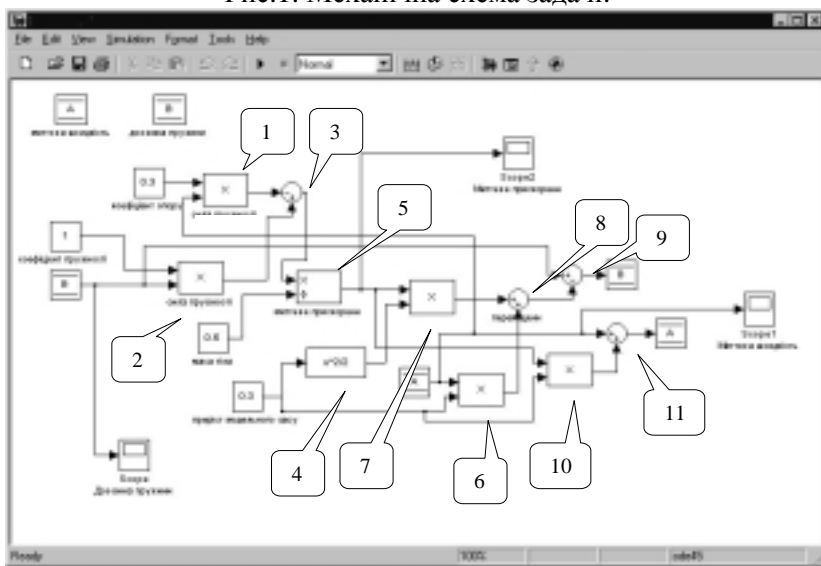


Рис. 2. Робочий лист MatLab з моделлю до задачі.

Зобразимо побудовану імітаційну модель за допомогою типових динамічних блоків у робочому листі MatLab.

1. Знаючи коефіцієнт пружності та розтяг пружини, обчислимо силу пружності – блок2;
2. Знаючи коефіцієнт опору та миттєву швидкість, обчислимо силу опору – блок1;
3. Віднімемо силу опору від сили пружності та отримаємо рівнодійну цих сил, прикладену до візка – блок3;
4. Для визначення миттєвого прискорення поділимо отриману силу на масу візка – блок5, миттєве прискорення відразу візуалізується блоком Score2;
5. Використовуючи константу приросту модельного часу, блок4 обчислює вираз $\frac{\Delta t^2}{2}$, а блок7 – вираз $\frac{a\Delta t^2}{2}$;
6. Для визначення переміщення візка, тобто зміни довжини пружини, враховуємо зміну миттєвої швидкості. Блок6 обчислює вираз ΔVt , а на виході блока8 отримуємо миттєве переміщення;
7. Блок9 обчислює результуюче переміщення та поновлює константу B;
8. Блок10 обчислює зміну миттєвої швидкості;
9. Блок11 обчислює результуючу зміну швидкості та поновлює константу A;
10. Миттєва швидкість та довжина пружини візуалізуються відповідно блоками Score1 та Score.

Модель створена, записана типовими динамічними блоками у робочому листі MatLab та відлагоджена.

Студентам видається завдання дослідити поведінку трьох параметричної механічної системи (змінюються коефіцієнт опору руху, маса візка, жорсткість пружини) при фіксованому значенні одного параметра, який для кожного студента заданий індивідуально, згідно його номера по журналу.

Використовуючи додаток Simulink, навіть слабкий студент на протязі 50–60 хвилин може провести повне дослідження математичної моделі, знайти критичні умови, які виділяють зони стійкої і нестійкої поведінки.

Наприклад, для дослідження зон стійкої та нестійкої поведінки, будемо змінювати коефіцієнт опору r . На рис. 3 показані

три основні зони (зліва направо) – амплітуда коливань наростає, амплітуда коливань стабільна, амплітуда коливань спадає (затухаючі коливання). Студент має визначити коефіцієнт опору, для якого амплітуда коливань є стабільною у часі. Для цього може використовуватися як метод спроб і помилок (найгірший варіант), так і методи аналізу моделі. Так, визначивши коефіцієнт зміни амплітуди коливань від коефіцієнту опору r : $Ak(r)F(t)$, можна прирівняти його до одиниці $k(r)=1$, та знайти відповідне значення для r будь-яким чисельним методом (у MatLab є десятки вбудованих функцій для чисельного розв'язування рівнянь).

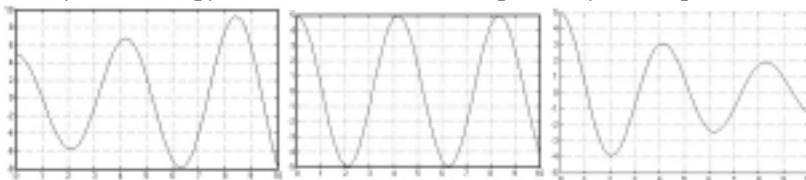


Рис. 3. Закон руху для $r=0.05, 0.15, 0.3$.

Аналогічно проводимо дослідження для різних значень маси візка m , та різних значень жорсткості пружини k . На думку авторів, доцільно вдосконалити застосування описаної методики з врахуванням можливості проведення дослідницької роботи студентами.

Література

1. Дьяконов В. Simulink4. Специальный справочник. – СПб: Питер, 2002. – 528 с.:ил.
2. Лукас В.А. Теория автоматического управления. – М: Недра, 1990. – 415 с.

ЗАСТОСУВАННЯ ДОСЛІДНИЦЬКОГО МЕТОДУ У ПРОЦЕСІ НАВЧАННЯ ІНФОРМАТИКИ

І.М. Лукаш

м. Чернігів, Чернігівський державний педагогічний університет
Lukash@cg.ukrtel.net

Дослідницький метод застосовується в різних педагогічних технологіях навчання, зокрема при реалізації розвивального навчання, в основі системи якого лежить уявлення про розвиток дитини як суб'єкта особистої діяльності. А це означає, що головна мета розвивального навчання – забезпечити розвиток дитини. Проте слід враховувати те, що не кожна зміна в дитині є зміною в її розвитку. Доцільніше звертати увагу на інтелектуальні зміни, психічні новоутворення, а не на розвиток умінь та навичок, хоч і це не треба відкидати.

Починаючи з вісімнадцятого століття, в педагогіці приділялася велика увага розумовому розвитку дитини. Так, видатний представник гуманістичного і демократичного напрямку в західноєвропейській педагогіці – французький просвітник Жан-Жак Руссо (1712–1778) головний шлях розвитку дитини вбачав в особистому досвіді. На його думку, необхідно розвивати спостережливість, допитливість на основі активної діяльності. Пізніше, в 20-ті роки двадцятого століття П.П. Блонський показав, що мислення пов'язане із загальним розвитком дитини: дія переходить в думку, думка породжує дію. Тоді педагогіка висунула ідею широкого впровадження в практику школи дослідницького методу навчання, за яким учень повинен в певній мірі повторити процес наукового дослідження.

Однією із задач вчителя, який планує застосовувати дослідницький метод в навчальному процесі, є ознайомлення учнів з методами наукового дослідження при вивченні програмного матеріалу з предмету. Серед методів наукового дослідження доцільно виділити найбільш загальні і доступні для учнів емпіричні методи, такі як: спостереження, порівняння, вимірювання, експеримент, моделювання, абстрагування, аналіз і синтез тощо. Програмний матеріал з інформатики надає вчителю безліч можливостей для формування в учнів умінь застосовувати методи

наукового дослідження при розв'язуванні практичних задач, для реалізації дослідницького методу навчання. Розглянемо кілька прикладів.

Метою проведення спостережень є отримання певної інформації про об'єкт у вигляді емпіричних тверджень. Так, при вивченні теми “Архівація файлів” на основі спостереження можна провести такі дослідження: 1) дослідження властивостей стиснення різних форматів текстових і графічних файлів при одному з режимів стиснення; 2) дослідження ефективності алгоритмів стиснення, що застосовуються до того самого файлу.

Перший варіант дослідження передбачає створення та архівування одним з способів кількох непустих файлів різних типів: .txt, .doc, .bmp, .gif тощо. Результатом проведеного дослідження буде створення та заповнення таблиці 1.

Таблиця 1.

Формат файлу	Початковий розмір файлу (Кбайт)	Розмір файлу в архіві (Кбайт)	Ступінь стиснення (%)

Другий варіант дослідження передбачає створення та архівування файлу певного формату різними способами стиснення: швидким, звичайним (нормальним), максимальним. Результатом проведеного дослідження буде створення та заповнення таблиці 2.

Таблиця 2.

Режим стиснення	Початковий розмір файлу (Мбайт)	Розмір файлу в архіві (Мбайт)	Час, с	Ступінь стиснення (%)	Ефективність методу

Ступінь стиснення можна визначити за формулою:

$R = \frac{S_a}{S} \cdot 100\%$, де R – ступінь стиснення, S – початковий розмір файлу, S_a – розмір файлу в архіві.

Ефективність методу стиснення оцінюється за абсолютною величиною нарощування ступеня стиснення до нарощування часу стиснення:

$$E_i = \frac{|R_i - R_{i-1}|}{|t_i - t_{i-1}|}$$

Формуючи варіанти завдань для учнів, можна змінювати як типи файлів для архівування, так і програми-архіватори.

При вивченні мов програмування можна запропонувати учням дослідити різні операції, зокрема роботу чисельних функцій.

Однією з таких функцій є $y = \sqrt{x}$. У процесі проведення дослідження вираз $y = \text{sqrt}(x)$, де x, y – дійсні числа, дублюється, наприклад, 100 разів в тілі циклу, що виконується α разів. Час виконання одного виразу буде обчислюватися за формулою

$t = \frac{(\tau - \tau_0)}{100\alpha}$, де τ – час виконання циклу з виразом та вимірювальної процедури `GetTime`, τ_0 – час виконання порожнього циклу та процедури `GetTime`.

Вивчення основ алгоритмізації та програмування на основі об'єктно-орієнтованого програмування надає учням можливість для оволодіння комплексом евристичних методів наукового дослідження. Звісно, що дослідницький метод, наприклад, при вивченні компонент `Delphi` носить імітаційний характер. Досліджувані учнями компоненти `Delphi` не є новими, вони створювалися своїми розробниками, але вони нові для учнів, а відкриття нового відбувається через відповідну розумову діяльність першовідкривачів, яка передбачає виконання комплексу інтелектуальних операцій (спостереження, аналіз, синтез, порівняння, виділення головного, класифікація, узагальнення, передбачення тощо).

Сутність науки вбачається не в простій реєстрації фактів, а в упорядкованому порівнянні та класифікації знань. Знайдені нові деталі порівнюються і включаються в класифікацію вже існуючих знань. Згідно з діями, які виконуються при створенні програмної реалізації проекту, компоненти `Delphi` можна поділити, наприклад на такі групи (за функціональним призначенням): 1) компоненти для реалізації введення даних; 2) компоненти для реалізації виведення результатів та повідомлень для користувача; 3) компоненти для здійснення управління процесами при роботі програмного проекту; 4) компоненти, що застосовуються для відображення стану виконуваної дії; 5) компоненти для естетичного оформлення вікон проекту; 6) компоненти спеціального призначення (для створення графічних зображень, мультимедійних додатків, баз даних, Web-сторінок, ділової графіки тощо).

Сутність науки вбачається не в простій реєстрації фактів, а в упорядкованому порівнянні та класифікації знань. Знайдені нові деталі порівнюються і включаються в класифікацію вже існуючих знань. Згідно з діями, які виконуються при створенні програмної реалізації проекту, компоненти `Delphi` можна поділити, наприклад на такі групи (за функціональним призначенням): 1) компоненти для реалізації введення даних; 2) компоненти для реалізації виведення результатів та повідомлень для користувача; 3) компоненти для здійснення управління процесами при роботі програмного проекту; 4) компоненти, що застосовуються для відображення стану виконуваної дії; 5) компоненти для естетичного оформлення вікон проекту; 6) компоненти спеціального призначення (для створення графічних зображень, мультимедійних додатків, баз даних, Web-сторінок, ділової графіки тощо).

Знаходячи для себе якусь нову компоненту (стандартну або нестандартну), визначивши її основні функції, учень на основі відповідних порівнянь з відомими йому компонентами повинен віднести її до цієї чи іншої групи, тим самим виконавши операцію класифікацію. Окрім цього, користування систематизованим інструментарієм спрощує учню орієнтацію в наборі компонент, виконання операції вибору тієї компоненти, яка на його думку буде більше підходити для реалізації деякого фрагменту програми.

В науці дослідження проводиться згідно з наступними етапами [2]: 1) виявлення незрозумілих фактів; 2) формулювання та уточнення проблеми, збір даних; 3) висування гіпотези; 4) складання плану досліджень; 5) виконання дослідження, перевірка гіпотез; 6) формулювання результату; 7) оцінка можливостей застосування нового знання. Перенесення дій наукового дослідження на процес вивчення окремих компонент Delphi потребує внесення відповідних коректив, після застосування яких дослідження компоненти доцільно проводити згідно з такими етапами: 1) виявлення нової компоненти; 2) виявлення її властивостей, методів, подій, на які реагує компонента, батьківського класу, нащадком якого вона є; 3) висування гіпотези про призначення компоненти і виконувані нею функції; 4) складання плану дослідження, експериментальної перевірки роботи методів і властивостей; 5) проведення експериментів на простих прикладах; 6) віднесення компоненти до відповідної групи в системі компонент згідно з отриманими результатами дослідження; 7) оцінка можливостей застосування досліджуваної компоненти.

Виходячи з вибраної тактики дослідження, розробка і реалізація експериментів значно залежить від знаходження аналогій між новими фактами, що спостерігаються, та попереднім досвідом. Планування експерименту доцільно проводити за такими етапами: 1) спостереження факту наявності якогось елемента компоненти та формулювання ідеї його функціонування або поведінки компоненти; 2) зіставлення нового з чимось вже відомим із попереднього досвіду; згідно з методом аналогії, якщо об'єкти подібні в ряді властивості, то вважається, що вони подібні і в інших; тобто, якщо нова компонента має ті ж самі властивості, що і вже відома, то можна передбачити, що вона буде мати ту ж са-

му поведінку, а, тим самим, і аналогічний спосіб використання; 3) висунення припущення, що нову компоненту можна застосувати в тих самих випадках, що і відому компоненту; 4) виявлення відмінностей між новою компонентою та компонентою з минулого досвіду; 5) висунення припущення, що завдяки цій відмінній інформації нова компонента може знайти ширше застосування, може використовуватись в інших випадках. В ході експериментів перевіряється використання компоненти в різних випадках, в ході чого відбувається підтвердження або спростування висунутих гіпотез.

За допомогою аналогії подібність предметів, виявлена в результаті їх порівняння, поширюється на нові властивості (нову властивість). В навчанні, як і в науці, аналогія часто корисна тим, що вона наводить людину на здогадки, тобто є евристичним методом. В навчанні не менш важливим, ніж вчити доказувати і перевіряти, є необхідність вчити догадуватися, що треба доказувати та перевірити і як знайти це доведення і спосіб перевірки, формувати вміння прогнозувати [1].

Як показують наведені приклади, застосовуючи дослідницький метод навчання, можна вирішити низку спеціальних педагогічних завдань: прищеплювання учням інтересу до навчальних і наукових досліджень; збагачення творчих спроможностей учнів на основі формування їхнього дослідницького досвіду; вивчення та аналіз впливу дослідницького досвіду учнів на їх інтелектуальний розвиток і виховання.

Література

1. Пойа Д. Математическое открытие / Пер. с англ. – М.: Наука, 1976. – 448 с.
2. Селье Г. От мечты к открытию: Как стать ученым / Пер. с англ. Общ. ред. М.Н. Кондрашовой, И.С. Хорола. – М.: Прогресс, 1987. – 366 с.
3. Информатика: Базовый курс / С.В. Симонович и др. – СПб: Питер, 2001. – 640 с.

МЕТОДЫ ОЦЕНКИ ВЛИЯНИЯ СОВРЕМЕННЫХ КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ НА ЧЕЛОВЕКА

Г.Ю. Маклаков

г. Севастополь, Севастопольский национальный технический университет
gmkromab@sevgtu.sebastopol.ua

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.

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 [1, 4, 6-7] позволяет сделать вывод о том, что современные компьютерные технологии могут представлять реальную угрозу для здоровья человека. Установлено, что интенсивное информационное воздействие вызывает измененные состояния сознания [1, 6, 8, 9]. Происходит изменение статуса личности (человек неадекватно оценивает себя и свои возможности) и статуса сознания.

Таким образом, исключительную важность приобретают исследования механизмов информационного влияния на личность и практические разработки по объективизации такого воздействия.

Выделим только два вида (обычно доминирующих) информационного воздействия на человека.

Информационно-психологическое воздействие. Речь идет о воздействии на индивидуальное сознание человека как гражданина, как субъекта политической жизни, обладающего духовны-

ми идеалами и ценностными установками. Поведение человека в обществе может принимать как острые формы политического экстремизма, так и вялые формы политического равнодушия, в не меньшей степени сказывающиеся на общественной и личной жизни людей.

Информационно-психосоматическое воздействие. Воздействии, которое прямо угрожает физическому или психическому здоровью человека. Источниками такого воздействия являются психоделическая музыка, особые мультимедийные компьютерные программы, Интернет-наркомания (Internet-addiction) и другие информационные технологии, формирующие измененные состояния сознания. Такие воздействия на протяжении многих лет создают специфическую морально-психологическую атмосферу в обществе, питают криминальную среду, способствуют росту психических заболеваний. Как результат – социальная и личностная дезадаптация, а в ряде случаев – разрушение психики человека.

Для оценки уровня воздействия первого вида с успехом может использоваться социологическое и психологическое тестирование (САН, Люшера, ЦОЕ и др.). Представляет интерес социометрические исследования структуры взаимоотношений в группе.

В настоящее время более актуальным является разработка методов оценки информационно-психосоматического воздействия на сознание и подсознание человека. Для оценки уровня и характера воздействия этого вида можно выделить следующие группы методов:

- методы классической медицины (ЭЭГ, ЭКГ, РЭГ, кардиоритмография, вариационная реография, анализ реологических характеристик крови и т.п.);
- методы «нетрадиционной» медицины (КГР в различных модификациях, газоразрядная визуализация и т.п.);
- биофизические датчики (биосенсоры, двойные электрические слои, р-п переходы, изменения плоскости поляризации света и т.п.);
- психосемантические методы;
- статистическо-информационные методы (оценка устойчивости гомеостаза организма с использованием шенноновского

определения энтропии).

Из первой группы методов можно выделить достаточно информативный метод вариационной реографии [3]. Он способен объективизировать информационное воздействие на человека путем анализа устойчивости системы биорегулирования подсистем организма на основе статистической оценки характеристик центральной и региональной гемодинамики. Дальнейшее развитие этот подход получил на основе разработанных методов оценки устойчивости гомеостаза организма с использованием шенноновского определения энтропии [4].

Из второй группы методов наиболее перспективным является метод газоразрядной визуализации. Однако отсутствие обоснованных математических подходов к оценке информационного воздействия на организм сдерживает развитие этого метода.

Сущность третьей группы методов заключается регистрации изменений происходящих в различных средах и физических объектах, на которые осуществляется внешнее информационное воздействие. В этом направлении достигнуты определенные успехи (двойные электрические слои [2, 5], p-n переходы [2]), однако возникают труднопреодолимые проблемы выбора единиц эталона измерений, подбора адекватных методов верификации и, самое главное, не учитывается конкретное состояние биосистемы, на которое осуществляется воздействие (проблема нормы и патологии). Наиболее перспективным направлением в этой группе методов является направление, развиваемое Бобровым А.В. [2, 5].

Особый интерес представляют психосемантические методы (методы лингвистического и особенно психолингвистического тестирования). В этой группе можно выделить научные направления, успешно развиваемые Шалаком В.И., Черепановой И.Ю., Спиваком Д.Л. и Спиваком Л.И. Анализ работ и практических разработок вышеприведенных авторов позволяет выделить работы Спивака Д.Л. и Спивака Л.И., посвященные изучению психологических и лингвистических аспектов формирования измененных состояний сознания у человека [8, 9]. Как указывалось в начале статьи, интенсивное информационное воздействие вызывает измененные состояния сознания. При неконтролируемом использовании информационных технологий создаются весьма

благоприятные условия для формирования особой наркотической зависимости. Поэтому разработка методов психолингвистического тестирования применительно к анализу информационного воздействия на человека представляются достаточно актуальной.

Методологическим принципом исследования информационного воздействия на человека должен являться системный, комплексный подход к анализу психосоматических структур человека. На таком подходе основана пятая группа методов [3, 4]. В этих методах организм человека рассматривается как сложная биокibernетическая саморегулирующаяся система, функционирующая в условиях нелинейности. На биосистему поступает некоторое информационное возмущающее воздействие, которое выводит его из режима нормального функционирования. Необходимо определить степень нарушения функционирования психосоматических структур человека. Метод объективизации информационных воздействий на человека реализуется путем комплексной оценки устойчивости гомеостаза организма на основе рассмотрения в динамике показателей центральной и региональной гемодинамики с учетом реологических характеристик крови и упругих свойств стенок различных сегментов сосудов. По полученным параметрам определяли степень неопределенности функционирования биосистемы с использованием шенноновского определения энтропии [4].

Развитие и распространение информационных технологий поставило перед кибернетиками, психологами, медиками, биологами важный вопрос о необходимости раскрыть механизмы влияния информационных технологий на психику и разработать действенные методы лечения и профилактики. По мере развития компьютерных технологий значимость решения этой проблемы будет все возрастать.

Сторона взаимодействия человека с компьютером – проблема психологических последствий информатизации заслуживает самого серьезного внимания. 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.

Литература

1. Актуальные проблемы информационного противоборства. / Сб. статей Московской академии комплексной безопасности предпринимательства. Под общ. редакцией проф. А.И. Гурова. – М.: Изд-во МГУЛ, 1999. – 333 с.
2. Бобров А.В., Шрайбман Ф.О. Использование интегральных микросхем в качестве датчика для биофизических исследований. / В сб.: «Информационные взаимодействия в биологии». Матер. Всесоюзного семинара в Кара-Даге в октябре 1986 г.. – Тбилиси: Изд. ТГУ, 1987. – С. 56-69.
3. Маклаков Г.Ю. Методика вариационной реоэнцефалографии в диагностике сосудистых заболеваний мозга. / В кн: Актуальные вопросы теоретической и клинической медицины. – Ч.2. – Днепропетровск: ДМИ, 1993. – С. 94-95.
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. Музалевская Н.И., Бобров А.В. О возможной роли двойных электрических слоев в реакции биологических объектов на внешние воздействия. – Биофизика. – 1988. – Т.33, №4. – С. 725.
6. Прокофьев В.Ф. Тайное оружие информационной войны. – М.: СИНТЕГ, 1999. – 152 с.
7. Расторгуев С.П. Информационная война. – М.: Радио и связь, 1998. – 416 с.
8. Спивак Д.Л. Измененные состояния сознания: психология и лингвистика. – СПб.: “Издательский Дом Ювента”; филологический факультет СПбГУ, 2000. – 296 с.
9. Спивак Л.И. Измененные состояния сознания у здоровых людей (постановка вопроса, перспективы исследований) // Физиология человека. – 1988. – № 1. – С. 138-147.

ПЕРСПЕКТИВЫ ИСПОЛЬЗОВАНИЯ СОВРЕМЕННЫХ МИКРОКОНТРОЛЛЕРОВ ДЛЯ СОЗДАНИЯ ПОРТАТИВНЫХ УСТРОЙСТВ МУЛЬТИМЕДИЙНОГО ОТОБРАЖЕНИЯ УЧЕБНОЙ ИНФОРМАЦИИ

Г.Ю. Маклаков, Е.А. Кожаяев, Г.Г. Маклакова
г. Севастополь, Севастопольский национальный технический
университет
gmkromab@sevgtu.sebastopol.ua

Одним из наиболее мощных факторов, позволяющих развить образную сторону мышления, является визуализация основных математических и физических понятий, процессов и явлений при помощи компьютера [1]. Визуализация как элемент современных информационных технологий позволяет совершенно по-новому подойти к методике преподавания в средней и высшей школе. 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. Использование современных мультимедийных технологий в учебном процессе во многом замедляется из-за высокой стоимости компьютеров, на которых они, как правило, и реализуются.

Одним из прогрессивных направлений компьютерных технологий является разработка специализированных аппаратно-программных комплексов, призванных решать конкретные задачи. На кафедре кибернетики и вычислительной техники Севастопольского национального технического университета ведется разработка специализированных портативных микропроцессорных комплексов, ориентированных на мультимедийное отражение учебной информации. Для выбора элементной базы специализированного портативного устройства были проанализированы существующие технологии создания мультимедийных комплексов.

Учитывая, что проектируемый комплекс должен быть малогабаритным и переносным, были выдвинуты следующие требования к архитектуре устройства: память должна быть энергонезависимой, в качестве центрального процессора необходимо использовать специализированный микропроцессор, в качестве устройства вывода – жидкокристаллический дисплей.

Проведенный анализ публикаций по данной теме позволил сделать вывод о том, что наиболее рациональным подходом для использования в проектируемом комплексе является Flash-память. Обзор существующих микросхем Flash-памяти показал, что для проектируемого комплекса предпочтительно использовать CMOS SEEPR0M семейства AT24 разработки фирмы Atmel.

Проведенный обзор существующих протоколов обмена данными позволил выделить перспективное направление – шину с последовательным способом передачи данных I²C. Шина I²C («Inter-Integrated Circuit») разработана фирмой Philips для подключения дополнительных устройств по двум линиям, обеспечивает двунаправленный обмен данными на высокой скорости (до 100 кбит/сек). Она представляет собой двунаправленную асинхронную шину с последовательной передачей данных и возможностью адресации до 128 устройств. Общие характеристики шины I²C:

- скорость обмена – до 100 кбит/сек для стандартного режима;
- максимальная длина линий – не более 2-х метров;
- суммарная емкость линий – не более 400 пФ;
- входная емкость на каждую ИС – не более 10 пФ.

В качестве специализированных микропроцессоров целесообразно использовать микроконтроллеры PIC-micro разработки фирмы Microchip [4]. Они объединили в себе все передовые технологии, применяемые в производстве микроконтроллеров: развитую RISC-архитектуру, малое энергопотребление, высокое быстродействие, ППЗУ, программируемое пользователем [4, 5].

Основными факторами популярности микроконтроллеров PIC-micro является их высокая надежность, очень низкая цена, низкое потребление питания. Проведенный литературный обзор и анализ сайтов, представленных в Интернет, позволяет утвер-

ждать, что оптимальным решением относительно выбора PIC-микроконтроллера является использование многофункциональных микроконтроллеров среднего и старшего семейства PIC16F и PIC18F [5].

Анализ существующих жидкокристаллических дисплеев показал [3], что для разработки специализированных портативных микропроцессорных комплексов, ориентированных на мультимедийное отображение учебной информации целесообразно использовать жидкокристаллические индикаторы фирмы Amprige. Вся продукция фирмы сертифицирована по международному стандарту качества ISO 9002. Фирма производит ЖКИ модули по нескольким технологиям STN, FSTN, TAB, COF и по новой технологии COG (чип на стекле), а также различные по назначению – символьные, графические и др. Важным достоинством символьных индикаторов является наличие в них встроенных микроконтроллеров с русифицированным знакогенератором.

Выбранная стратегия создания специализированных аппаратно-программных комплексов для мультимедийного отображения учебной информации позволила начать практические разработки таких устройств. Для успешного внедрения в учебный процесс необходимо, чтобы портативные устройства изготавливались с самыми разнообразными функциональными возможностями и соответственно с разной стоимостью устройства. Для реализации такого подхода разработка портативных устройств ведется в трех направлениях (деление на классы условные, т.к. в каждом классе реализуются мультимедийные обучающие системы, только с разными возможностями):

- устройство с минимальными мультимедийными возможностями (класс «электронная книга»);
- устройство со средними мультимедийными возможностями (класс «электронная учебник»);
- устройство с максимальными мультимедийными возможностями (класс «обучающая система»).

Кроме вышеприведенной классификации предусматривается в каждой серии разработка устройств, существенно отличающихся по объему хранимой учебной информации. Такой подход позволяет создать устройства с достаточно широким разбросом стоимости каждого устройства (от 70-80 грн. – минимальная

конфигурация, до 550-600 грн. – максимальная конфигурация).

Более подробное описание одного из устройств представлено в настоящем сборнике (статья Кожаев Е.А., Маклакова Г.Г., Маклаков Г.Ю. Специализированный аппаратно-программный комплекс «электронная книга»).

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 [2].

Литература

1. Маклаков Г.Ю. Особенности использования мультимедийных технологий в преподавании фундаментальных дисциплин в высшей школе. / 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. – С. 211-214.
2. Маклаков Г.Ю. Современные компьютерные технологии и информационная безопасность личности / 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. – С. 215-218.
3. Самарин А.В. Жидкокристаллические дисплеи. Схемотехника, конструкция и применение. – М.: Солон, 2002. – 304 с.
4. Тавернье К. PIC-микроконтроллеры. Практика применения. – М.: ДМК Пресс, 2003. – 272 с.
5. Яценко В.С. Микроконтроллеры Microchip. Практическое руководство. – М.: Горячая линия–Телеком, 2002. – 296 с.

РЕАЛІЗАЦІЯ АЛГОРИТМІВ МАТЕМАТИЧНИХ ОБЧИСЛЕНЬ В СИСТЕМІ MATHCAD

Ю.С. Матвієнко

м. Полтава, Полтавський державний педагогічний університет
імені В.Г. Короленка
uuri@rambler.ru

Мільйони людей займаються математичними розрахунками в силу професійної або іншої необхідності, не говорячи вже про навчання студентів математичних спеціальностей. Жодна серйозна розробка в будь-якій галузі науки не обходиться без математичних розрахунків. Для їх проведення використовуються програми, складені з використанням конструкцій мов високого рівня (таких як Fortran, Pascal, C та ін.). Вивчаються ці мови програмування студентами-інформатиками, математиками в межах різних дисциплін. Але для математичних розрахунків, які потребують швидкого отримання результату, складання таких програм ускладнює і дещо сповільнює процес.

Заслужену популярність ще в середині 80-х років отримали інтегровані системи для автоматизації математичних розрахунків класу MathCAD, розроблені фірмою MathSoft (США) [1, 2]. І зараз вони залишаються такими математичними пакетами, в яких опис розв'язування математичних задач дається за допомогою звичних математичних формул і знаків. Такий же вигляд мають і результати обчислень.

В останніх версіях MathCAD [1-3] користувачам надана можливість складати “власні” програми-функції і використовувати принципи модульного програмування для реалізації оригінальних обчислювальних алгоритмів користувача. Однак в літературі ці можливості описані зовсім слабо. Навіть при вивченні MathCAD в межах різних математичних та інформаційних дисциплін програмуванню майже зовсім не приділяється ніяка увага.

Наведемо елементарний приклад вивчення та використання можливостей програмування в MathCAD.

Реалізувати той або інший алгоритм обчислення в пакеті MathCAD можна двома способами:

- вставляючи відповідні оператори або функції в текст документа MathCAD. Такий спосіб називається *програмуванням в тексті документа*;
- використовуючи так звані програми-функції, які містять конструкції подібні конструкціям таких мов як Pascal або Fortran: оператори присвоєння, оператори циклів, умовні оператори та ін. Створення програм-функцій в MathCAD дозволяє розв'язати задачі, які неможливо розв'язати, використовуючи лише оператори та функції MathCAD. Такий спосіб називається *програмуванням в програмі-функції*. Таке програмування включає два етапи:
 - опис програми-функції;
 - виклик програми-функції.

Перед тим, як використати програму-функцію, її необхідно задати, тобто виконати опис. Опис програми-функції розміщується в робочому документі перед викликом програми-функції і включає в себе ім'я програми-функції, список формальних параметрів (який може бути відсутніми) та тіло програми-функції. Зупинимося трохи на цих поняттях.

Кожна програма-функція MathCAD має *оригінальне ім'я*, використовуючи яке реалізується звернення до цієї програми-функції. Через це ж ім'я повертається в робочий документ результат виконання програми-функції.

Після імені програми-функції в круглих дужках іде *список формальних параметрів*. Через формальні параметри в програму-функцію передаються необхідні для виконання обчислень дані. В якості формальних параметрів можуть використовуватися імена простих змінних, масивів і функцій.

Тіло програми-функції включає будь-яку кількість операторів локальних, а також виклик інших програм-функцій і функцій користувача.

Для введення в робочий документ опису програми-функції необхідно виконати наступні дії:

- ввести ім'я програми-функції і список формальних параметрів;
- ввести символ “:” – на екрані відображається як “:=”;
- відкрити панель *Програмування* і натиснути кнопку

Add line. На екрані з'явиться вертикальна риска і вертикальний стовпчик з двома полями для введення операторів (рис. 1).

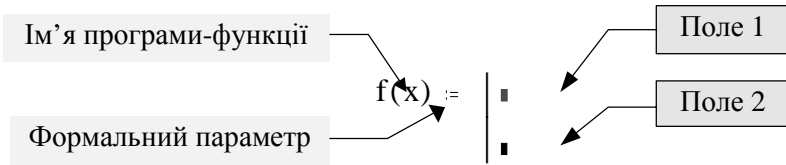


Рис. 1. Структура програми-функції.

- перейти в поле 1 і ввести перший оператор тіла програми-функції. Поля введення для додаткових операторів відкриваються за допомогою кнопки *Add line* панелі програмування (див. рис.2);

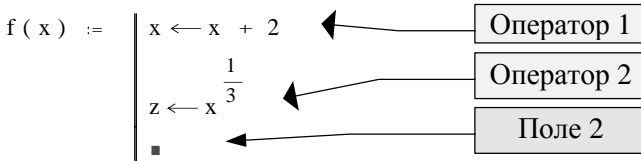


Рис. 2. Додавання операторів в тіло програми-функції.

- заповнити най нижнє поле введення (поле 2), увівши туди вираз, який повертає значення через ім'я програми-функції.

В наведеному прикладі формальним параметром є проста змінна x , тіло програми містить два локальних оператори присвоєння і значення змінної z визначає повернення результату виконання програми-функції через ім'я функції.

$$f(x) := \left| \begin{array}{l} x \leftarrow x + 2 \\ \\ z \leftarrow x \frac{1}{3} \\ z \end{array} \right.$$

Рис. 3. Кінцева структура програми-функції

Наведемо приклад завдання, яке може розв'язуватися студентами при вивчення програмування в системі MathCAD.

Приклад. Реалізуйте у вигляді програми-функції обчислення визначеного інтегралу виду

$$\int_a^b f(x)dx,$$

використовуючи формулу Сімпсона з автоматичним вибором числа вузлів. При цьому програма-функція Simpson(f, a, b, N) повинна обчислювати визначений інтеграл за формулою Сімпсона при фіксованій кількості інтервалів N, а програма-функція Adapt(f, a, b) повинна вибирати із заданою точністю обчислення інтеграла (рівній 10^{-8}) кількість інтервалів.

$$\text{Simpson}(f, a, b, N) := \left| \begin{array}{l} h \leftarrow \frac{b - a}{N} \\ S \leftarrow f(a) + f(b) \\ \text{for } i \in 0..N - 1 \\ \quad S \leftarrow S + 4 \cdot f\left(a + i \cdot h + \frac{h}{2}\right) \\ \text{for } i \in 1..N - 1 \\ \quad S \leftarrow S + 2 \cdot f(a + i \cdot h) \\ \frac{h}{6} \cdot S \end{array} \right.$$

$$\text{Adapt}(f, a, b) := \left| \begin{array}{l} \text{eps} \leftarrow 10^{-8} \\ I1 \leftarrow \text{Simpson}(f, a, b, 5) \\ I2 \leftarrow \text{Simpson}(f, a, b, 10) \\ I2 \text{ if } |I1 - I2| < \text{eps} \\ \text{Adapt}\left(f, a, \frac{a+b}{2}\right) + \text{Adapt}\left(f, \frac{a+b}{2}, b\right) \text{ otherwise} \end{array} \right.$$

Використовуючи ці програми-функції, обчислюємо визначений інтеграл від функції $f(x)=x^2$ на відрізку $[0, 1]$. Точне значення інтегралу рівне $1/3=0,(3)$. Звернення до програми-функції Adapt дає результат $\text{Adapt}(f, 0, 1)=0.3333333333333333$.

Перед зверненням до програми-функції Adapt необхідно описати функцію користувача у вигляді

$$f(x) := x^2,$$

так як ім'я функції $f(x)$ використовується в якості фактичного параметру.

Отже, як видно з попередніх прикладів та опису можливостей програмування в системі MathCAD, на основі складання програм-функцій можна вивчати не тільки математичний пакет, але і використовувати його при вивченні чисельних методів, теорії алгоритмів, під час обчислювальної практики тощо.

Література

1. MathCAD 6.0 Plus. Руководство пользователя. Пер. с англ. – М.: Филинъ, 1996. – 712 с.
2. Дьяконов В.П. Справочник по MathCAD 6.0 Plus Pro. – М.: СК Пресс, 1997. – 336 с.
3. Дьяконов В.П., Абраменкова И.В. MathCAD 7.0 в математике, физике и в Internet. – М.: Нолидж, 1998. – 352 с.
4. Дьяконов В.П. Компьютерная математика: теория и практика. – М.: Нолидж, 2001. – 1296 с.

РАБОТА С ФИНАНСОВЫМИ ФУНКЦИЯМИ EXCEL

О.А. Медведева

г. Краматорск, Донбасская государственная

машиностроительная академия

ese@dgma.donetsk.ua

Деятельность экономиста высшей квалификации предполагает умение использовать в своей работе базовые модели и методы финансово-экономических расчетов. Проведение этих расчетов – рутинный циклический вычислительный процесс. Использование финансовых функций EXCEL позволяет значительно сократить время расчетов и тем самым существенно повысить эффективность работы экономиста.

В качестве примера рассмотрим определение будущей стоимости. Понятие будущей стоимости основано на принципе неравноценности денег, относящихся к разным моментам времени. Вложения, сделанные сегодня, в будущем составят большую величину. В EXCEL существует функция, предназначенная для расчета будущей стоимости – БЗ. Функция БЗ (будущее значение) рассчитывает будущую стоимость периодических постоянных платежей и будущее значение единой суммы вклада или займа на основе постоянной процентной ставки.

Рассмотрим расчет будущей стоимости единой суммы вклада (табл. 1).

Табл. 1. Определение будущей стоимости единой суммы вклада

Математическая формула	Функция EXCEL	Смысловая нагрузка
$f = p(1+r)^n$ (1)	=БЗ(норма;число_периодов;;нз)	будущее значение вклада
r (в долях от 1)	норма (в %)	процентная ставка за один период начисления
n	число_периодов	общее число периодов начисления процентов
p	нз	начальное значение вклада

При решении конкретной задачи вместо названий аргументов следует записать соответствующие числа.

Рассмотрим ситуацию, когда необходимо определить будущую стоимость не единой суммы вклада, как в предыдущем случае, а будущую стоимость периодических платежей. Эти платежи могут осуществляться в начале каждого расчетного периода (так называемые платежи «пренумерандо») или в конце («постнумерандо») в течение n периодов. Допустим, что в каждом периоде вносится одинаковая сумма. Требуется найти совокупную величину таких вложений (их будущую стоимость) в конце n -го периода для обоих случаев. Отличие в расчете при этом заключается в том, что во втором случае не происходит начисления процентов на последний вклад, т.е. все вклады «пренумерандо» увеличиваются на сложные проценты на один расчетный период больше, чем вклады «постнумерандо». В этих случаях также используется функция БЗ, но расчеты производят по другим формулам (табл. 2).

Табл. 2. Определение будущей стоимости периодических платежей

Математическая формула	Функция EXCEL	Смысловая нагрузка
$f = pmt \frac{(1+r)^n - 1}{r} (1+r)$ (2)	=БЗ(норма; число_периодов; выплата;;1)	будущее значение платежа типа «пренумерандо» (обязательные платежи)
$f = pmt \frac{(1+r)^n - 1}{r}$ (3)	=БЗ(норма; число_периодов; выплата;;0)	будущее значение платежа типа «постнумерандо» (обычные платежи)
pmt	выплата	фиксированный регулярный взнос

Замечание. Здесь и в дальнейшем значения параметров, смысловая нагрузка которых не изменялась, повторно не описываются.

Рассмотрим *типовой пример 1*.

Рассчитаем, какая сумма окажется на счете, если 15000 грн. положены на 3 года под 15% годовых. Проценты начисляются ежемесячно.

Решение:

Для расчета применяется формула (1), т.к. требуется найти будущее значение единой суммы вклада. Необходимо обратить внимание на то, что в условии задачи указан *годовой* процент и число *лет*. Если проценты начисляются несколько раз в год, то необходимо рассчитать общее количество периодов начисления процентов и ставку процента за период начисления. Эти величины легко определить по таблице 3, в которой приводятся расчеты для наиболее распространенных методов начисления процентов в году.

Табл. 3. Расчет основных величин при внутригодовом учете процента

Метод начисления процентов	Общее число периодов начисления процентов	Ставка процента за период начисления, %
ежегодный	n	r
полугодовой	$n \cdot 2$	$r/2$
квартальный	$n \cdot 4$	$r/4$
месячный	$n \cdot 12$	$r/12$
ежедневный	$n \cdot 365$	$r/365$

Таким образом, в данном примере при ежемесячном учете процента общее число периодов начисления равно $3 \cdot 12$ (аргумент *число_периодов*), а процент за период начисления равен $15\%/12$ (аргумент *норма*). По условию аргумент *нз* равен -15000 . Это отрицательное число, означающее вложение денег. Используя функцию БЗ, получим

$$=БЗ(15\%/12; 3 \cdot 12; -15000) = 23459,16 \text{ грн.}$$

Проведем для сравнения расчет по формуле (1):

$$f = 15 \cdot (1 + 0,15/12)^{3 \cdot 12} = 23459,16 \text{ грн.}$$

Рассмотрим *типовой пример 2*.

Предположим, есть два варианта инвестирования средств в течение 5 лет: в начале каждого года под 18% годовых или в

конце каждого года под 20% годовых. Пусть ежегодно вносится 50000 грн. Определим, сколько денег окажется на счете в конце 5-го года для каждого варианта.

Решение:

В данном случае производятся периодические платежи, и расчет ведется по формуле (2) для первого варианта (обязательные платежи) и по формуле (3) для второго (обычные платежи).

В первом случае будущая стоимость платежей составит

$$f = 50000 \frac{(1 + 0,18)^5 - 1}{0,18} (1 + 0,18) = 422098,38 \text{ грн.},$$

а во втором –

$$f = 50000 \frac{(1 + 0,2)^5 - 1}{0,2} = 372080 \text{ грн.}$$

При работе с функцией БЗ следует указать аргументы *норма* = 18%, *число_периодов* = 5, *выплата* = -50000 для первого варианта; *норма* = 20% для второго варианта.

Тогда

БЗ(18%;5;-50000;;1) = 422098,38 – для первого варианта,

БЗ(20%;5;-50000;;0) = 372080 – для второго варианта.

Расчеты показали, что первый вариант предпочтительнее.

Ниже предлагаются задания для самостоятельной работы студентов по данной теме.

1. Рассчитайте, какая сумма будет на счете, если сумма размером 5000 тыс. грн. размещена под 12% годовых на 3 года, а проценты начисляются каждые полгода.

2. По вкладу размером 2000 тыс. грн. начисляется 10% годовых. Рассчитайте, какая сумма будет на сберегательном счете через 5 лет, если проценты начисляются ежемесячно.

3. На сберегательный счет вносятся платежи по 200 тыс. грн. в начале каждого месяца. Рассчитайте, какая сумма окажется на счете через 4 года при ставке процента 13.5% годовых.

4. Определите, какая сумма окажется на счете, если вклад размером 900 тыс. грн. положен под 9% годовых на 19 лет, а проценты начисляются ежеквартально.

5. Какая сумма должна быть выплачена, если шесть лет назад была выдана ссуда 1500 тыс. грн. под 15% годовых с ежемесячным начислением процентов.

6. Взносы на сберегательный счет составляют 200 тыс. грн. в начале каждого года. Определите, сколько будет на счете через семь лет при ставке процента 10%.

7. Рассчитайте, какая сумма будет на счете, если вклад размером 5000 тыс. грн. положен под 12% годовых на три года, а проценты начисляются каждые полгода.

8. Вклад размером 2000 тыс. грн. положен под 10% годовых. Рассчитайте, какая сумма будет на сберегательном счете через пять лет, если проценты начисляются ежемесячно.

9. На сберегательный счет вносятся обязательные ежемесячные платежи по 200 тыс. грн. Рассчитайте, какая сумма окажется на счете через четыре года при ставке процента 13.5% годовых.

10. По вкладу размером 5000 тыс. грн. начисляется 5% годовых. Рассчитайте, какая сумма будет на сберегательном счете через 3 года, если проценты начисляются ежеквартально.

Литература

1. Четыркин Е.М. Методы финансовых и коммерческих расчетов. – 2-е изд., испр. и доп. – М.: Дело ЛТД, 1995. – 320 с.

МЕТОДИКА ПРЕПОДАВАНИЯ КОМПЬЮТЕРНЫХ ДИСЦИПЛИН СТУДЕНТАМ СПЕЦИАЛЬНОСТИ «ЭКОНОМИЧЕСКАЯ КИБЕРНЕТИКА»

А.Ю. Мельников

г. Краматорск, Донбасская государственная
машиностроительная академия
alexandr.melnikov@dgma.donetsk.ua

В последнее время при изложении студентам экономических специальностей компьютерных дисциплин наблюдается стойкая тенденция увеличения объема изложения офисных приложений, электронных таблиц, стандартных систем управления базами данных и т.п. за счет изучения основ алгоритмизации и языков программирования высокого уровня. Для «обычных» экономистов такой подход отчасти оправдан, однако для будущих специалистов в области экономической кибернетики совершенно не приемлем. Ибо им как при изучении специализированных дисциплин на старших курсах, так и в дальнейшей работе крайне необходимо владение именно средствами создания собственных полнофункциональных приложений, а также применением «программного подхода» к анализу экономических проблем. В то же время нельзя слепо копировать рабочие программы технических специальностей (таких, как «Информационные технологии» и пр.) из-за экономической направленности специальности.

На кафедре экономической кибернетики Донбасской государственной машиностроительной академии в течение нескольких лет разрабатывалась, тестировалась и сейчас принята следующая концепция обучения студентов. Она базируется на двух составляющих: во-первых, предполагается, что поступившие на «ЭК» студенты уже имеют определенный опыт работы с современными компьютерами в соответствующих операционных системах; во-вторых, согласно учебной программе специальности, основные дисциплины по направлению начинают преподаваться, начиная с 3-го курса. Поэтому задача первых двух курсов (дисциплины «Компьютерная техника и программирование», «Алгоритмизация и программирование») – изложение студентам основополагающих знаний в области информатики и программиро-

вания.

Первый семестр первого курса можно назвать «приведением к общему знаменателю» знаний, полученных в школе. Предполагается, что студенты знакомы с базовыми понятиями алгоритмизации и знают основы какого-нибудь языка программирования высокого уровня (как правило, это Бейсик или Паскаль). Содержание первого семестра – это «Паскаль с самого начала»: алфавит языка, данные, ввод-вывод, линейный, условный и циклический вычислительные процессы, массивы (одномерные и двумерные), подпрограммы, строки, записи, файлы числовые, текстовые и типа запись, причем именно в такой последовательности, от простого – к сложному. Помимо практических работ, где происходит закрепление знаний Паскаля, предусматривается самостоятельная работа по освоению ряда возможностей ОС Windows и текстового процессора MS Word (все работы обеспечены методическими указаниями).

Второй семестр состоит из трех блоков-«модулей» и предусматривает: овладение процессором электронных таблиц MS-Excel; изучение дополнительных возможностей Турбо Паскаля в области работы с динамическими переменными и построения графических изображений; изучение основ работы в среде визуального программирования Borland Delphi. Также студенты изучают некоторые алгоритмы сортировки данных, динамические структуры (стеки, очереди, связные списки) и основы объектно-ориентированного программирования. В рамках самостоятельной работы выполняются расчетно-графические работы, органично увязывающие все получаемые знания (например, предлагается создать интегрированную систему обработки данных, в которой бы расчетная часть была представлена Паскаль-программой, а результаты работы визуализировались бы в Excel и т.п.).

Третий семестр состоит из двух блоков-«модулей» и предусматривает: изучение основ строения всемирной сети Интернет и языка разметки гипертекста HTML, включая начальные сведения о JavaScript; освоение специальных пакетов для математических расчетов (MathCAD, Maple и т.п.). Первый блок практически совпадает с тем, что предлагается другим специальностям, а вот второй имеет отличительные особенности. Студентам не просто

излагаются принципы работы математических пакетов: от них требуется каждую задачу решить с применением как минимум четырех программных инструментов – двух новых (MathCAD, Maple) и двух уже известных (Excel, Паскаль или Delphi). Таким образом, они могут, во-первых, прочувствовать все преимущества и недостатки каждого пакета, а во-вторых, «не забыть» знания и умения, приобретенные на первом курсе. Рассматриваются задачи от простых математических (исследование функции и построение ее графика, решение систем уравнений и т.п.) до межотраслевой балансовой модели (модель Леонтьева) и различных задач линейного программирования.

Четвертый семестр полностью посвящен изучению языка программирования Си. Это вызвано, во-первых, необходимостью знаний нескольких языков программирования, а во-вторых, понимаем того факта, что его синтаксис вошел во многие широко используемые сейчас языки – такие, как Perl, Java, JavaScript и другие. Программа почти полностью повторяет разделы первого семестра первого курса, за тем лишь исключением, что основы даются менее подробно, а отдельные специфические моменты – наоборот, детально.

К третьему курсу студент-экономкибернетик обладает достаточными знаниями и навыками работы, чтобы приступить к изучению специализированных дисциплин, не отвлекаясь на частности. На четвертом-пятом курсах его ждет завершающая «компьютерно-программный» цикл дисциплина – «Объектно-ориентированное программирование», где, помимо теоретической части собственно ООП, его ждет изучение работы в Delphi с базами данных и JavaScript.

САМОСТІЙНА РОБОТА СТУДЕНТІВ В УМОВАХ СИСТЕМНОЇ ІНТЕГРАЦІЇ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

О.І. Миронова

м. Луцьк, Волинський державний університет ім. Лесі Українки
mirelena@ukr.net

Відповідно до Положення про організацію навчального процесу у вищих навчальних закладах, “самостійна робота студентів є основним засобом оволодіння навчальним матеріалом у час, вільний від обов’язкових навчальних занять. Самостійна робота забезпечується системою навчально-методичних засобів, передбачених для вивчення конкретної навчальної дисципліни”. Характерною особливістю сучасного етапу розвитку вищої школи в Україні є посилення уваги до активізації самостійної роботи студентів, її ефективності щодо засвоєння змісту дисципліни. Динамізм зміни технологій, знань, інформації, які відбуваються у світі, потребують змін і у навчальному процесі, зокрема, не тільки надавати студентам певну суму знань і навичок, а і навчати їх самостійно вчитися, самостійно отримувати інформацію, використовувати її у повсякденній діяльності. Студент при цьому активно вибудовує свій власний навчальний процес, формуючи власну траєкторію в освітньому середовищі. Комп’ютерні і телекомунікаційні технології дозволяють організовувати самостійну роботу на якісно новому рівні, передбачаючи при цьому розвиток навчальних курсів, їх гнучкість і диверсифікацію.

Найбільш розповсюджений вид самостійної роботи (через доступність засобу) – це робота з навчальною літературою. Останнім часом популярним стає використання електронних підручників, мультимедіа-посібників, інших інтерактивних засобів, що пояснюється активним розвитком зазначених технологій і зручністю їх використання [6].

Застосування навчаючих систем, електронних навчальних видань для організації навчального процесу з дисциплін, в основі яких лежать інформаційні технології, в різних формах навчання потребує постійного поновлення всіх компонент фахової підготовки – змісту теоретичного матеріалу та набору практичних за-

вданий, способів представлення та методів організації їх виконання, комплексних форм перевірки та контролю знань і умінь студентів.

Засобом ефективного досягнення названих цілей є системна інтеграція програмного забезпечення навчального та професійного призначення, Інтернет-технологій. Така системна інтеграція передбачає цілеспрямоване об'єднання існуючих та створюваних інформаційних проектів у цілісну систему, яка забезпечуватиме задані вимоги як елементів навчального процесу, так і умов їх взаємодії. При цьому сукупність інформаційних технологій супроводжуватиме студента-інформатика з початку навчання до досягнення ним професійного рівня і навіть в межах післявузівської освіти.

В контексті означених проблем організації самостійної роботи студентів проводиться дослідження, метою якого є проектування і реалізація інформаційно-навчаючого середовища та розробка методики його застосування.

Створена версія системи проблемно орієнтована на ІТ-дисципліни (наприклад, "Програмне забезпечення ЕОМ"). Основне її призначення – допомогти студенту без участі викладача вивчати основні теоретичні питання, що стосуються конкретного програмного продукту, виконувати практичні завдання в реальному програмному середовищі із фіксуванням результатів такої роботи та відповідей на тести.

Програма складається з модулів, які реалізують заняття наступних типів: лекція (перегляд частин теоретичного (текст, графіка, відео, звук) матеріалу з можливістю виходу у середовище, яке розглядається), практичне заняття (виконання завдань у середовищі, що розглядається, з контролем з боку системи), тестування (перелік питань, на які потрібно дати відповідь, в результаті – оцінка та аналіз), комплексна робота, яка об'єднує елементи занять попередніх типів.

При організації роботи з системою реалізована можливість вибору:

- теми (залежно від режиму роботи користувач або самостійно здійснює вибір, або система контролює даний процес і перехід від теми до теми буде здійснено лише після опрацювання попередньої (за результата-

- ми тестування));
- видів занять;
- рівнів складності (аналогічно до вибору теми).

Під час роботи з системою передбачається постійний збір інформації про користувача (обов'язкова реєстрація). На основі цих статистичних даних системою кожен раз пропонуватиметься траєкторія проходження курсу. Результати статистичного збору відображаються на екрані, при потребі – виводяться на друк.

Для забезпечення потреб конкретного викладача у системі передбачається можливість власного конструювання занять. На основі запропонованих форм можна здійснити: створення лекційного заняття; створення бази тестових питань; підключення до системи файлів практичної роботи; створення комплексного заняття.

При проектуванні інтерфейсу та його основного елементу – діалогу дотримано основних психолого-педагогічних вимог та передбачено можливість настроювання інтерфейсу (розмірів шрифтів, кольорів тощо).

Для зручної роботи, оперативного пошуку інформації та ознайомлення з додатковою інформацією (списком літератури, словником і т.д.) проектуються навігаційна, пошукова, довідкова системи.

Організація самостійної діяльності студента по засвоєнню конкретного матеріалу здійснюється з урахуванням вимог теорії поетапного формування розумових дій [4].

Мета роботи студента із системою полягає у засвоєнні конкретної теми, що може виступати мотивацією для нього. Додатковими елементами тут є: інтерес студента ІТ-спеціальності до комп'ютера і програмного забезпечення як об'єктів вивчення, інтуїтивно зрозумілий та зручний інтерфейс програми, які викликають позитивне сприймання нового матеріалу (перший етап формування дій і понять – мотиваційний).

На етапі роботи з програмою в режимі “Лекція” (рис. 1) студент знайомиться з основними поняттями теми, поданими концептивно, має можливість доступу до ілюстрації відповідних дій та процесів, що забезпечує побудову у свідомості студента орієнтовної основи дії.

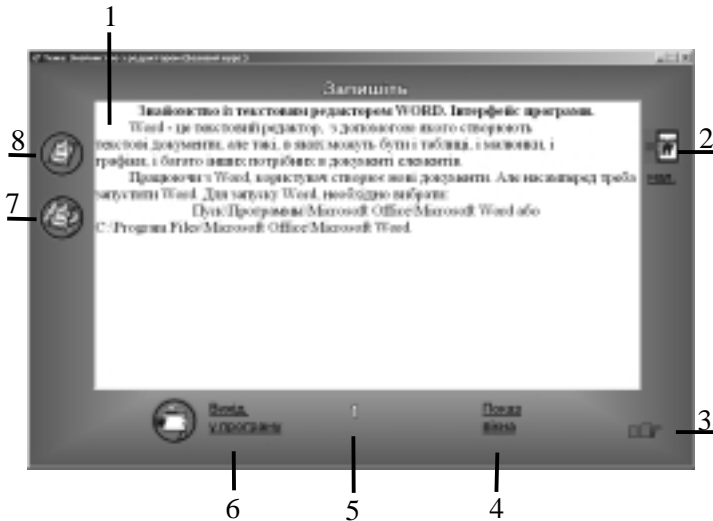


Рис. 1.

1 – поле виводу тексту; 2 – зображення піктограм; 3 – управляюча кнопка (перехід на крок вперед); 4 – можливість перегляду відео-роликів; 5 – номер поточної сторінки; 6 – можливість виходу у програмний продукт, який розглядається; 7 – перехід до вікна вибору типу заняття; 8 – перехід до вікна вибору теми заняття.

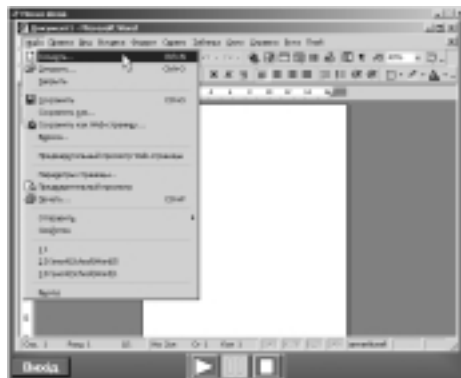


Рис. 2.

В режимі “Лекція” на екран подається невелика кількість інформації, студент сам контролює процес пересування по темі. Це

забезпечує третій етап формування поняття – отримання і закріплення відповідної інструкції, формування дії як зовнішньомовної.

Для кращого закріплення матеріалу (наступний етап – етап беззвучного усного мовлення) студенту надається можливість виходу у розглянуте середовище, де він використовує набуті знання згідно отриманої інструкції і самостійно контролює правильність їх виконання.

Детальне відпрацювання навичок роботи з програмою та осмислене використання набутих вмій (заключний етап) реалізується при роботі з програмою в режимі “Практична робота” (рис. 3).



Рис. 3.

Після завершення роботи видається результат (рис. 4).



Рис. 4.

Система контролю знань організована в режимі “Тестування” (рис. 5).

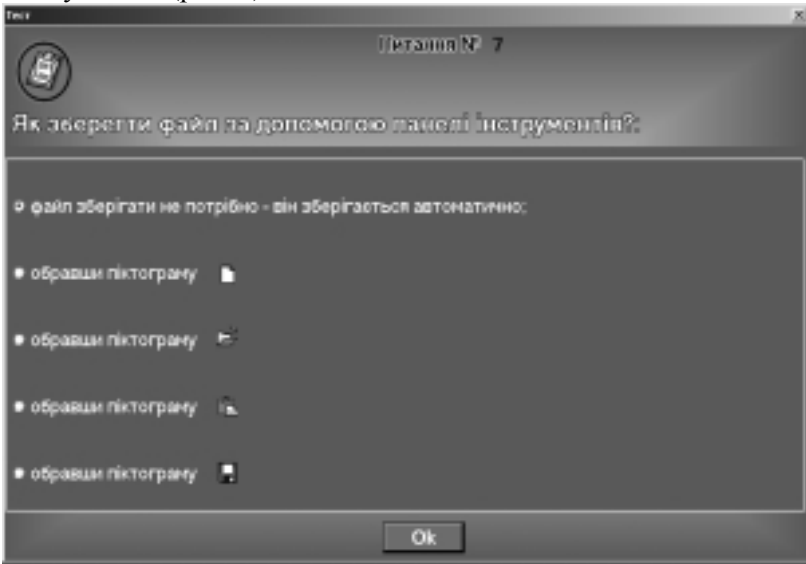


Рис. 5.

У результаті дослідження буде розроблено та введено інформаційно-навчаче середовище для вивчення курсу “Програмне забезпечення ЕОМ”. В основу створюваного проекту покладено системну інтеграцію ІТ-технологій.

Система може бути використана студентами для самостійної роботи і викладачем на лабораторних заняттях (для цього передбачена можливість переключення способів роботи програми (режим оцінювання та режим самостійного вивчення нового матеріалу і відпрацювання практичних навичок без фіксації результатів оцінювання)).

Література

1. Аленичева Е.А., Монастырев Н.Ф. Электронный учебник (проблемы создания и оценки качества). // Высшее образование в России. – 2001. – №1. – с.121-123.

2. Андреев А.А. Педагогика высшей школы (Новое издание). – М.: МЭСИ, 2002. – 192 с.

3. Миронова О.І. HelpForTeacher – мультимедійна навчальна система // Матеріали Міжнародної науково-практичної конференції студентів, аспірантів та молодих вчених “Комп’ютери. Програми. Інтернет. 2003”. – 21-23 квітня 2003 р. – Київ. – С. 47.

4. Талызина Н.Ф. Управление процессом усвоения знаний. – М.: МГУ, 1975. – 230 с.

5. <http://ito.bitpro.ru/2003/II/4/II-4-1818.html>

6. <http://www.it-education.ru/reports/sokolova.htm>

АНАЛИЗ ЦЕЛЕСООБРАЗНОСТИ ИСПОЛЬЗОВАНИЯ SQUID ДЛЯ ДОСТУПА В ИНТЕРНЕТ

А.А. Мясищев, В.М. Полозова

г. Хмельницкий, Хмельницкий государственный университет

Проведение занятий по некоторым курсам информатики и компьютерной техники связано с обучением студентов решению практических задач. В настоящее время наиболее популярными задачами является организация доступа к сети Интернет локальных сетей организаций, фирм. При их решении часто необходимо выполнять следующие действия.

1. Приобретение у ближайшего провайдера Интернет сетки реальных IP-адресов для каждого компьютера компании. Чем больше компьютеров в локальной сети, тем больше адресов необходимо купить. Так, в настоящее время стоимость одного IP-адреса составляет ~0.6 у.е. в месяц. При установке в локальной сети около 500 компьютеров придется платить провайдеру ~300 у.е. в мес.

2. Администрирование сети, т.е. предоставление каждому локальному компьютеру заданных полномочий доступа к сети. Например, ограничение доступа по времени работы с Интернет, ограничение скорости доступа, предоставление возможности работы с конкретными сайтами и запрещение доступа к другим серверам и т.д.

3. Проведение мероприятий по экономии средств по оплате дорогостоящего канала Интернет. Например, установка сервера, кеширующего предыдущие запросы с тем, чтобы при повторном обращении к внешним серверам информация частично считывалась с кеша (жесткого диска) с накопленной за заданный промежуток времени информацией. Чем больше будет идти обращений к ранее накопленной информации, тем меньше будет загружен внешний канал.

Решение первых двух задач возможно с использованием прокси-сервера squid. При решении первой задачи необходимо, чтобы интерфейс прокси сервера имел один реальный ip-адрес а второй локальный, например из сетки класса В – 172.20.0.0/255.255.0.0. Тогда все компьютеры с локальными ip-

адресами организации должны быть прописаны в конфигурационном файле squid.conf, что даст возможность прокси-серверу удовлетворять их запросы к внешним серверам, подменяя их локальные адреса своим реальным ip-адресом. Ниже приведен фрагмент squid.conf для шести локальных компьютеров.

```
acl net172_011 src 172.20.0.11/255.255.255.255
acl net172_012 src 172.20.0.12/255.255.255.255
acl net172_013 src 172.20.0.13/255.255.255.255
acl net172_014 src 172.20.0.14/255.255.255.255
acl net172_015 src 172.20.0.15/255.255.255.255
acl net172_016 src 172.20.0.16/255.255.255.255
#
http_access allow net172_011
http_access allow net172_012
http_access allow net172_013
http_access allow net172_014
http_access allow net172_015
http_access allow net172_016
```

Таким образом, использование squid для сетки в ~500 компьютеров приводит к экономии ~300 у.е. в месяц. Для обеспечения работа сети из 500 компьютеров достаточно одного сервера с процессором Pentium IV частотой 2Гц, оперативной памятью ~512 Mbyte и дисковой ~10 Gbyte стоимостью не более 600 у.е. Следовательно, такой сервер для организации окупится за 2 месяца.

Решение второй задачи также возможно путем внесения изменений в файл squid.conf. Разрешение доступа для конкретных ip-адресов к прокси рассмотрено выше. Ограничение скорости до 4000байт/с из общего потока 24000байт/с для приведенных выше адресов может быть выполнено добавлением следующих строк в squid.conf:

```
delay_pools 1
delay_class 1 2
delay_access 1 allow net172_011
delay_access 1 allow net172_012
delay_access 1 allow net172_013
delay_access 1 allow net172_014
delay_access 1 allow net172_015
```

```
delay_access 1 allow net172_016
delay_access 1 deny all
delay_parameters 1 24000/24000 4000/4000
```

Для организации авторизованного доступа с каждого локального компьютера к прокси-серверу squid, а, следовательно, и к интернет – добавлением таких строк:

```
authenticate_program /usr/local/squid/bin/ncsa_auth
/usr/local/squid/etc/passwd
authenticate_children 5
#
acl net172_011_p proxy_auth h011
acl net172_012_p proxy_auth h012
acl net172_013_p proxy_auth h013
acl net172_014_p proxy_auth h014
acl net172_015_p proxy_auth h015
acl net172_016_p proxy_auth h016
#
http_access allow net172_011 net172_011_p
http_access allow net172_012 net172_012_p
http_access allow net172_013 net172_013_p
http_access allow net172_014 net172_014_p
http_access allow net172_015 net172_015_p
http_access allow net172_016 net172_016_p
```

Здесь /usr/local/squid/bin/ncsa_auth – программа аутентификации, поставляемая с squid, а /usr/local/squid/etc/passwd – файл паролей, который может создаваться программой htpasswd. Она поставляется вместе с web-сервером apache.

Для решения третьей задачи, т.е. проверки действительной эффективности работы squid-а с внешним каналом вначале сделаем следующий эксперимент. Отключим возможность squid-а сбрасывать полученную с сайтов информацию в кеш на магнитном диске. Для этого в конфигурационный файл добавим строку:

```
no_cache deny all
```

А затем включим кеш на диске размером 150 Mbyte. Далее с помощью программы MRTG прорисуем графики, сколько информации было передано squid-м компьютерам локальной сети, сколько информации squid-м было получено от внешних серверов, сколько информации прошло через внешний канал, под-

ключенный только к этому прокси-серверу . Дополнительно с помощью MRTG прорисуем график кривых среднего количества запросов в минуту за каждые пять минут, которое было извлечено из кеша для локальных клиентов и среднего числа запросов, которые были возвращены клиентам от прокси. Очевидно, что взяв отношение этих запросов, получим эффективность работы прокси squid. Так, если отношение равно в среднем 30%, например, за сутки, то внешний канал должен быть загружен на 30% меньше, чем если бы не было прокси и выход в Интернет был организован с компьютеров организации с реальными ip-адресами без использования прокси. В качестве экспериментального канала использовался канал на Lucky Link EU со скоростью 256 Kbit/c. Прокси squid версии 2.4 под Unix FreeBSD версии 4.5.

Для снятия данных с прокси использовался протокол SNMP. Для этого squid собирался с ключом `–enable-snmp`, а в конфигурационном файле `squid.conf` были добавлены строки:

```
acl      snmppublic      snmp_community      public
snmp_access allow all
```

Для проверки работоспособности `snmp squid` воспользуемся утилитой `snmpwalk`. Объекты squid хранятся в MIB файле `/usr/local/squid/etc/mib.txt`: здесь надо убрать фигурные скобки со всем содержимым из описания модуля и положить его в `/usr/local/share/snmp/mibs`. Для извлечение полного дерева объектов для squid необходимо выполнить команду:

```
snmpwalk -p 3401 -m SQUID-MIB имя-хоста имя-сообщества squid
```

Конкретно для узла 195.230.134.97 и сообщества `public` команда примет вид:

```
snmpwalk -m SQUID-MIB -p 3401 195.230.134.97 public squid
```

Далее будет следовать распечатка всех объектов, лежащих в поддереве `iso.org.dod.internet.private.enterprises.nlanr.squid (1.3.6.1.4.1.3495.1)`. Наиболее важные для нас следующие:

```
cacheSystem
```

```
cacheSysVMsize (объем кеша в оперативной памяти, в KB)
```

```
cacheSysStorage (объем кеша на диске, в KB)
```

```
cacheUptime (время работы прокси после последней перезагрузки в 1/100 секунды)
```

cachePerf

cacheProtoStats

cacheProtoAggregateStats

cacheProtoClientHttpRequests (Полное число HTTP запросов, возвращенное от кеша (squid-a) клиентам)

cacheHttpHits (Полное число HTTP запросов, которые попали в кеш, который создан squid-ом)

cacheHttpInKb (число килобайт информации, полученное от клиентов)

cacheHttpOutKb (число килобайт информации, отданное клиентам)

cacheServerInKb (число килобайт информации, полученное squid от серверов)

cacheServerOutKb (число килобайт информации, отданное squid серверам)

cacheClients (число IP адресов клиентов, которые обратились к кешу)

cacheMedianSvcTable (усредненная статистика за интервал времени, указанный в cacheMedianTime)

cacheMedianSvcEntry (индекс доступа к экземпляру: cacheMedianTime)

cacheHttpAllSvcTime.5 (среднее время обслуживания всех запросов, миллисекунд)

cacheHttpMissSvcTime.5 (среднее время обслуживание запросов, отсутствующих в кеш или устаревших, миллисекунд)

Для формирования графиков с помощью MRTG составляем конфигурационный файл mrtg.cfg [1]:

```
WorkDir: /usr/local/apache/htdocs/mrtg6
```

```
LoadMIBs:/usr/local/mrtg-2/cfg6/mib.txt,
```

```
/usr/local/share/snmp/mibs/IF-MIB.txt
```

```
RunAsDaemon: Yes
```

```
Interval: 5
```

```
Language: russian
```

```
kilo[_]:1024
```

```
#
```

```
# Http Hits/Requests
```

```

#
Target[cacheHits]:
cheHttpHits&cacheProtoClientHttpRequests:public@172.20.0.15 ca-
Title[cacheHits]: HTTP Hits
PageTop[cacheHits]: <h1>HTTP Hits / Requests</h1>
MaxBytes[cacheHits]: 400
Supress[cacheHits]: y
YLegend[cacheHits]: perminute
ShortLegend[cacheHits]: req/min
LegendI[cacheHits]: HTTP hits
LegendO[cacheHits]: HTTP requests
Legend1[cacheHits]: hits
Legend2[cacheHits]: requests
Options[cacheHits]: nopercents, perminute, dorelpercent
#
# Сколько байт получено по протоколу HTTP клиентами от
# squid-a
#
Target[OutKb]:
cheHttpOutKb&cacheHttpOutKb:public@172.20.0.153:3401 * 1024 ca-
MaxBytes[OutKb]: 4000000000
Title[OutKb]: HTTP Out Traffic
Options[OutKb]: nopercents,gauge
PageTop[OutKb]: <h1>HTTP Out Traffic </h1>
YLegend[OutKb]: Bytes
ShortLegend[OutKb]: Bytes
LegendI[OutKb]: HTTP Out
LegendO[OutKb]: HTTP Out
Legend1[OutKb]: Out
Legend2[OutKb]: Out
#
# Сколько байт получил squid от внешних серверов
#
Target[InKb]:
InKb&cacheServerInKb:public@172.20.0.153:3401 * 1024 cacheServer-
MaxBytes[InKb]: 4000000000
Title[InKb]: Server In Traffic
Options[InKb]: nopercents,gauge

```

```
PageTop[InKb]: <h1>Server In Traffic </h1>
YLegend[InKb]: Bytes
ShortLegend[InKb]: Bytes
LegendI[InKb]: Server In
LegendO[InKb]: Server In
Legend1[InKb]: In
Legend2[InKb]: In
#
# Сколько байт было переслано squid-у с внешнего
# спутникового канала
#
Target[16]: ifInOctets.16&ifInOctets.16:public@172.20.0.160:
MaxBytes[16]: 4000000000
Title[16]: Traffic Analysis for switch libr.
PageTop[16]: <H1>Lucky Link</H1>
Options[16]: nopercents,gauge
YLegend[16]: Bytes
ShortLegend[16]: Bytes
LegendI[16]: HTTP In
LegendO[16]: HTTP In
Legend1[16]: Out
Legend2[16]: Out
```

Результат работы представлен следующими графиками в последовательности конфигурационного файла.

Правая часть графиков относится к случаю выключенного кеша на диске, левая – с организацией кеша на диске в размере 150 Мбайт. В случае отключенного кеша отдано squid-м локальным компьютерам 421 Мбайт (рис. 2), принято squid-м от внешних серверов 414 Мбайт (рис. 3), переслано squid-у со спутника 476 Мбайт (рис. 4). Таким образом, squid с внешнего спутникового канала взял ~ на 13% больше информации, чем отдал клиентам локальной сети. Следовательно squid без дискового кеша работает крайне неэффективно. Для включенного кеша отдано squid-м локальным компьютерам 533 Мбайт (рис. 2), принято squid-м от внешних серверов 466 Мбайт (рис. 3), переслано squid-у со спутника 528 Мбайт (рис. 4). Таким образом, с созданием небольшого дискового кеша эффективность работы squid увеличивается и примерно соответствует количеству полученной

информации с внешнего канала.

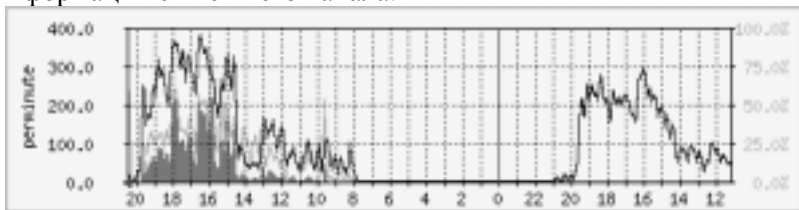


Рис. 1.

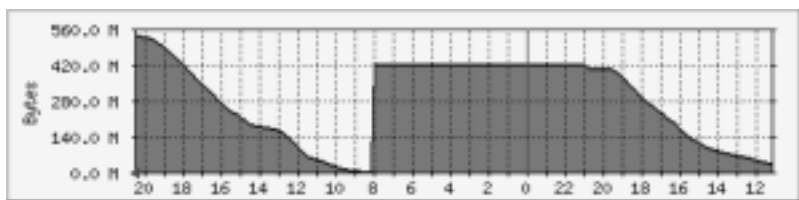


Рис. 2.

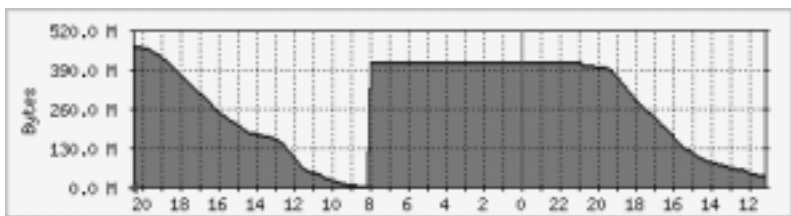


Рис. 3.

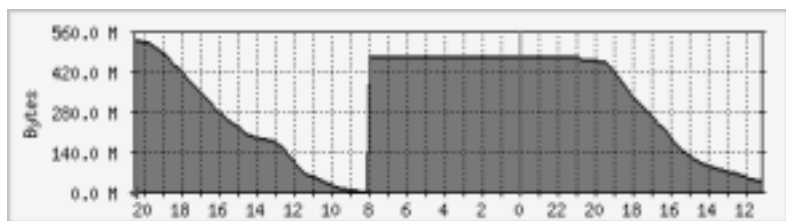


Рис. 4.

Увеличим размер кеша до 1500 Мбайт и проанализируем эффективность squid-а вновь. За несколько дней работы отдано squid-ом локальной сети 2005 Мбайт (рис. 6), принято squid-м от внешних серверов 1619 Мбайт (рис. 7), переслано squid-у со спутника 1757 Мбайт (рис. 8). Таким образом, реальная эффек-

тивность работы прокси-сервера для большого кэша со временем его обновления 7 дней составила примерно $((2005-1757)/1757) \times 100\% = 14\%$ (рис. 6, 8). Хотя среднее попадание в кэш находится в пределах 30-35% (см. рис. 5), то очевидно, что и теоретическая эффективность работы squid-а должна быть около 30-35%. Однако, если принять во внимание данные из рис. 6 и рис. 7, то из них следует что эффективность работы прокси примерно равна $((2005-1619)/1619) \times 100\% = 24\%$. Такое различие (14% и 24%) связано, очевидно, с тем, что squid посылает внешним серверам пакеты запросов и принимает пакеты ответов размером примерно на 10% больше, чем они формируются локальными компьютерами. В этих 10% прокси формирует необходимую ему служебную информацию, в частности, ip-адрес локальных компьютеров, которым необходимо вернуть запрошенные ими данные с внешних серверов.

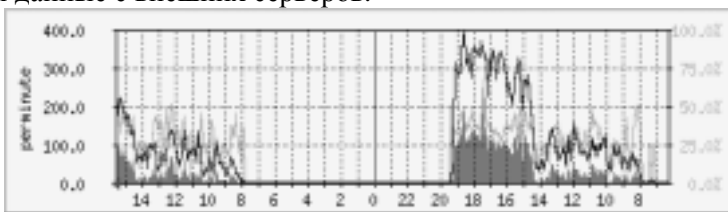


Рис. 5.

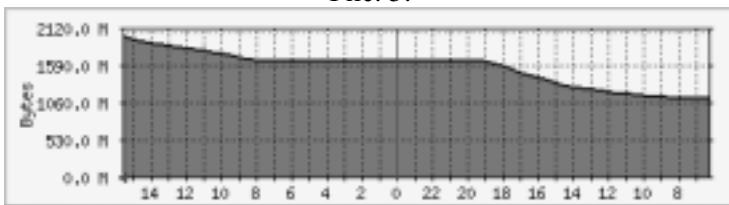


Рис. 6.

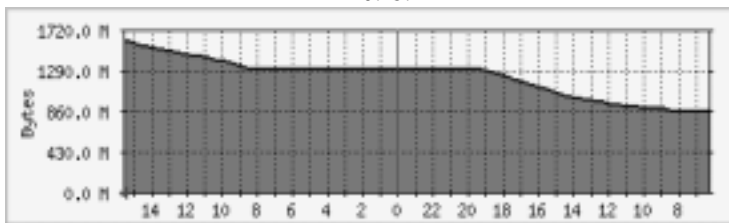


Рис. 7.

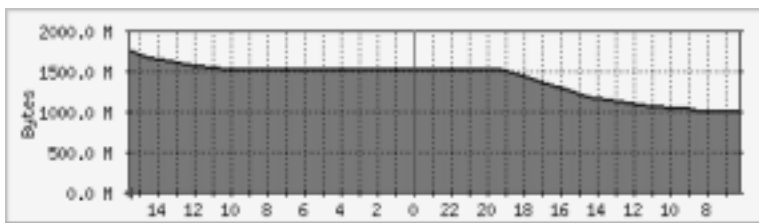


Рис. 8.

Литература

1. Мясичев А.А. Полозова В.М. Использование MRTG для учета загрузки интерфейсов. // Збірник наукових праць. – №24, ч. II. – Хмельницький: Видавництво НАПВУ, 2003. – С. 173-177.

ОЗНАКОМЛЕНИЕ С КОМАНДАМИ POP3-СЕРВЕРА ПРИ ИЗУЧЕНИИ ДИСЦИПЛИНЫ «КОМПЬЮТЕРНЫЕ СЕТИ» ДЛЯ ПОЛУЧЕНИЯ E-MAIL

А.А. Мясищев, В.М. Полозова

г. Хмельницкий, Хмельницкий государственный университет

С распространением сети Интернет в учебном процессе все больше внимания уделяется изучению ресурсов Интернет. В настоящее время важнейшим ресурсом является электронная почта. Основную роль в системе электронной почты играют программы трех типов – это транспортные агенты, агенты доставки и пользовательские агенты.

Пользовательский агент является оболочкой пользователя для работы с электронной почтой и выполняет функции:

- получение сообщений с почтового сервера;
- презентация, хранение, удаление и каталогизирование почтовых сообщений;
- создание нового сообщения и передача его транспортному агенту для дальнейшей обработки и доставки.

Рассмотрим использование POP (Post Office Protocol) протокола, POP-сервера для формирования пользовательского агента. Предполагается, что агент формируется на сервере под Unix, на нем запущены Web-сервер Apache с интерпретатором php, а просмотр почты осуществляется с любого браузера. Созданный агент может принимать только одно почтовое сообщение за одно обращение к POP-серверу и распознать один рисунок, закодированный uuencode (для простоты).

Протокол POP используется для пересылки новой почты пользователя с сервера на рабочую станцию пользователя. В настоящий момент используется версия 3; по набору команд она не совместима с предыдущими версиями. Номер порта сервера POP – TCP/110. После установления соединения с клиентом сервер ожидает ввода команд и данных в текстовом виде. Строчные и прописные буквы в командах не различаются. Реакция сервера – строка, начинающаяся с метки «+OK» или «-ERR», за которой следует текстовый комментарий и, если команда это подразумевает, с новой строки выводятся данные

(текст сообщения или листинг сообщений). Вывод данных заканчивается строкой, содержащей только символ «.» («точка»). Если среди данных есть такая строка, то точка в этой строке удаляется.

Некоторые команды POP3:

USER *имя_пользователя*

– первая команда сеанса, вводится имя пользователя (идентификатор почтового ящика).

PASS *пароль*

– вторая команда сеанса, вводится пароль.

STAT

– после метки «+OK» выводит два числа: число сообщений и их общий объем в байтах.

LIST *n*

– если *n* указано, то после метки «+OK» выводит размер сообщения номер *n* в байтах. Иначе выводит список из двух колонок: номер сообщения, пробел, размер сообщения в байтах; вывод списка заканчивается строкой, содержащей только символ «.» («точка»).

RETR *n*

– выводит сообщение номер *n*. Вывод заканчивается строкой, содержащей только символ «.» («точка»).

DELE *n*

– удаляет сообщение номер *n* с сервера; при этом нумерация сообщений не изменяется, а все удаленные в данном сеансе сообщения могут быть восстановлены командой RSET.

QUIT

– конец связи.

Ниже представлена учебная программа, которая выполняет получение почтового сообщения с перечисленными выше допущениями. Пояснения работы программы представлены комментариями.

```
<html>
<head>
<title>Lab1b</title>
</head>
<?php
print "<pre>";
// соединение с POP3
```

```

$fp=fsockopen("cel.tup",110,&$errno,&$errdesc,5)
;
if ( ! $fp ) { echo "$errno"; echo "$errdesc";
}
echo fgets($fp,100);
fputs ($fp,"user alex\n");
echo fgets($fp,100);
fputs ($fp,"pass 123456\n");
echo fgets($fp,100);
// получение 1-го письма
fputs ($fp,"retr 1\n");
$s="0";
$fpl=fopen("let1",'w');
while ( !(strpos($s, ".") === 0) )
{
$s=fgets($fp,80);
fputs($fpl,$s);
}
fclose($fpl);
fputs ($fp,"dele 1\n");
fputs ($fp,"quit\n");
fclose($fp);
// чтение письма
$fprl=fopen("let1",'r');
$pics="0";
$i=0;
while (!feof($fprl)) {
$letl[$i]=fgets($fprl,100);
if (strstr($letl[$i],"begin") ) {$b=explode("
",$letl[$i]); $pics=$b[2];}
$i += 1;}
$ni=$i-1;
fclose($fprl);
if ( $pics != "0" ) `uudecode let1`;
// формирование заголовка
for($i=0;$i<=$ni;$i++)
{ if( strpos($letl[$i],"From:") === 0 )
$frl=$letl[$i];
if( strpos($letl[$i],"To:") === 0 )
$tol=$letl[$i];
if( strlen($letl[$i]) == 2 ) {$n=$i; break;}
}
$k=0;

```

```

if ( $pics != "0" ) {
while ( !strstr($let1[$k],"begin" ) )
{ $let1_1[$k]=$let1[$k];
$k=$k+1;}
}
if ( $pics == "0" ) $let1_1=$let1;
// вывод заголовка, письма и рисунка
print "<br>letter N1<br><br>";
echo $frl;
echo $tol;
for ( $j=$n;$j<=$ni-2;$j++)
echo $let1_1[$j];
print "<CENTER><IMG
SRC='http://cel.tup/~alex/php/$pics'></CENTER>"
?>
</body>
</html>

```

Эта программа может быть просто расширена для получения n -го количества писем, включающих в себя m -е количество рисунков.

СТВОРЕННЯ ТА ОПУБЛІКУВАННЯ НАВЧАЛЬНИХ ПРОГРАМНИХ ПРОДУКТІВ

Я.Л. Напреев

м. Умань, Уманський державний педагогічний університет
імені Павла Тичини

За умов розвитку дистанційної освіти, актуальним постає питання розробки навчальних програмних засобів та відкриття доступу до них. Комп'ютеризація галузі науки та освіти призводить до того, що деякі викладачі (вчителі) намагаються створювати власні розробки навчальних комп'ютерних програм, курсів лекцій, лабораторних занять в електронній формі. Слід зазначити, що не кожен з них володіє мовами програмування і може створити за їх допомогою проекти, які могли б відповідати сучасним умовам освіти. Крім того, далеко не кожен навчальний заклад може забезпечити своїх викладачів таким ліцензованим програмним забезпеченням, як середовища програмування. Частіше за все у навчального закладу є комп'ютерний клас або лабораторія, на комп'ютерах якої є тільки “стандартні” програми – додатки операційної системи Microsoft Windows, та пакунок програм Microsoft Office. Більшість викладачів вміє працювати з такими програмами, хоча б на рівні створення документів Microsoft Word (створення документу, набирання, редагування, форматування тексту тощо, робота з графічними об'єктами та таблицями). Навіть цих знань достатньо для розробки навчальних програм у електронному варіанті. Розглянемо основні компоненти пакунку Microsoft Office.

Насамперед, це “флагман”, одна з найбільш популярних (відповідно і розповсюджених) програм – **текстовий редактор Microsoft Word**. В ньому можна проводити різноманітні операції з текстом, таблицями, графікою. Крім того в документ можна додавати звукові та відео файли, об'єкти інших програм, проінстальованих на комп'ютері. Власне, Microsoft Word можна розглядати як засіб створення текстів, пояснень, різноманітної документації.

Наступний компонент – **редактор електронних таблиць Microsoft Excel**. Електронні таблиці, на відміну від звичайних

(властивих тому ж Microsoft Word) дозволяють автоматизувати обчислення, чим, в свою чергу, можна скористатись при створенні тестів (або інших контролюючих засобів).

Microsoft Access – система управління базами даних. Використання баз даних у власному проєкті – складне питання. Для створення професійної бази даних знань оперування з текстом замало, відповідно не кожен зможе її створити. Але проста база даних може бути використана для розробки різноманітних довідників, словників тощо.

Редактор презентацій Microsoft PowerPoint – найбільш вдала програм з точки зору створення різноманітних демонстраційних засобів, в яких використовуються ефекти анімації, звуки, відео. Створення презентації у Microsoft PowerPoint схоже з створенням документу Microsoft Word, і не потребує певних спеціальних знань. Простий, зручний інтерфейс цієї програми, засоби роботи з графікою (що властиво також Microsoft Word і Microsoft Excel) дозволяє без наявності специфічних знань (наприклад, дизайнерських) робити за допомогою шаблонів яскраві презентації, які можна використовувати як навчальні демонстраційні програми, електронні підручники, енциклопедії тощо.

З допоміжних засобів Microsoft Office слід виділити наступні. Графічний редактор **Microsoft Photo Editor**, який (у поєднанні з стандартним для операційної системи Microsoft Windows редактором **Paint**) дозволяє створювати та редагувати графічні файли різних форматів, сканувати зображення (за наявності сканера). Редактор **Microsoft Equation** дозволяє набирати та редагувати формули, що містять символи, недоступні на клавіатурі. Колекція **Microsoft Clipart** містить малюнки, відео- та звукові файли, які можна застосовувати у текстах, електронних таблицях та презентаціях.

Окремо слід розглянути створення та запис макрокоманд (так званих *макросів*). Макрокоманди, записані у програмах Microsoft Office призначені для спрощення та автоматизування роботи під час створення документа (таблиці, презентації). Дуже рідко їх розглядають як засоби програмування. Фактично макроси – це записані команди мови програмування Microsoft Visual Basic. Microsoft Word, Excel та PowerPoint містять вбудовані редактори Microsoft Visual Basic, в яких можна переглянути, відре-

дагувати, створити нові макрокоманди. Але тут необхідно мати певні знання цієї мови програмування.

Основні програми пакунку Microsoft Office дозволяють використовувати при створенні документів гіпертекстові посилання, які надають можливість швидко та зручно відкривати інші документи, переходити до інших частин цього ж документу. За допомогою цих гіперпосилань можна об'єднувати документи різних типів. Прикладом цього може стати текстовий документ (теоретичний матеріал певного курсу), при ознайомленні з яким можна переглянути певні демонстрації (створені у вигляді презентації або відеофайлу), виконати тестові (контролюючі) завдання (створені у вигляді електронної таблиці), прослухати пояснення (записані у звуковий файл автором); під час виконання контрольних завдань учню допоможе електронний довідник або словник (створений у середовищі електронних таблиць, баз даних). У комплект програм пакунку Microsoft Office може входити і редактор web-сторінок (документів HTML) – **Microsoft FrontPage**, якій (за умов знання принципів створення документів HTML) дозволяє створити електронний документ на більш професійному рівні.

Крім того, програми з Microsoft Office і самі дозволяють зберігати документи у вигляді електронних сторінок, які можна об'єднувати за допомогою гіпертекстових посилань. Створені таким чином електронні документи (сайти) можна розміщувати як на безкоштовних web-серверах, так і на комерційних (наприклад власного провайдера) за певну плату. Після цього студент (учень) або особа, яка навчається буде мати доступ до навчальних матеріалів за допомогою комп'ютерної мережі Internet з будь-якого місця планети.

Не слід забувати і стандартні мультимедійні засоби операційної системи Microsoft Windows. Програми пакунку Microsoft Office відносно відео та звукових файлів не є креативними. Вони використовують мультимедійні файли, створені в *інших* програмах. В стандартних додатках операційних систем Microsoft Windows є також і стандартні засоби створення (запису) та редагування звукових файлів, їх відтворення.

Відтворення звукових файлів, відеофайлів, аудіо дисків доступне за допомогою стандартного програвача – **Windows Media**

Player. Крім того, в ця ж програма може оцифрувати записи з аудіо CD у звукові файли. Запис та редагування звукових файлів – стандартна програма звукозапису.

Для відеомонтажу та відео захвату можна скористатись програмою, яка з 2000 року входить до стандартних програм-додатків Microsoft Windows – **Microsoft Windows Movie Maker.**

Зрозуміло, що ці програми дуже спрощені (відносно їх професійних аналогів), але є достатніми для простих операцій з мультимедійною інформацією.

Ще одним питанням опублікування власних програмних продуктів є запис і розповсюдження програм на лазерних компакт-дисках. Функція запису на диски CD, DVD присутня, як звичайна функція копіювання та переміщення файлів у операційній системі Microsoft Windows XP. Для створення власного звукового диску, можна скористатись програвачем Windows Media Player.

Таким чином, можна зробити висновки, що за наявності звичайного програмного забезпечення (операційної системи Microsoft Windows, та пакунку програм Microsoft Office) можна створювати власні навчальні програмні засоби, переводити їх у вигляд web-сторінок, опубліковувати їх у комп'ютерній мережі Internet, записувати лазерні комплект диски. Вищезазначені операції є актуальними для розробки авторського програмного забезпечення в умовах розвитку дистанційної освіти.

ФОРМУВАННЯ ПЕДАГОГІЧНОЇ СПРЯМОВАНОСТІ НАВИЧОК ПРОГРАМУВАННЯ МАЙБУТНІХ УЧИТЕЛІВ ІНФОРМАТИКИ

С.М. Овчаров

м. Полтава, Полтавський державний педагогічний університет
ім. В.Г. Короленка
allmail@pdpu.septor.net.ua

Аналіз сучасних психолого-педагогічних досліджень свідчить про зростання інтересу до питань ефективної професійної підготовки фахівців з використанням комп'ютерних навчальних засобів, що являють собою програмні засоби навчального призначення, які використовуються під час навчально-виховного процесу у вищій школі. Вони дозволяють індивідуалізувати та диференціювати процес навчання студентів, здійснювати контроль з діагностикою помилок, а також самоконтроль і самокорекцію їхньої навчальної діяльності, моделювати та імітувати процеси та явища, що вивчаються, підсилити мотивацію навчання. Тому використання комп'ютерних навчальних засобів у сучасній освіті є необхідною умовою її якості.

Особливої уваги ця проблема набуває в наш час, коли, враховуючи сучасні тенденції розвитку інформатизації освіти, вимогою сьогодення стає необхідність підготовки фахівців з педагогічної інформатики, яка є науково-методичним напрямком в інформатиці, присвяченим вивченню процесів взаємодії освіти та інформатики, проблемам навчання інформатиці й вихованню молодого покоління за допомогою інформаційних технологій, питанням інформаційного забезпечення та автоматизації педагогічної діяльності й адміністративного керування педагогічними системами [1].

На нашу думку, підготовку фахівців з педагогічної інформатики доцільніше було б започаткувати на певних факультетах саме педагогічних університетів (фізико-математичних, природничих тощо), де викладається досить широкий спектр навчальних дисциплін: як загальнонаукових, так і психології, педагогіки, інформатики (у тому числі програмування) та деяких інших. Це дозволило б випускникам інформатичних спеціальностей педа-

гогічних ВНЗ оволодіти основами застосування існуючих і створення власних навчальних програмних засобів, допомагати вчителям-предметникам широко впроваджувати нові інформаційні технології в шкільну практику, використовувати шкільний комп'ютерний клас для проведення уроків з різних предметів на сучасному рівні. (Навчальним програмним засобом (НПЗ) називають такий, у якому відображається деяка предметна галузь, у тій чи іншій мірі реалізується технологія її вивчення, забезпечуються умови для здійснення різних видів навчальної діяльності).

Ця проблема набуває особливої значущості в наш час тому, що бурхливий розвиток інформаційних технологій значно випереджає розробку відповідного методичного та програмного забезпечення навчального призначення. До того ж, існуючі навчальні програмні засоби не завжди повністю відповідають навчальним планам сучасних закладів освіти різних рівнів. Як наслідок, виникла нагальна необхідність готувати учителів інформатики на такому рівні, який забезпечував би їм можливість широко використовувати навчальні програмні засоби під час майбутньої педагогічної діяльності, а за необхідності – і створювати власні НПЗ, тобто набувати певних навичок з педагогічної інформатики.

Виходячи з вищезазначеного, на фізико-математичному факультеті ПДПУ ім. В.Г. Короленка нами розроблений і впроваджений в практику навчання спецкурс з інформатики “Навчальні програмні засоби: методика їх створення та застосування”. Даний спецкурс викладається студентам спеціальностей, де інформатика є однією з основних навчальних дисциплін. Метою розробленого спецкурсу є ознайомлення студентів з існуючими навчальними програмними засобами та методикою їх застосування під час проведення занять з різних предметів, а також оволодіння ними основами створення власних НПЗ засобами сучасних систем програмування. Він спирається на знання, отримані студентами під час вивчення таких курсів з інформатики, як “Програмне забезпечення ЕОМ”, “Програмування та інформатика” та “Сучасні системи програмування”, вивчення якого дозволяє студентам оволодіти основами об'єктно-орієнтованого програмування. Даний курс також був розроблений нами й виклада-

ється на фізико-математичному факультеті ПДПУ ім. В.Г. Короленка вже три роки [2].

Розроблений нами спецкурс “Навчальні програмні засоби: методика їх створення та застосування” складається з двох розділів. І розділ має назву “Загальна характеристика та принципи застосування комп’ютерних програм навчального призначення” і передбачає розгляд таких питань: психолого-педагогічні аспекти використання комп’ютерів у навчальному процесі, класифікація комп’ютерних програм навчального призначення та специфіка їх використання, застосування навчальних програмних засобів в умовах дистанційної освіти та деяких інших.

ІІ розділ називається “Методика розробки й використання навчальних програмних засобів”. Він присвячений вивченню таких питань: загальні принципи побудови сценаріїв тренажерів та навчальних пакетів прикладних програм, методика розробки автоматизованих дидактичних засобів, особливості їх застосування та деяких інших.

Лабораторний практикум під час вивчення даного спецкурсу проводиться в два етапи. На першому з них студенти опановують методику використання існуючих навчальних програмних засобів під час проведення занять з різних навчальних дисциплін. На другому етапі виконання лабораторних робіт організується на основі використання методу проектів. Студенти, за узгодженням з викладачем, виходячи з рівня особистої підготовки і власних уподобань, обирають тематику для розробки власного навчального програмного засобу, а також інструментальне середовище, яке вони планують використовувати для створення свого проекту. Найбільш підготовлені з основ програмування студенти, як правило, обирають роботу з використанням сучасних систем програмування: Borland Delphi, Visual Basic, Java тощо. Менш підготовлені – будь-які інші середовища, наприклад: Turbo Pascal, PowerPoint, DemoShield та інші. Вибір здійснюється з урахуванням індивідуальних здібностей та нахилів студентів. В якості навчальних програмних засобів студенти створюють слайд-лекції, навчаючі програми, електронні розв’язники задач, комплекси навчально-контролюючих або тестових програм з різних навчальних дисциплін. Тематика для їх створення обирається такою, що відповідає шкільній програмі з предметів: алге-

бра та початки аналізу, геометрія, фізика, інформатика тощо.

Після вивчення спецкурсу проводиться захист проектів, для чого готується відповідна документація (аналогічно до захисту курсових робіт). Захист проводиться методом демонстрації розробленого проекту та його обговорення навчальною групою. Загальна оцінка виставляється з урахуванням зауважень і думок не тільки викладача, але й усієї навчальної групи. Це забезпечує об'єктивність та відкритість оцінювання знань, що відповідає вимогам демократизації освіти на сучасному етапі.

Деякі з найкращих проектів – навчальних програмних засобів, створених студентами нашого університету, впроваджені до практики навчально-виховного процесу нашого університету та інших навчальних закладів міста Полтави й області, що підтверджує ефективність запропонованої методики.

Таким чином, враховуючи сучасні тенденції розвитку інформатизації освіти, вимогою сьогодення є необхідність оволодіння майбутніми вчителями інформатики основами створення власних навчальних програмних засобів, а також набуття вмінь та навичок використання існуючих НПЗ при проведенні занять за допомогою комп'ютерної техніки на сучасному рівні.

Література

1. Ильина Т.Ю., Румянцев И.А. Педагогическая информатика и информационная педагогика // Педагогическая информатика. – 2002. – №3. – С.48 – 51.
2. Овчаров С.М. Сучасні тенденції викладання основ програмування студентам фізико-математичних факультетів педагогічних ВНЗ // Імідж сучасного педагога. – 2002. – №10(29). – С. 26–27.

ПІДРУЧНИК ЯК ОДИН ІЗ ОСНОВНИХ ДИДАКТИЧНИХ ЗАСОБІВ РОЗВИТКУ ТВОРЧОГО ПОТЕНЦІАЛУ СТУДЕНТІВ ПІД ЧАС НАВЧАННЯ ІНФОРМАТИКИ

О.Г. Онишко

м. Хмельницький, Хмельницький державний університет
dzyoban@ief.km.ua

Підручник займає особливе, провідне місце серед засобів навчання і є одним із важливих дидактичних засобів розвитку творчого потенціалу студентів під час навчання інформатики.

У порівнянні з іншими дидактичними засобами, у ньому найбільш повно передається зміст навчання, його матеріал відповідає навчальній програмі і є обов'язковим для засвоєння. У підручнику формулюються цілі вивчення навчального матеріалу, даються пояснення, коментарі, алгоритми розв'язування типових задач, завдання для самостійної роботи, зразки правильних та раціональних розв'язків, відповіді та інші компоненти процесу навчання. Такий підхід до визначення структури підручника сприяє активізації цілеспрямованої роботи студентів з метою досягнення запланованих результатів навчання та розвитку творчого потенціалу студентів під час навчання інформатики.

Виходячи з того, що підручник створюється як елемент системи педагогічного управління навчанням студентів, його можна представити у вигляді двох складових: того, що повинно бути засвоєним, і того, що забезпечує процес засвоєння [1]. У підручнику відображається об'єктивно-предметний зміст (освіта, виробництво, наука), послідовність засвоєння навчального матеріалу (розуміння, оволодіння, опрацювання, контроль), самостійність застосування (вправ, самостійність застосування в стандартних умовах, самостійність застосування в нових умовах). Форми подання навчального матеріалу в підручнику є тим кодом, за яким сховані засоби управління пізнавальною діяльністю студентів, у ньому цілеспрямовано закладені певні дидактичні схеми. Наприклад, таблиці в тексті можуть мати лише інформаційний зміст, а можуть виступати і засобом розвитку їх логічного мислення, підпорядковуватись цілям адаптації, схематизації знань студентів або засобом розвитку їх логічного мислення, підпорядковува-

тись цілям адаптації, схематизації змісту. Різні типи завдань і вправ (як самостійна форма подання навчального матеріалу) є тими засобами, за допомогою яких здійснюється управління пізнавальною діяльністю студентів, формування в них пізнавальної самосвідомості. З метою розвитку творчого потенціалу студентів завдання спонукають їх до аналізу власних розв'язків за допомогою уточнюючих, проблемних запитань. Крім того, завдання характеризуються наявністю попередньої мотивації, багатоваріантністю вихідних даних та шляхів обробки.

Зокрема, цій меті підпорядкована наочність підручника. Відмітимо основні функції ілюстрацій: пізнавальна, інтерпретаційна, естетична, управління пізнавальною діяльністю.

Так, розділ курсу інформатики “Вступ до комп'ютерної графіки” доцільно супроводжувати контрольними запитаннями-тестами, питаннями для повторення і прикладами підсумкових творчих робіт з фаху. З метою самоконтролю рівня засвоєння навчального матеріалу студентам пропонується виконати лабораторні роботи, які можуть мати таку структуру: теоретичні відомості; завдання та порядок виконання роботи; обладнання й програмне забезпечення; методичні вказівки щодо виконання завдання тощо.

Засобами, які забезпечують управління пізнавальною діяльністю студентів, здійснюють організацію засвоєння навчального матеріалу, контроль за процесом навчання є:

- розподіл тексту на головний і додатковий,
- виділення суттєвого,
- підкреслювання,
- наявність розгорнутих висновків, резюме,
- наявність блоків задач, які дозволяють проводити адаптивне навчання.

Важливими засобами є також послідовність подання матеріалу, елементи евристичних програм, дослідницьких методів, використання інших джерел інформації, тлумачення незнайомих слів, форми самоконтролю тощо. Все це спрямовано на головне в навчальній діяльності студентів і на формування творчих здібностей – розуміння ними того, що вони вивчають, і того, що відбувається з ними в процесі вивчення матеріалу та розв'язання завдань.

Загальновідомо, що у світі накопичено такий великий обсяг інформації, що повідомити її всю студентів у процесі навчання неможливо. Тому потрібно, щоб викладач умів із усієї відомої інформації вибрати необхідну, а потім, дотримуючись стратегічної цілі навчально-виховного процесу, умів розробити або вибрати методику передачі студентів конкретної інформації.

Так, вивчення курсу інформатики повинно відповідати завданням навчального предмету інформатики і забезпечити рівень викладання, близький до світового. У підручнику з інформатики повинна бути відображена синтетичність цього предмета, стрімкий розвиток її засобів і методів пізнання світу. Уміння здійснювати пошук необхідної інформації, працювати з книгами й іншими джерелами інформації, проводити спостереження і зіставлення складають основу інформаційної культури фахівця.

Кожна навчальна дисципліна складається з двох частин: перша має класичний зміст, тобто не змінюється досить значний проміжок часу, друга – потребує періодичного відновлення в зв'язку з досягненнями теорії і практики. Уміння використовувати нову інформацію – запорука успіху викладання і свідчення майстерності викладача.

В умовах інформатизації суспільства необхідно використовувати нові технології в навчанні, які дозволяють максимально індивідуалізувати навчання, значно збільшити час самостійної роботи студента, відійти від простої репродукції знань і перейти до їхнього глибокого засвоєння й осмислення.

У сучасних умовах постійного науково-технічного розвитку необхідно, щоб спеціаліст був підготовлений до самоосвіти і саморозвитку. Отже, необхідно до складу цілей навчання, які відображаються у навчальному посібнику, включати такі види діяльності, за допомогою яких можна навчити майбутнього спеціаліста вчитися. Дослідження [2] показали, що уміння учитися містить у собі такі види діяльності:

1) пошук нової інформації, що потребує, зокрема, уміння працювати в бібліотеці;

2) розуміння прочитаного, упорядкування конспекту;

3) засвоєння виділеного змісту.

Тому включаються завдання, які б відповідали умовам повнішої реалізації функціональних можливостей підручника. Зок-

рема, завдання, які відображають сутність елементарного, системовірного, внутрішньосистемного та міжсистемного видів взаємозв'язку та характеру розумової діяльності студентів. Наприклад, система завдань підручника може створюватись на основі послідовного їхнього ускладнення:

- назвати об'єкт логічної структури тексту, позатекстових компонентів, інших джерел інформації;
- виконати дії з об'єктом логічної структури навчального матеріалу, який збігається з логікою його розвитку у тексті, позатекстових компонентах, інших джерелах інформації;
- виконати дії з об'єктом логічної структури навчального матеріалу, який лише частково розгортається в логічній структурі тексту, позатекстових компонентах, інших джерелах інформації; інші типи завдань.

До системи включаються завдання на здатність візуального подання знаково-символьної інформації, включення поняття в систему зв'язків з іншими поняттями, на оволодіння операціями аналізу синтезу, узагальнення, порівняння, конкретизації тощо.

Усе більш широке визнання знаходить думка про те, що навчальний посібник (підручник) – це не джерело готових знань, які повинні бути запам'ятованими (таке відношення до нього, на жаль, нерідко зустрічається у студентів, а іноді і у викладачів), а насамперед джерело пізнавальних задач або проблем, що їх треба уміти виявити і вирішити [3].

У дослідженнях проблем розуміння тексту в психології загальноновизнаними є такі два положення. Перше: розуміння тексту (у широкому змісті) є встановлення читачем зв'язків між предметами реальної дійсності, що відбиті в цьому тексті. Друге: розуміння тексту завжди спирається на досвід читача, зокрема на знання ним значень слів, розуміння змісту речень, з яких складається текст; воно ґрунтується на застосуванні наявних у читача понять, за допомогою яких розкривається зміст тексту [3]. Для дослідження розуміння тексту виникає необхідність в аналізі його значеннєвої структури – закономірних зв'язків і систем зв'язків, що існують між елементами тексту [3].

Часто трапляється так, що студент звертає увагу, а значить і запам'ятовує, не на основні знання, що пояснюють той або інший процес або явище, а на другорядні, що є наслідком перших.

Допомогти студентів розібратися де базові знання, а де другорядні, показати, на чому варто акцентувати увагу при розгляді тих чи інших процесів, що потрібно запам'ятати в першу чергу – одна з важливих цілей сучасної дидактики. Тим більше така мета важлива зараз, коли студент одержує інформацію не тільки при вивченні предмета, а найчастіше з засобів масової інформації: телебачення, радіо, Internet.

Навчальний матеріал завжди являє собою систему, що має ту чи іншу структуру, основними елементами якої є поняття. Проте варто мати на увазі, що існують дуже різноманітні означення терміна “поняття”. Поняття являють собою сукупність визначальних ознак фактів, подій, явищ тощо. У понятті зосереджене те загальне, що об'єднує окремі елементи в один клас, узагальнює предмети деякого класу за їх специфічними ознаками. Ознаками називають когнітивні елементи, які відокремлюють поняття одне від одного.

Л.С. Виготський, вивчаючи закономірності розумового розвитку дітей, дійшов до висновку, що саме утворення понять є ключем до розуміння процесів психологічного (зокрема інтелектуального) розвитку підлітка. Це пов'язано, перш за все, з тим, що на основі понять студент починає розуміти зв'язки, відношення, взаємозалежності, які сховано поверхнею явищ, відбувається перебудова елементарних пізнавальних функцій на основі їхнього синтезу з функцією утворення понять: сприйняття фактично перетворюється у мислення образами, запам'ятовування спирається на значення зв'язки, увага набуває довільного характеру [4].

Управління навчальною діяльністю студентів за допомогою підручника, орієнтованого на розвиток творчого потенціалу студентів, здійснюється шляхом цілеспрямованої організації навчального матеріалу. Кожний структурний компонент підручника реалізує певні дидактичні функції, створює умови формування умінь і навичок розв'язання проблем. Текст підручника характеризується поєднанням інструктивного і самостійного, алгоритмічного і проблемно-дослідницьких моделей навчального процесу.

У підручнику з інформатики повинна бути відображена синтетичність цього предмету, стрімкий розвиток її засобів і методів пізнання світу. Доцільно навчальний матеріал з інформатики до-

повнювати іншим контекстом з використанням психологічних коментарів. При правильному доборі фактичного історико-культурного матеріалу розвитку майже кожної галузі науки, техніки та культури, можна до певної міри спрямувати розвиток особистості, зорієнтувати її на певну систему цінностей. Тому автор підручника повинен уважно відноситись до історії курсу інформатики, тобто включати відомості з історії курсу. Посилання на історичні факти, які свідчать про пріоритет вітчизняної науки і техніки на різних етапах їхнього розвитку, впливають на мотиваційну сферу студента, здійснюють певні освітньо-виховні функції. Такий надлишковий контекст є важливою умовою створення смислового простору в межах навчального тексту, з метою створення можливості для студента засвоювати поняття з інформатики в ширших світоглядних та міжпредметних зв'язках.

Таким чином, у підручнику з інформатики повинна бути відображена синтетичність цього предмету, стрімкий розвиток її засобів і методів пізнання світу. Кожний структурний компонент підручника реалізує певні дидактичні функції, створює умови формування умінь і навичок розв'язання проблем. Необхідним компонентом засобів організації засвоєння навчального матеріалу є система завдань.

Література

1. Гамезо М.В. Учебная книга вуза в контексте задач по перестройке образования// Новые методы и средства обучения. – 1989. – № 3(7). – С. 90-119.
2. Усачёва И.В., Ильясов И.И. Методика поиска научной литературы, чтение и составление обзора по теме исследования. – М.: МГУ, 1980. – 36 с.
3. Добраев Л.П. Смысловая структура учебного текста и проблемы его понимания. – М.: Педагогика, 1982. – 176 с.
4. Выготский Л.С. Психология. М.: ЭКСМО-Пресс, 2000. – 1008 с.
5. Онишко О.Г. Програмування мовою Pascal та використання Delphi (навчальний посібник): – Хмельницький, НВП “Еврика”, 2003. – 112 с.

ВВЕДЕНИЕ В ОТКРЫТЫЕ ОПЕРАЦИОННЫЕ СИСТЕМЫ

М.Е. Отставнов
г. Москва, TechInvestLab.com
maksim@otstavnov.com

Современная (2003 г.) редакция международного стандарта на операционные системы (ОС) [1] не только подразумевает семантику, достаточную для введения базовых понятий прикладной информатики, но и предоставляет терминологию и синтаксис, в которых их введение оказывается логически непротиворечивым и эстетически приемлемым.

Характерное для высокой культуры компьютеринга отсутствие произвольных барьеров между областями компетенции позволяет легко переходить от непосредственного исполнения команд к программированию.

Кроме того, команд оператора, определенных в стандарте, достаточно для введения одного из важнейших приложений информационных технологий, а именно, обработки текста, в качестве своего рода «демонстрационной площадки», на которой основные понятия обрастают плотью содержательных интерпретаций.

К сожалению, методической литературы на русском, ориентированной на введение в стандартные ОС непрофильных профессионалов (в том числе – педагогов), пока недостаточно. Книжки, подготовленные на рубеже 1980-х и 90-х гг., частично устарели, а частично перешли в категорию библиографических редкостей, в течение 90-х гг. их издавалось крайне мало; издания последних лет зачастую смешивают стандартную и специфичную для тех или иных реализаций части, а также материал, необходимый каждому оператору, и специфичный для администратора.

Специфика предлагаемых материалов

В этой статье приведен учебный план «Введения в открытые ОС» с почасовой разбивкой (из расчета одного академического часа лекционных занятий и одного академического часа само-

стоятельных занятий или организованного практикума на урок), ориентированный, в основном, на педагогов средней школы непрофильной (не связанной с информатикой) специализации.

Материалы к курсу были представлены на «Страницах повышения квалификации» московской газеты «Информатика» [2], а затем, с учетом высказанных педагогической аудиторией замечаний и предложений, включены в книгу [3].

Эти материалы включают подробный дискурс по тематике, приведенной в учебном плане, а также *справочник по вводимым в курсе командам*. Подготовка последнего оказалась самой неприятной частью работы, но, к сожалению, абсолютно необходимой, поскольку ни одна из вышедших за последнее десятилетие (после ОС «Демос») ОС не была снабжена полным переводом страниц руководства. В материалах нет контрольных вопросов и упражнений, что не позволяет назвать их *законченным* учебным курсом.

В настоящее время мы (без обязательств по срокам и объему) готовим их доработку для издания отдельной книгой, в связи с чем крайне заинтересованы в откликах, замечаниях и предложениях.

Скомпонованный нами материал по составу весьма близок к материалу, представленному в классическом руководстве Кернигана и Пайка [4], в части, относящейся к освоению интерфейса оператора. Исключение составляет лишь материал, касающийся визуального редактирования (редактор «vi» успел за прошедшее десятилетие стать частью международного стандарта на ОС).

Последовательность, в которой он вводится, однако, несколько отличается от принятой в курсах для программистов и специалистов в области ИТ.

Базовая понятийная структура

Информационные технологии могут быть смоделированы как определенная *вычислительная мощность* и определенная *запоминающая емкость*, распределенные на *коммуникациях* определенной пропускной способности.

Понятийная структура этого курса абстрагируется от аспектов, связанных с распределенностью вычислений и (меж)сетевой коммуникацией. Она основана на трех положениях.

1. Вычислительная мощность представлена исполнителем, способным исполнять определенные алгоритмы (точнее, вычислительные методы) в виде *процессов*, обладающих определенными *атрибутами*, порождающих друг друга и образующих иерархию наследования.
2. Процессы способны задействовать запоминающую емкость, организованную в виде *файловой структуры*, состоящей из файлов, обладающих определенными *атрибутами*.
3. Оператор взаимодействует с системой, подавая определенные *команды* оболочке или другому процессу; причем прием команд процессом является частным случаем чтения файла. Подавая команды, оператор может совершать различные действия с процессами и файлами.

Сверх приведенных трех базовых концепций практической информатики, в курсе вводится некоторое количество периферийного материала, не относящегося к их развертыванию, но прагматически важного (например, касающегося сигнализации процессов или редактирования командной строки).

Программное обеспечение практикума

Для программного обеспечения практикума достаточно любой ОС, полностью или частично следующей стандарту (любого дистрибутива «ГНУ/Линукс», «БСД», «Солярис» и т.п., а также «Дарвин»/«МакОС Х»), в базовой поставке.

При использовании ОС семейства «БСД» (включая «Дарвин»/«МакОС Х») следует установить для обучаемых оболочку по умолчанию «bash» или другую Борн-совместимую, а не «tcsh» или другую Си-совместимую.

С помощью пакетов «cygwin» (<http://www.cygwin.com>) или «Windows Services for UNIX» (<http://www.microsoft.com/windows/sfu/>) можно развернуть базу для практикума под ОС семейства «Майкрософт Уиндоуз НТ».

Урок 1. Введение

Основные понятия: операционная система, терминал, оболочка, команда, сеанс.

Осваиваемые команды: «exit», «echo».

Содержание урока:

- 1.1. Назначение операционных систем.
- 1.2. Стандарт на операционные системы (ИСО 9945). Открытые (стандартные) и альтернативные операционные системы.
- 1.3. Загрузка и разгрузка ОС.
- 1.4. Интерфейс оператора. Терминал. Физический терминал, виртуальная консоль и окно терминала в графической оконной системе.
- 1.5. Вход (регистрация) в ОС и выход из ОС. Регистрационное имя и пароль.
- 1.6. Оболочка ОС. Команда. Сеанс.
- 1.7. Вывод строки на терминал.

Урок 2. Простая команда

Основные понятия: ошибка в команде, имя и аргументы (ключи и операнды) команды, умолчание, интерактивное руководство.

Осваиваемые команды: «date», «cal», «man».

Содержание урока:

- 2.1. Имя команды. Команды «date», «cal».
- 2.2. Синтаксические и семантические ошибки при вводе команды.
- 2.3. Аргументы команды: ключи и операнды. Умолчания.
- 2.4. Страницы руководства. Команда «man».

Дополнительный материал: редактирование командной строки; гипертекстовая система помощи «info».

Урок 3. Файлы и файловые структуры

Основные понятия: файл, каталог, файловая структура, текущий каталог, домашний каталог, имя и шаблон имени файла.

Осваиваемые команды: «touch», «rm», «ls», «mkdir», «rmdir», «mv», «cp», «cd».

Содержание урока:

- 3.1. Файл. Имя файла. Допустимые в имени символы. Переносимый набор символов.
- 3.2. Создание файла: команда «touch». Удаление файла: команда «rm».
- 3.3. Каталоги и файловая структура. Просмотр каталога командами «ls», «ls -l».
- 3.4. Создание каталога командой «mkdir». Перемещение и

копирование файла командами «mv», «cp». Удаление каталога командами «rmdir» и «rm -r».

3.5. Текущий каталог. Смена текущего каталога командой «cd». Специальные имена каталогов «.» и «..». Краткое и полное имена файла.

3.6. Домашний каталог. Специальное имя каталога «~».

3.7. Шаблоны имен файлов. Специальные символы «*» и «?». Списки и диапазоны в квадратных скобках.

3.8. Экранирование специальных символов символом «\» и кавычками.

Урок 4. Ввод-вывод команды

Основные понятия: ввод и вывод, перенаправление ввода и вывода, стандартный ввод, стандартный вывод, вывод ошибок, дескриптор.

Осваиваемая команда: «cat».

Содержание урока:

4.1. Команда «cat». Ввод и вывод команды.

4.2. Перенаправление стандартного ввода конструкцией «<>».

4.3. Перенаправление стандартного вывода конструкциями «>>», «>>>».

4.4. Понятие дескриптора. Перенаправление вывода ошибок конструкцией «2>». Слияние стандартного вывода и вывода ошибок.

4.5. Стандартные файлы-устройства «/dev/null», «/dev/tty».

4.6. Оболочка как команда. Сценарий.

Урок 5. Правомочия доступа к файлу

Основные понятия: пользователь, главный пользователь, категория пользователей, права, маска прав, ссылка.

Осваиваемые команды: «chmod», «umask».

Содержание урока:

5.1. Права доступа. Команда «chmod» и «gwx»-нотация прав доступа.

5.2. Категории пользователей: владелец файла, группа-владелец файла, прочие.

5.3. Главный пользователь («root», суперпользователь).

5.4. Особенности прав на каталоги.

5.5. Маска прав на вновь создаваемые файлы по умолчанию. Команда «umask».

5.6. Ссылки (жесткие).

Дополнительный материал: числовая нотация прав доступа; символические ссылки.

Урок 6. Введение в процессы

Основные понятия: процесс, порождение и структура процессов, сигнал, задание (группа процессов). *Осваиваемые команды:* «ps», «fg», «bg».

Содержание урока:

6.1. Процесс. Порождение процессов. Исследование структуры процессов командой «ps».

6.2. Параметры процессов: идентификатор процесса, идентификатор родителя, время процесса, состояние процесса.

6.3. Наследование параметров при порождении процессов.

6.4. Сигналы; отправка сигналов из оболочки (команда «kill») и с терминала (Ctrl-C, Ctrl-Z).

6.5. Задания (группы команд). Задания переднего и заднего плана. Команды «fg» и «bg».

6.6. Сложные команды: конструкции «&», «;» и «(...)».

Урок 7. Переменные

Основные понятия: переменная (именованный параметр), присвоение и раскрытие переменной, экспорт, локаль.

Осваиваемые команды: «export», «set», «env».

Содержание урока:

7.1. Именованные параметры (переменные).

7.2. Определение переменной (присвоение значения) и вызов переменной (раскрытие).

7.3. Экспорт переменных. Команда «export».

7.4. Команды «set» и «env».

7.5. Стандартные переменные оболочки.

7.6. Локаль. Интернационализация и локализация команд.

7.7. Переменная \$PATH. Команда (внешняя) как файл.

Урок 8. Конвейер

Основные понятия: конвейер.

Осваиваемые команды: «tee», «sort», «uniq», «nl».

Содержание урока:

- 8.1. Связывание команд в конвейер конструкцией «|».
- 8.2. Дублирование вывода командой «tee».
- 8.3. Примеры: команды «sort», «uniq», «nl».

Урок 9. Интерактивное редактирование текста

Основные понятия: редактирование текста, регулярные выражения.

Осваиваемые команды: «vi», «sed».

Содержание урока:

- 9.1. Основные команды редактора «vi».
- 9.2. Потокное редактирование. Команда «sed».
- 9.3. Базовые регулярные выражения.

Уроки 10-11. Элементы программирования оболочки

Основные понятия: последовательное, условное и циклическое исполнение команд, код возврата.

Осваиваемые команды: «expr», «test», «read».

Содержание уроков:

10.1. Последовательное исполнение команд. Соединение команд знаком конца строки и конструкцией «;».

10.2. «Арифметические» и «логические» выражения: конструкция «\$(...)». Команда «expr».

10.3. Коды возврата. Команда «test»

10.4. Условное исполнение: конструкции «if ... then ... fi», «if ... then ... else ... fi» и «if ... then ... elif ... fi».

11.1. Циклическое исполнение с предусловием: конструкции «while ... do ... od» и «until ... do ... od» (Рис. 1-83).

11.2. Циклическое исполнение со списком значений: конструкция «for ... in ... do ... done».

11.3. Ввод-вывод: команды «echo», «read».

11.4. Передача аргументов сценарию и их раскрытие.

Дополнительный материал: определение и вызов функций; вызов сценариев; раскрытие команды (конструкция «` ... `»); включенный документ (конструкция «<<<>>>»).

Урок 12. Повторение пройденного

Содержание урока:

Пошаговое решение нескольких типичных задач.

Литература

1. ISO/IEC 9945-1:2003 (информационные технологии – интерфейс переносимых ОС) – Часть 1: «Базовые определения»; ISO/IEC 9945-2:2003 (информационные технологии – интерфейс переносимых ОС) – Часть 2: «Системные интерфейсы»; ISO/IEC 9945-3:2003(информационные технологии – интерфейс переносимых ОС) – Часть 3: «Оболочка и утилиты»; ISO/IEC 9945-4:2003 (информационные технологии – интерфейс переносимых ОС) – Часть 4: «Обоснование». – Zurich: ISO, 2003. Тж. доступен на: <http://www.unix.org/online.html>.
2. <http://inf.1september.ru>.
3. Максим Отставнов. Свободные программы и системы в школе. – М.: Институт логики, 2003. – 195 с., ил.. Тж. доступна на: <http://www.otstavnov.com/fsft> и на диске, прилагающемся к настоящему сборнику.
4. Брайан Керниган, Роб Пайк. UNIX – универсальная среда программирования. – М.: Финансы и статистика, 1992. – 304 с.: ил.

СОЗДАНИЕ И СТАТИСТИЧЕСКАЯ ОБРАБОТКА ПСИХОЛОГИЧЕСКИХ ТЕСТОВ СРЕДСТВАМИ ТП EXCEL

Е.А. Панина, Л.В. Гусева

г. Харьков, Академия пожарной безопасности Украины
fd.apbu@list.ru

В условиях интенсивного развития новых информационных технологий особую актуальность приобретает организация подготовки специалистов-психологов, владеющих современными компьютерными технологиями. На основе опыта подготовки психологов в Академии пожарной безопасности Украины представляем методику обучения студентов созданию и обработке результатов компьютерных психологических тестов на базе ТП MS EXCEL [1].

Особенность этой методики обучения заключается в том, что не требует от специалистов-психологов специальных знаний и подготовки в области программирования. Разработанный курс позволяет создавать практически любые компьютерные тесты, основываясь на базовых знаниях, которые приобретаются в рамках курса «Основы информатики и компьютерной техники».

Процесс создания компьютерного психологического теста можно разделить на три этапа. На первом этапе необходимо создать собственно вариант самого теста, т.е. ввести в компьютер все варианты вопросов и ответов, которые используются в тесте. На втором этапе необходимо создать обработчик теста, который позволяет обрабатывать полученные результаты с использованием ключа к тесту. На третьем этапе осуществляется компьютерное оформление самого теста т.е. разрабатывается необходимый интерфейс теста, который способствует концентрации внимания опрашиваемого на ключевых моментах на этом этапе выполняется скрытие и защита от тестируемого служебной информации теста (обработчик теста и ключ теста).

Каждый тест состоит из ряда вопросов, каждый вопрос имеет свой вес в зависимости от варианта ответа. Чаще используются варианты ответа: «да» или «нет».

Для создания самого теста студента достаточно основных

знаний по вводу и форматированию ячеек рабочего листа Excel. При создании обработчика теста используются стандартные функции ЕСЛИ(), СУММА(), а также проверка ввода, с целью предоставления пользователю возможных вариантов ответов на тест. При создании интерфейса теста применяются макросы, а также кнопки управления для обеспечения дружеского и понятного пользователю интерфейса. Пример теста для риск-тейкеров, созданного по предложенной методике представлен на рис.1

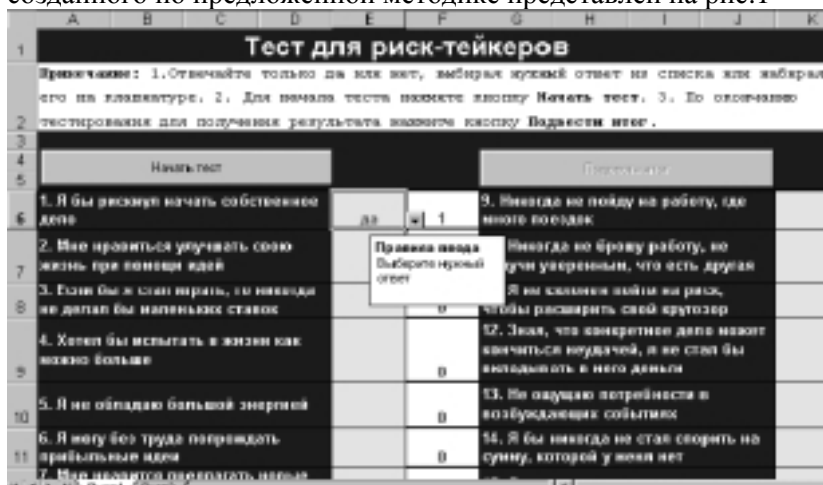


Рис. 1. Пример компьютерного психологического теста для риск-тейкеров.

Следует отметить, что после создания интерфейсной части теста легко можно создать новый тест путем замены вопросов теста и ответов теста и ключей, что позволяет намного ускорить создание компьютерных психологических тестов. Данная методика позволяет слушателям, получившим основные навыки работы в ТП EXCEL, создавать красочные, дружественные и эффективные компьютерные психологические тесты.

Литература

1. Яковлева И.А., Панина Е.А., Гусева Л.В., Шматко А.В. Методические рекомендации по выполнению контрольной работы по курсу «Основы информатики и компьютерной техники». – Харьков: АПУ, 2003. – 18 с.

ДЕЯКІ АСПЕКТИ НАВЧАННЯ КОМП'ЮТЕРНОЇ ЛЕКСИКИ НА КІБЕРНЕТИЧНИХ СПЕЦІАЛЬНОСТЯХ (НА МАТЕРІАЛІ АНГЛІЙСЬКОЇ МОВИ)

О.В. Пахомова

м. Кривий Ріг, Криворізька філія Інституту економіки та нових
технологій
helen@mail.ru

Як відомо, розвиток науки і техніки в наш час невід'ємний від загального обміну професійною інформацією між людьми, які розмовляють різними мовами. Практичне оволодіння іноземними мовами в сучасних умовах набуває особливо великого значення для фахівців різних професій.

Більшість вищих навчальних закладів України всіх рівнів акредитації як традиційного, так і альтернативного типу концентрують зусилля на підготовці спеціалістів першочергового попиту, а саме: спеціалісти в галузі комп'ютерних наук, мікроелектроніки, робототехніки. І це природно, адже з кожним роком наша держава все більше інтегрується у світову співдружність, глибшими і тіснішими стають стосунки з іноземними країнами, тому зростає потреба у спеціалістах, які, окрім належної фахової підготовки, володіють іноземними мовами.

Сфера інформаційних технологій постає зараз однією з найбільш перспективних та швидко зростаючих галузей сучасної науки. Розвиток науки та комп'ютерних технологій природно призвели до виникнення нового шару спеціальної лексики, який є складним за структурою та не завжди може бути зрозумілим через вузьку спрямованість. Проте, чим більше входять в сучасне життя нові технології, тим більше вони вносять в повсякденну мову технічні поняття, такі, як, наприклад, Інтернет, модем, принтер, ксерокс, мережа та інші, що зараз відомі майже всім. Спеціалісти в галузі інформаційних технологій та програмування повинні володіти не лише загальноповживаною технічною лексикою, а також базовою професійною термінологією та більш вузькими професійними поняттями.

Формування лексичних вмій та навичок постає однією з найважливіших цілей підготовки фахівців та навчання іноземній

мові у ВНЗ.

Оскільки майбутні спеціалісти повинні отримувати фахову інформацію з іншомовних джерел (переважно англійською мовою), вони мають опанувати технічну лексику на рівні вільного володіння. Основним засобом її вивчення є читання неадаптованих технічних текстів, при цьому головна увага приділяється саме лексичному, а не граматичному аспекту.

Ми вважаємо, що при вивченні комп'ютерної лексики треба також звертати увагу на мовні одиниці комп'ютерного жаргону, які часто трапляються в текстах комп'ютерної тематики, через те що частина формальної комп'ютерної лексики утворилася з комп'ютерного жаргону.

Загалом весь шар комп'ютерної лексики може бути поділений на технічну мову (*techspeak*) – формальну комп'ютерну лексику, яка зафіксована в спеціальних технічних словниках, також комп'ютерний жаргон (*jargon*) – неформальну розмовну лексику програмістів та користувачів. Чітку межу між технічною лексикою та комп'ютерним жаргоном провести дуже складно, тому що існує стабільна тенденція переходу жаргонізмів в комп'ютерні терміни, а також численна кількість жаргонізмів виникнула із узагальнення та переосмислення комп'ютерних термінів: наприклад, такі жаргонні слова як *comp*, *PeeCee* утворилися від комп'ютерних термінів *computer*, *PC* – *personal computer*. Семантичне походження мовних одиниць комп'ютерної лексики треба враховувати при їх введенні в мову студентів.

Отже, беручи до уваги семантичну складність усього пласту комп'ютерної лексики, на першому етапі вивчення необхідно синтезувати способи її викладення. Цей синтез має поєднувати перекладний, безперекладний, тлумачний, контекстуальний методи та метод мовної догадки. Наприклад, такі терміни як *hardware*, *software* зі складною семантикою потребують перекладу; аббревіатури мають бути розшифровані та розтлумачені. Значення деяких лексичних одиниць може бути зрозумілим із контексту. Наприклад: “Third Generation Computers began with the invention of the *microchip* or *integrated circuit* by Jack St. Clare Kilby.” В цьому реченні значення термінів *microchip* та *integrated circuit* зрозуміле з контексту, оскільки в українській мові існують

еквіваленти мікрочип та інтегральна схема.

На подальшому етапі вивчення комп'ютерної лексики великого значення набуває закріплення та актуалізація мовних одиниць в усній та писемній мові. Це може бути зроблено через різноманітні імітативні, репродуктивні та інші види комунікативних вправ.

Практичне оволодіння комп'ютерною лексикою на останньому етапі означає не тільки вміння самостійно читати та перекладати без допомоги словника літературу на англійській мові за спеціальністю та вільно висловлювати свою думку з заданої теми, а й вміння виконувати різноманітні творчі завдання англійською мовою.

Творчі завдання є не тільки кінцевою метою вивчення комп'ютерної лексики, а й одним з найефективніших засобів навчання. Різні види завдань творчого характеру, без сумніву, матимуть більш позитивний результат за умов, що студенти володіють певною мовною базою, тобто пройшли початковий етап вивчення іноземної мови та мають високий рівень мотивації.

Існує багато різновидів творчих завдань від достатньо простих (закінчити, розпочати чи доповнити певний текст необхідними елементами, дискусії, різні граматичні та лексичні вправи) до більш складних (написати діловий та дружній листи іноземному партнеру чи товаришеві, заповнити анкету, написати об'яву, скласти діалог, написати рецензію, реферат чи анотацію до тексту або твір на матеріалі вивченої теми).

Виходячи з мети та задач навчання іноземної мови на кібернетичних спеціальностях, ми вважаємо, що основними умовами успішного оволодіння комп'ютерною лексикою є професійні вміння та навички майбутніх спеціалістів, а також високий рівень мотивації. Провідними аспектами при навчанні комп'ютерної лексики є: по-перше, вивчення комп'ютерної лексики на матеріалі неадаптованої технічної літератури за фахом; по-друге, перевага лексичного аспекту навчання над граматичним; по-третє, відповідно добраний засіб викладення лексичних одиниць комп'ютерної лексики, беручи до уваги їх семантику; та на кінцевому етапі, практичне оволодіння комп'ютерною лексикою на рівні вільного читання текстів за фахом без допомоги словника та використання їх в усній та писемній мові.

ЗНАЙОМСТВО З ОС WINDOWS ТА MS-WORD У ХОДІ ВИВЧЕННЯ ІНФОРМАТИКИ

Ю.М. Пилипенко, Л.П. Голубєв
м. Київ, Київський національний університет
технологій та дизайну
GolubevL@ukrpost.net

Спеціаліст сучасного рівня повинен досконало володіти комп'ютерною технікою, без якої сьогодні важко уявити будь яку галузь науки, техніки, суспільства. Однією з поширених в світі операційних систем є ОС Windows, під керуванням якої працюють такі додатки, як MS-Office, вивчення яких передбачається окремими навчальними програмами.

З гіркотою хочеться визнати, що абсолютна більшість першокурсників недостатньо володіє відповідним шкільним матеріалом з інформатики, а студенти, що навчалися за межами міста, часто-густо взагалі бачили комп'ютер тільки по телебаченню. Серед тих же, хто має комп'ютер вдома, немало таких, що крім того, як включити-виключити та двічі клацнути мишою, щоб погратися у “розумні” ігри, мають слабке уявлення про роботу з ПК. Цікаво ще й таке спостереження – якщо у курсах, пов'язаних з вивченням інформаційних технологій, вивчення операційної системи та офісних додатків віддається на самостійну роботу, у студентів не має досконалого знання базового матеріалу – то там, то там виникають прогалини у знаннях, які доводиться латати під час вивчення більш складного матеріалу.

Тому ми вважаємо за необхідне на початковому етапі вивчення курсу інформатики у вузі виконати ряд лабораторних робіт, спрямованих на засвоєння основних прийомів роботи з операційною системою (на прикладі ОС Windows). Тим більше, що це не займає багато часу і робиться досить легко всіма студентами. При цьому ми ще й уникаємо такої психологічної проблеми, як побоювання роботи з ПК у тих, хто ніколи не працював з обчислювальною технікою і вважає, що він цим ніколи досконально не оволодіє, “бо в школі не було комп'ютера, а всі інші з ним вже давно працюють” – після перших же робіт у таких студентів з'являється впевненість у своїх силах.

Нижче подаються тексти лабораторних робіт, які ми пропонуємо для оволодіння ОС Windows та MS Word, та відповідне погодинне планування.

Тема	Лекції (год.)	Лабораторні роботи (год.)	Самостійна робота (год.)
ОС Windows	4	6	6
MS Word	6	6	6

ЛАБОРАТОРНА РОБОТА №1

Тема: Перше знайомство з комп'ютером та ОС Windows 2000. Папки: створення, копіювання, переміщення.

Завдання.

1. Виключити, а потім включити комп'ютер.
2. Перезавантажити комп'ютер за допомогою **Ctrl+Alt+Del**.
3. Знайти серед стандартних програм “Калькулятор”. Користуючись нею, підрахувати, скільки буде:
 - а) рік народження * число народження;
 - б) число народження : рік народження.
4. Чи можна змінити розмір цього вікна? Пересунути вікно на вільне місце.
5. Знайти серед стандартних програм “Блокнот”, ввійти в програму і набрати “Сьогодні 7 вересня”. Закрити вікно. Що відбулося? Як закрити вікно?
6. Налаштувати “Мій комп'ютер” так, щоб для кожного об'єкта відкривалося своє вікно («Мій комп'ютер» – Сервіс – властивості папки).
7. На “робочому столі” створити папку з назвою групи. Ввійти в папку. Пересуньте вікно. Змінити його розміри. Згорнути, розгорнути вікно.
8. У середині цієї папки створити нову папку під своїм прізвищем.
9. Визначити обсяг загального і вільного місця на диску С. У папці зі своїм прізвищем створити текстовий документ під своїм ім'ям. У тексті відобразити кількість загального і вільного місця на диску С. (Текстовий документ створити використовуючи “Блокнот”.)
10. Скопіювати створений файл у ту ж папку за допомогою

“Буфера обміну”. Як називається новий файл ? Замінити цю назву на “Інформація про диск”.

11. Скопіювати отримані файли в папку з назвою групи двома різними способами (один файл – одним, інший – іншим).
12. Згорнути усі вікна (не закривати, а згорнути).
13. Розгорнути вікно папки з назвою групи і перейменувати файл зі своїм ім’ям “файл 1” , а іншої в “файл 2”.
14. Закрити усі вікна.
15. Знайти “файл 1” – за допомогою команди “Пошук” головного меню, а файл “Інформація про диск” – за допомогою програми “Провідник”.
16. Відформатувати свою дискету. Перемістити створену папку з назвою групи на диск А.
17. Переконавшись, що створених папок немає на диску С.
18. Упорядкувати файли в папці вашої групи на диску А за алфавітом.
19. Закрити усі вікна.

Наш коментар до лабораторної роботи №1.

Робота, за бажанням викладача, може бути розбита на дві:

1. Первинне знайомство с ОС Windows.
2. Папки: створення, копіювання, переміщення.

Це залежить від контингенту студентів, з якими ви спілкуєтесь.

Звертаємо увагу, що викладачу слід на лекції розглянути питання про кодування інформації, одиницях, в яких вимірюється об’єм інформації, форматування дискети.

ЛАБОРАТОРНА РОБОТА №2.

Тема. Налаштування Windows-2000. Робота з “Корзиною” і “Довідкою”.

Завдання.

1. Використовуючи “Мови і стандарти”;
 - а) визначити, який роздільник дробової і цілої частини в числі;
 - б) вписати назви всіх режимів опції “числа”.

2. “Клавіатура”.
 - а) додати німецьку мову до стандартної розкладки мов клавіатури;
 - б) видалити німецький зі стандартної розкладки мов;
 - в) якою комбінацією клавіш можна переключатися між мовами при роботі з клавіатурою?
3. Використовуючи об’єкт “Система”, визначити:
 - тип процесора ЕОМ;
 - ємність ОЗП (оперативного запам’ятовуючого пристрою).
4. У папці “Мої документи” створити папку з назвою Вашої групи (якщо такої ще немає). У цій папці (використовуючи “Блокнот”) створити текстовий файл, у якому записати інформацію, отриману в пп. 1, 2в, 3). Файл назвати “Про систему”.
5. На робочому столі створити ярлик документа “Про систему”. Закрити усі відкриті вікна (**Alt+F4**).
6. Переглянути уміст файлу “Про систему”, використовуючи Ярлик – Документи – Пошук. У зошиті записати, як це робиться.
7.
 - а) Переглянути параметри вікна “Корзина” і настроїти їх так, щоб папки і файли не зникали відразу після їхнього видалення в “Корзину”. Інформацію про те, яку частину місця на жорсткому диску виділено під “Корзину”, записати в документ “Про систему”.
 - б) Видалити ярлик документа “Про систему” у “Корзину” при відкритому вікні папки, де знаходиться Ваш файл. Чи видалився файл із папки?
 - в) Видалити файл “Про систему” у “Корзину”.
 - г) Відновити файл “Про систему” з “Корзини”, використовуючи контекстне меню.
 - д) видалити ярлик документа з “Корзини”.
 - е) Файл “Про систему” скопіювати на дискету. Перевірте, чи відкривається ваш файл із дискети!
 - ж) Файл “Про систему” видалити з жорсткого диска. У зошиті записати, як це робити. З дискети файл не видаляти!
8. “Дата і час”.

- а) Налаштувати системну дату на свій день народження. Налаштувати час на 9 годин ранку.
 - б) Повернутися до початкових установок.
9. “Екран”.
- а) змінити фон, час активної роботи Windows-2000. У рухливому малюнку зробіть напис “Відмінник навчання”. У зошиті описати, як це робиться.
 - б) повернутися до попередніх установок.
10. “Миша”.
- а) змінити швидкість double click. Перевірте, чи спрацьовує він.
 - б) повернутися до колишніх установок.
- 11.
- а) Видалити зі стандартних програм “Калькулятор”.
 - б) Додати до стандартних програм “Калькулятор”. У зошиті описати, як це робиться.
- 12.
- а) Використовуючи інформаційну систему “Довідка” знайти інформацію про те, як роздрукувати документ. Виділивши мишею потрібний текст, уставте його після останнього запису у файл “Про систему”, що знаходиться на диску А. Для цього використовуйте “Буфер обміну”. Відкрийте вікно “Перегляд буфера обміну” для того, щоб подивитися, яка інформація там зберігається.
 - б) Роздрукуйте файл “Про систему” і підклейте роздруковку в зошит.

Наш коментар до лабораторної роботи №2.

При виконанні другої роботи студенти повинні самостійно знайти у довідковій системі “Довідка” інформацію про те, як роздрукувати документ і, включивши принтер, зробити це на практиці. Потрібно звернути увагу на налаштування “Корзини”, яке потрібно перед видаленням файлів.

ЛАБОРАТОРНА РОБОТА №3.

Тема. Графічний редактор Paint.

Завдання.

За допомогою графічного редактора **Paint** намалювати карту-схему, у якій відобразити, як пройти від будинку, де ви живете, до найближчої зупинки транспорту, записавши назви вулиць і їхнього номера по вашому шляху.

При цьому в малюнку обов'язково повинні бути:

- а) коло, еліпс, прямокутник;
- б) назви, розміщені вертикально і горизонтально;
- в) розфарбування елементів малюнка різними кольорами;
- г) зверху малюнка повинен йти напис з номером і назвою роботи, а також із прізвищем і ім'ям виконавця, зроблені різними розмірами шрифтів, причому прізвище повинне бути набране курсивом;
- д) записати файл на свою дискету, попередньо з'ясувавши і записавши в зошит, скільки він займає місця;
- е) роздрукувати файл.

Наш коментар до лабораторної роботи №3.

Робота з Paint проводиться тільки у комп'ютерному класі, де викладач показує, як працювати з тим чи іншим елементом малювання. Зверніть увагу, що перед тим, як щось намалювати, потрібно встановити розмір малюнка (бажано не більше 20x15 см.), оскільки растрові малюнки займають великий об'єм інформації. Необхідно познайомити студентів з форматами (.bmp, .jpg, .tif, .gif), у яких можна зберегти малюнки.

ЛАБОРАТОРНА РОБОТА №4.

Тема: Текстовий редактор MS-WORD.

Завдання.

1. Налаштувати «Параметри сторінки». Вибрати для набору тексту потрібну мову.
2. Занести довільний текст (1 сторінку). Текст має містити не менше 10 абзаців.
3. Третій абзац тексту перенести в кінець тексту. Зробити в першому абзаці червоний рядок 2 см. Зробити третій абзац завширшки 8 см.
4. Вирівняти: 4-й абзац по ширині сторінки, 5-й по центру, 6-й вправо, 7-й вліво.

5. Розставити нумерацію в 1-2 абзацах.
6. Розставити крапки пунктів перед 3-4 абзацом.
7. Перший абзац відформатувати шрифтом **Arial 14**, 2-й — **Arial 16**, 3-й — **Courier 10**.
8. Зробити 2-й абзац напівжирним, 1-й – курсивом, 5-й – підкресленим, 3-й – напівжирним курсивом, 10-й напівжирним, підкресленим, курсивом.
9. Встановити збільшений міжабзацний відступ перед 1-2 абзацами.
10. Зробити заголовок “І рости і діяти нам треба!” перед 3-м абзацом. Розбити першу сторінку на 3 рівні частини, уставивши примусовий перехід на нову сторінку. Розставити нумерацію сторінок згори по центрі, включаючи першу сторінку.
11. Речення останнього абзацу розставити по одному в комірки таблиці 4 на 4. Зайві речення відкинути, недостаючі комірки не заповнювати.
12. Поміняти в таблиці місцями 3 і 1 стовпчики. Відформатувати в різні способи всі 4 рядки. Зробити різні абзацні відступи для кожного стовпчика.
13. Вставити попереду 1 стовпчик і розставити в ньому нумерацію рядків.
14. Відсортувати таблицю по 4 стовпчику. Зробити тло в 1-му рядку. Відокремити 1-й рядок подвійною лінією.
15. Провести обмежувальні лінії таблиці: внутрішні тонкими лініями, зовнішні – товстими.
16. За допомогою “Редактора формул” вставити в кінці тексту формули:

$$\cos x = \sum_{i=0}^{\infty} \frac{x^{2i}}{(2i)!}, \quad \cos^2 x = \frac{1 + \sin 2x}{2}, \quad \int \sin x dx = -\cos x$$
17. Скласти таблицю вигляду, використовуючи, де потрібно, “формули”.

Продажів по кварталах	1	2	3	4	Всього по товару (Σ)
Зерно	65	78	78	223	444
Сир	345	147	261	69	822
Всього (Σ)	411	227	342	296	1266

Записати текст і таблицю у власну папку.

Наш коментар до лабораторної роботи №4.

Приведений один варіант роботи. Для кожного студента бажано мати окремий варіант. Необхідно звернути увагу, що перед тим, як набрати текст:

1. Потрібно встановити параметри сторінки.
2. Вибрати мову, на якій буде набиратися текст.

Під час перевірки роботи акцентуйте увагу на те, як програма “перевіряє” орфографію, як потрібно коректувати текст та дані в таблицях (настроювати потрібну висоту та ширину рядків та стовпчиків, робити потрібні лінії, проводити обчислення за формулами у таблицях і т.д.), як винести на панель інструментів “Редактор формул”, якщо його там немає. Обов’язково навчити продивлятися текст у режимі “Попередній перегляд”, звернути увагу на формати, у яких зберігаються файли MS-Word, приділивши увагу (.rtf).

ЛАБОРАТОРНА РОБОТА №5

Тема: MS Word: майстри і шаблони, створення і зв’язування малюнків.

Завдання.

1. Використовуючи майстер листів, напишіть лист бабусі, у якому: укажіть свою адресу, запишіть інформацію про батьків (до 2 коліна, тобто бабусі, дідусі, мати, батько, де народилися).
2. У лист після адреси вставте малюнок, що був зроблений раніше в Лабораторній роботі №5 (графічний редактор Paint), забезпечивши **зв’язок** графічного файлу з файлом MS WORD. Не виходячи з MS Word, зробіть зміну в назві малюнка “Лабораторна робота 5. Текстовий редактор Word”.
3. Після інформації про батьків, за допомогою “Автофігур” панелі “Малювання”, зобразите генеалогічне дерево від другого рівня до вас, використовуючи різні елементи “Блок-схеми”.
4. За допомогою “Попереднього перегляду” оцініть лист і

зробіть так, щоб він розміщалося на одній сторінці документа. Доможіться того, щоб його можна було помістити на дискету. Якщо інформація не вміщається, то зменште розмір малюнка в графічному редакторі Paint.

5. Переконавшись (!), що інформація вміщається на дискету (тобто файл займає менш 1.44 Мб), очистіть дискету – перенесіть файл із графічним об'єктом на жорсткий диск у папку з номером групи. Перекопіюйте Ваш лист на дискету і роздрукуйте файл. Підклейте текст завдання й отриману роздруківку в зошит.

Наш коментар до лабораторної роботи №5.

Робота дає можливість познайомитись з “майстрами” MS-Office. Крім того, студенти знайомляться з OLE – технологією зв'язку та й впровадження файлів, що підготовлені за допомогою різних додатків.

О РЕАЛИЗАЦИИ ПРАКТИКУМА ПО ПРОГРАММИРОВАНИЮ ЛЕКСИЧЕСКИХ И СИНТАКСИЧЕСКИХ АНАЛИЗАТОРОВ ПРИ СОЗДАНИИ ЯЗЫКОВЫХ ИНТЕРПРЕТАТОРОВ

А.П. Полищук, С.А. Семериков

г. Кривой Рог, Криворожский государственный педагогический
университет
cc@kpi.dp.ua

Требования к квалификации учителя информатики средней школы (что должен знать и уметь) Министерством образования и науки до настоящего времени не определены. Более того, в номенклатуре специальностей педагогических университетов специальность «Информатика» отсутствует, информатика проходит как некоторое дополнение к предметной области в виде «Учитель физики (химии, математики, трудового обучения и т.п.) и основ информатики». При этом под «Основами информатики» часто понимается не система питающих научное направление корней (методов и средств решения основных кибернетических задач), а первичная компьютерная грамотность типа «мышкой по кнопочкам».

В то же время реальные, выдвигаемые практикой, требования к учителю информатики достаточно высоки – он, помимо основной работы по преподаванию курса «Алгоритмизация и программирование», является в школе единственным проводником для внедрения компьютерных технологий в учебный процесс по многим школьным дисциплинам, должен хорошо ориентироваться в системном администрировании и методах защиты программного обеспечения, в приложениях, которые к собственно информатике отношения не имеют (текстовые, графические, музыкальные редакторы и пр.).

Это связано, прежде всего, с созданием и использованием предметно-ориентированных обучающих и тестирующих программ. При работе с такими программными изделиями рано или поздно приходится сталкиваться с необходимостью анализа (лексического, синтаксического, семантического) текстов на формализованных языках и их интерпретации. Этими текстами мо-

гут быть математические формулы или ответы на тестирующие вопросы (не столь простые, как выбор одного из предложенных вариантов ответа), или программы на алгоритмическом языке.

Нелегкие для восприятия курсы системного программирования, связанные с изучением формальных грамматик и их использования в задачах синтаксического анализа, всегда присутствуют в вузовской подготовке специалистов по информатике и обязательно должны быть, по нашему мнению, в арсенале школьного учителя, получающего право преподавания информатики.

Классический курс синтаксического анализа включает в себя:

- теории языков и формальных грамматик (классификация и способы определения языков, грамматики и метаязыки для их описания, генерация языков на основе грамматик);

- постановку и методы решения задач лексического анализа (классификация лексем и их выделение из входного потока);

- постановку и методы решения задач синтаксического анализа (правильность следования лексем и методы вычисления выражений);

- методы создания и использования программ-генераторов программных лексических и синтаксических анализаторов;

- программирование интерпретаторов и компиляторов алгоритмических языков программирования.

Полная реализация этого курса в практикуме по программированию невозможна даже в классических университетах на профильных специальностях, не говоря уж о слишком коротком (и непрерывно укорачиваемом) прокрустовом ложе учебных планов по информатике в педагогических университетах (в лучшем случае – 36 часов на курс синтаксического анализа и интерпретации).

Понимая важность и практическую полезность курса, мы начали его реализацию еще при изучении дисциплины «Алгоритмизация и программирование»: разбор текстов – прекрасная тренировка для развития алгоритмического мышления и полезнее наборов задач типа «вычислить сумму произведений нечетных элементов четных строк матрицы на четные элементы нечетных столбцов». Первая задача программирования простейшего лексера, поставленная перед студентами, формулировалась так:

составить функцию, осуществляющую ввод и распознавание десятичных, восьмеричных, шестнадцатеричных целых и вещественных чисел. За один вызов функции должно возвращаться одно числовое значение или признак того, что введенные данные не являются числом. Для контроля правильности распознавания распечатывалось как распознанное значение, так и результат его преобразования в числовую форму. Для выполнения работы студентам предоставлялась достаточно подробная методическая помощь, о характере которой можно судить по следующему фрагменту, касающемуся определения системы счисления:

Анализ следующего за знаком считанного символа позволяет предположить тип лексемы:

```

если в ch содержится не '0', то, вероятно, это целое или
вещественное число
|   установим тип лексемы в DEC
|   если ch - не цифра, то
|   |   установим тип лексемы в NON
|   -
иначе
|   если в ch содержится '0', то проверяем, число 8-ричное или
|   16-ричное
|   |   добавляем ch в массив parse_result и увеличиваем
|   |   parse_length на 1
|   |   считываем в ch следующий символ
|   |   если в ch содержится 'x' или 'X', то число 16-ричное
|   |   |   установим тип лексемы в HEX
|   |   |   добавляем ch в массив parse_result и увеличиваем
|   |   |   parse_length на 1
|   |   |   считываем в ch следующий символ
|   |   иначе
|   |   |   если в ch содержится '.', то число вещественное и
|   |   |   начинается с 0
|   |   |   |   установим тип лексемы в DBL
|   |   |   |   добавляем ch в массив parse_result и увеличиваем
|   |   |   |   parse_length на 1
|   |   |   |   считываем в ch следующий символ
|   |   |   иначе
|   |   |   |   установим тип лексемы в OCT
|   |   |   -
|   |   -
иначе
|   |   если обнаружен признак конца ввода
|   |   |   завершаем работу функции, возвращая -1
|   |   -
-

```

-
Дальнейшие действия выполняются в цикле до тех пор, пока лексема не будет распознана полностью либо пока не произойдет ошибка распознавания.

Схематически их можно изобразить следующим образом:

```
for(;!parse_error && !end;)  
{  
    switch(mode) //выбор типа лексемы  
    {  
        case DEC: //десятичное целое - действия  
            break;  
        case OCT: //восьмеричное целое - действия  
            break;  
        case HEX: //шестнадцатеричное целое- действия  
            break;  
        case DBL: //вещественное число - действия  
            break;  
        case NON: //не число  
            break;  
    }  
}
```

По окончании цикла нам останется только проверить, была ли при распознавании ошибка, и не связана ли она с концом ввода. Если это так, признак ошибки можно сбросить установкой значения `parse_error` в 0.

Доведенная до детализации алгоритма методическая помощь предоставлялась и по каждому распознающему действию. Всего студенты выполняли 3-4 подобных заданий и на следующем этапе ставилась задача программирования интерпретатора функций одной переменной, заданных в параметрическом виде с графической иллюстрацией в заданном диапазоне изменения параметра.

На этом этапе осваивается алгоритм синтаксического разбора выражений, которые могут строиться из следующих элементов: числа, операторы `+` `-` `/` `*` `^` `%` `=` `()` `<>` `;`, переменные, именованные константы, математические функции из набора, поддерживаемого стандартной математической библиотекой C или C++. Все лексемы классифицируются и переводятся во внутреннее числовое представление:

```
//Коды классов лексем  
enum LexemClass {OPERAND=1, OPERATION, BRACKET, EOL};  
//Подклассы (коды) операндов  
enum OperandCodes {STRING=1, //Строка  
                   NUMBER, //Число  
                   VARIABLE, //Переменная  
                   CONSTANT, // Именованная константа
```

```

EXPRESS}; // Выражение в скобках
//Коды операций
enum OperationCodes
{ MUL=1,DIV,POW,PLUS,MINUS,ABS,ACOS,ASIN,ATAN, COS,
  COSH, EXP,LOG,LOG10,SIN,SINH,SQRT,TAN,TANH};
//Массив структур для именованных констант
struct cnst {
    char cn[6];//Имя константы
    double cv; //Значение константы
}tc[2]={{ "PI",M_PI},{ "E",M_E}};
//Все обозначения (имена) операций и их коды сведем в таблицу
//(массив структур по шаблону)
struct {
    char on[10];//Имя операции
    int ov;      //Числовой код операции
}op[]={
    {"ABS",ABS}, {"ACOS",ACOS}, {"ASIN",ASIN}, {"ATAN",ATAN},
    {"COS",COS}, {"COSH",COSH}, {"EXP",EXP}, {"LN",LOG},
    {"LOG",LOG10}, {"SIN",SIN}, {"SINH",SINH}, {"SQRT",SQRT},
    {"TAN",TAN}, {"TANH",TANH}, {"+",PLUS}, {"-",MINUS},
    {"*",MUL}, {"/",DIV}, {"^",POW}, {"",0}};
char *OP="+-*/^"; //Перечень арифметических операций
// Коды ошибок ввода пользователя
enum InputErrors {
    SyntaxError, UnpairedParentheses, NotExpression, NotVariable,
    InvalidOperation, InvalidConstName, DivisionByZero
};

```

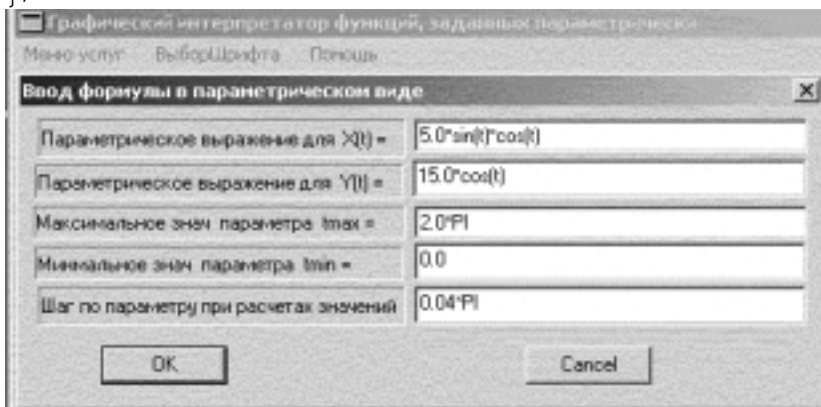


Рис. 1. Вид диалогового окна для ввода интерпретируемых формул

На этом этапе основная трудность состояла в освоении студентами одного из методов синтаксического анализа – рекурсив-

ного спуска или последовательной свертки выражений в соответствии с приоритетами операций. Метод рекурсивного спуска давался трудно и для экономии времени на этом этапе мы ограничились интуитивно легко воспринимаемым методом свертки выражений, используя рекурсию только для раскрытия вложенных скобочных выражений. Материал подавался одновременно с методами программирования интерфейса пользователя, построения графиков функций и других сопутствующих разделов программирования (рис. 1, 2).

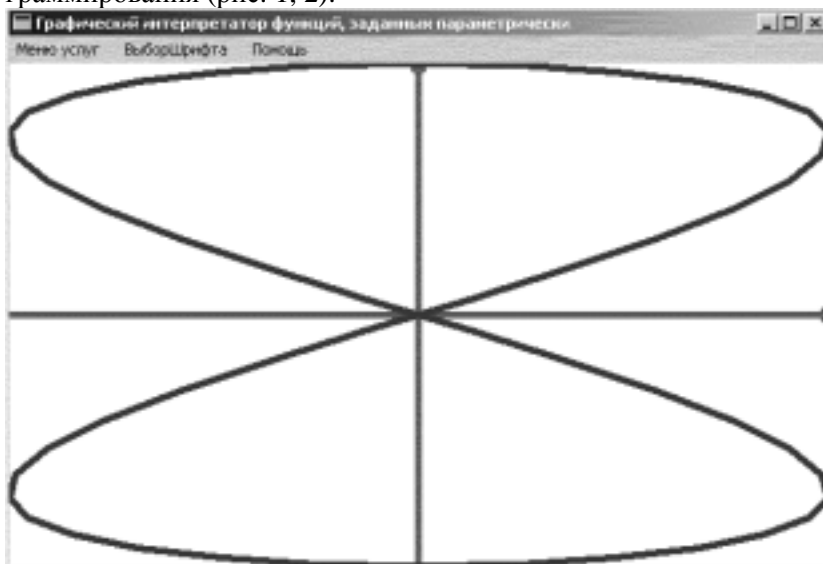


Рис. 2. Результат интерпретации в виде графика функции

После такой предварительной подготовки следующий этап, посвященный синтаксическому анализу формализованных алгоритмических языков программирования, реализовывался в составе 36-часового практикума по программированию в составе спецкурса «Синтаксический анализ и интерпретация языков программирования» и включал в себя разработку интерпретатора упрощенного варианта языка С без использования метаязыка для описания его грамматики в форме Бэкуса–Наура. Выбор С в качестве объекта разработки и тестирования объясняется тем, что программирование интерпретатора осуществлялось на этом языке и выбор любого другого языка для интерпретации привел бы

к потерям и без того скудного времени на его освоение. Кроме того, интерпретация того же языка, на котором реализован интерпретатор, позволяет глубже понять его свойства и особенности.

В качестве основы был взят двухпроходный интерпретатор Г. Шилдта [3]. В первом проходе составлялись таблицы глобальных переменных и функций тестируемого текста программы; второй проход начинался с первого оператора главной функции и завершался ее последним оператором с временными переходами в вызываемые вспомогательные подпрограммы любого уровня вложенности. Здесь уже использовался метод рекурсивного спуска в его классическом варианте. В составе программы – три основные функции: лексический и синтаксический анализаторы и собственно интерпретатор, выполняющий действия, предусмотренные блоком операторов, ограниченных парой фигурных скобок с рекурсивными вызовами по уровням вложенности. Эта низкоуровневая фаза во многом перекликается с пройденным ранее программированием интерпретатора формул и поэтому воспринималась относительно неплохо, позволяя в этот период осуществить начитку лекций по формальным грамматикам и генераторам лексических (на примере FLEX, совместимого с широко известным генератором LEX) и синтаксических (на примере BISON, совместимого с YACC) анализаторов.

После этого была поставлена задача выполнить описание грамматики языка в БНФ для LEX, составить операторы распознавания и сопутствующих действий, выполнить генерацию кода лексического анализатора и заменить собственный вариант лексического анализатора анализатором, сгенерированный с помощью LEX. Описание грамматики иллюстрирует следующий фрагмент:

```
LETTER      [A-Z_a-z]
NOLETTER    [^A-Z_a-z]
DIGIT       [0-9]
IDENT       {LETTER}({LETTER}|{DIGIT})*
NUMBER      {DIGIT}+
WHISP1      [ \t\r]
WHISP       ({WHISP1}|\n)
.....
```

Секция распознавания и действий иллюстрируется фрагментом программы для ключевых слов, идентификаторов, чисел, операций, комментариев и пр.:


```

%%
<S>"int" { strcpy(token,yytext);tok = INT;
fprintf(yyout,"%s\n",token); return token_type=KEYWORD; }
<S>"char" { strcpy(token,yytext);tok = CHAR;
fprintf(yyout,"%s\n",token); return token_type=KEYWORD; }
<S>"while" { strcpy(token,yytext);tok = WHILE;
fprintf(yyout,"%s\n",token); return token_type=KEYWORD; }
. . . . .
<S>{IDENT} { strcpy(token,yytext);tok =
STRING;fprintf(yyout,"%s\n",token); return token_type =
IDENTIFIER; }
<S>{NUMBER} { strcpy(token,yytext);tok=0;
fprintf(yyout,"%s\n",token); return token_type=NUMBER; }
<S>"+" { strcpy(token,yytext);tok=PLUS;
fprintf(yyout,"%s\n",token); return (token_type = ARITHOP ); }
<S>"-"{ strcpy(token,yytext);tok = MINUS;
fprintf(yyout,"%s\n",token); return(token_type=ARITHOP ); }
<S>"*"{strcpy(token,yytext); tok = MUL;
fprintf(yyout,"%s\n",token); return( token_type=ARITHOP ); }
. . . . .
<S>"{"{strcpy(token,yytext);tok = 0;
fprintf(yyout,"%s\n",token); return(token_type=DELIMITER ); }
<S>"}" { strcpy(token,yytext);tok = 0;
fprintf(yyout,"%s\n",token); return(token_type=DELIMITER ); }
. . . . .
<S>"/**" { BEGIN COM; }
<COM>[^*]* { }
<COM>"*/" / [^/] { }
<COM>"*/" { BEGIN S; }
<S>{WHISP}+ { }
%%

```

При использовании сгенерированного лексера в предварительном проходе по тексту пользовательской программы составлялись таблицы глобальных переменных и функций, а также массив записей со сведениями о лексемах. Записанные в дисковый файл для облегчения отладки, эти таблицы для одного из тестирующих вариантов пользовательских программ имеют вид:

Таблица функций с именем, кодом возвращаемого типа и строкой местонахождения в тексте:

FuncName -	main	RetType -	2	Loc -	10
FuncName -	sum	RetType -	2	Loc -	111
FuncName -	print_alpha	RetType -	2	Loc -	147

Таблица глобальных переменных (имя, код типа и значение):

VarName -	i	VarType -	2	Value -	0
VarName -	j	VarType -	2	Value -	0
VarName -	ch	VarType -	1	Value -	0

Фрагмент таблицы лексем (индекс, обозначение, тип, подтип):

Index -	0	Name -	int	Type -	3	Tok -	2
Index -	1	Name -	i	Type -	1	Tok -	5
Index -	2	Name -	,	Type -	0	Tok -	0
Index -	3	Name -	j	Type -	1	Tok -	5
Index -	4	Name -	;	Type -	0	Tok -	0
Index -	5	Name -	char	Type -	3	Tok -	1
Index -	6	Name -	ch	Type -	1	Tok -	5
Index -	7	Name -	;	Type -	0	Tok -	0
Index -	8	Name -	int	Type -	3	Tok -	2
Index -	9	Name -	main	Type -	1	Tok -	5
Index -	10	Name -	(Type -	0	Tok -	0
Index -	11	Name -)	Type -	0	Tok -	0
Index -	12	Name -	{	Type -	7	Tok -	0
Index -	13	Name -	int	Type -	3	Tok -	2

Наличие таких таблиц максимально упрощает не только получение очередной лексемы простым наращиванием индекса массива, но и получение любой предыдущей или следующей лексемы при последующем синтаксическом анализе.

Несмотря на описанную выше предварительную подготовку, два следующих этапа (использование генератора синтаксических анализаторов BISON и ознакомление с методами программирования генераторов кода) реализованы не были – отведенного учебным планом времени для этого оказалось недостаточно.

Выводы

Обобщая опыт реализации описанного спецкурса и ряда других курсов по информатике, направленных на повышение уровня подготовки специалистов в этой области, хотелось бы отметить следующее. Информатика (по существу синоним термина «кибернетика») – очень непростая для освоения наука, которая не должна изучаться как «попутный» с другой наукой предмет, если ставится задача подготовки квалифицированного специалиста в этой области. Учитывая очевидную тенденцию ко все более раннему началу изучения иностранных языков и компьютерных дисциплин в общеобразовательной средней школе (начиная с младших классов) и острый дефицит квалифицированных преподавателей в этой области как в школе, так и в вузах, назрела необходимость внесения в номенклатуру учительских специальностей для педагогических университетов отдель-

ной квалификации «Учитель информатики» с фундаментальной подготовкой по математическим методам решения основных кибернетических задач (теории оптимального управления, методам идентификации управляемых систем, исследования операций), анализу алгоритмов и компьютерному программированию и т.д.

При сохранении существующего положения с квалификацией преподавателей информатики удлинение сроков обучения в школе будет иметь тот же плачевный результат, что и многолетнее изучение иностранных (а часто и родных) языков в школе и затем в вузе, когда для перевода студентом простейших англоязычных сообщений компилятора типа «Invalid parameter» приходится предоставлять программу-переводчик.

Литература

1. Ахо А.В., Сети Р., Ульман Д. Компиляторы: принципы, технологии и инструменты. – М.: Издательский дом “Вильямс”, 2001.
2. Кнут Д. Искусство программирования. Т.1. Основные алгоритмы. – 3 е изд. – М.: Издательский дом “Вильямс”, 2000.
3. Шилдт Г. Полный справочник по С. – 4-е издание. – М.: Издательский дом “Вильямс”, 2000.
4. Хантер Р. Основные концепции компиляторов. – М.: Издательский дом “Вильямс”, 2002.

МОТИВАЦІЙНА ФУНКЦІЯ СИСТЕМИ ЗАДАЧ У НАВЧАННІ ІНФОРМАТИКИ

Н.І. Праворська

м. Хмельницький, Хмельницький державний університет
margana@ukrpost.net

Впровадження в практику навчання інформатики прикладних, практичних задач, вправ – один із шляхів удосконалення процесу навчання, активізації пізнавальної діяльності студентів, що посилює світоглядні аспекти навчання. Водночас, широке використання в навчальному процесі мотиваційної функції задач сприяє усвідомленому сприйняттю студентами програмного матеріалу, оволодінню міцними знаннями, розвиткові їх логічного мислення. Рівень мотивації вивчення нового матеріалу з інформатики суттєво можна підвищити за рахунок задач з практичним змістом. Життєвою необхідністю розв'язання подібних задач є найбільш природно обґрунтувати необхідність у нових знаннях, методах з інформатики.

Акцентування уваги на необхідності оволодіння теорією під впливом необхідності практики сприяє формуванню в студентів наукових поглядів. Використання задач для мотивації знань, умінь, методів створює умови для реалізації на етапі введення нового навчального матеріалу міжпредметних зв'язків, зв'язок навчання інформатики з майбутньою професійною діяльністю.

А широкий підбір задач, вправ з практичним змістом, лабораторні роботи, питання-завдання приводять не тільки до необхідності набувати нові знання та вміння, але й до застосування набутих під впливом цієї необхідності знань для розв'язання поставленої та широкого кола інших завдань.

Тому, враховуючи потреби та запити сьогодення, такими важливими є інтеграційні процеси інформатики та економіки. Завдання економічної освіти студентів по-новому постали перед вищими навчальними закладами, тому що з розвитком ринкових відносин в Україні особливо актуальним стає розвиток економічного мислення, виховання людини, здатної сприймати ринкові перетворення, легко адаптуватися до змін у житті, вільно оперувати основними економічними поняттями, володіти практични-

ми навичками діяльності в умовах ринку.

Таким чином, в студентів з'являється пізнавальний інтерес не тільки до змісту навчання, а й до способів отримання знань стосовно цього змісту. Реальну збуджувальну силу до діяльності має пізнавальний інтерес саме до способу отримання знань. Тому в процесі навчання такими важливими є використання різних форм практичних, самостійних, творчих робіт, елементів дослідження.

Навчальна діяльність студентів є головною їхньою діяльністю, тому інтелектуальний розвиток і професійне становлення, хоч і не виключно, але в основному, відбувається у процесі навчальної діяльності. Навчальна діяльність студентів багатогранна і досить велика за обсягом. Це пов'язано із значною кількістю предметів і досить великою складністю частини з них.

Для підвищення рівня навчальної діяльності необхідно продовжувати формувати у студентів загальні розумові дії і прийоми розумової діяльності, підсилювати мотивацію навчання і використовувати традиційні та нові технології, сучасні інформаційні технології, які активізують і інтенсифікують навчально-пізнавальну діяльність.

Проблеми мотивації і врахування потреб особистості студента є центральним пунктом дослідження властивостей особистості. “Жодну якість особистості не можна зрозуміти і пояснити, якщо невідомо, для задоволення якої потреби виникла ця якість” [1]. Мотиваційно-потребова сфера – складне психологічне явище, в якому стійко домінуючі мотиви створюють направленість особистості.

Мотиваційно-особистісний аспект пов'язаний із формуванням позитивних навчальних мотивів і особистих якостей майбутнього спеціаліста, дійових цілей, оскільки мотиви і цілі є важливішими детермінантами діяльності. Структура мотивів студента, яка формується в період навчання, стає стрижнем особистості майбутнього спеціаліста. Отже, розвиток позитивних навчальних мотивів – невід'ємна складова процесу навчання і виховання студентів, хоча не завжди цілеспрямовано здійснюється. Для ефективної практичної роботи в цьому напрямку потрібне серйозне вивчення закономірностей розвитку навчальної мотивації студентів.

Механізмом навчальної мотивації є формування цілісної структури цілей навчальної діяльності. Звідси випливає важливість своєчасної і систематичної постановки викладачами цілей навчання, які студенти мають прийняти і спрямувати свою діяльність на досягнення поставлених викладачем і самостійно цілей навчання в процесі розв'язання завдань.

Мотиваційна сфера майбутнього спеціаліста може формуватися лише в умовах повноцінного особистого спілкування з досвідченим, творчо мислячим, висококваліфікованим і захопленим своєю професією викладачем.

З першого курсу необхідно показувати студентам суспільну значимість обраної ними професії і важливість розвитку студентом свої професійно значущих якостей.

Принцип вмотивованості студентів визначає позицію і відношення студентів до участі в педагогічному в процесі.

Вмотивованість студентів – це їхнє зацікавлене відношення до удосконалювання фахової компетентності і загальної культури. При цьому забезпечується самостійне, активне просування до вершин майстерності, з орієнтацією на власні зусилля в навчальному процесі, потреби і цілі, мотиви і волю. Причому організація навчально-пізнавальної діяльності і керівництво нею з боку викладача орієнтується на їхню творчу участь у навчальному процесі.

Це потребує від викладача такої організації занять, щоб кожен студент розумів значимість навчання в цілому, цілей і задач навчального предмета, кожного заняття, взаємозв'язок їхнього змісту із змістом попередніх занять, формував усвідомлене позитивне відношення до навчального процесу.

Від сили та структури мотивації в значній мірі залежить як навчальна активність студентів, так і їхня успішність. При досить високому рівні розвитку навчальної мотивації вона поповнює відсутність спеціальних здібностей або відсутність у студентів необхідних знань, умінь та навичок. Вона може бути створена шляхом відповідної організації заняття, постановкою проблемних, пізнавальних задач або задач з виробничим змістом.

Мотиваційний етап дидактичного процесу дозволяє швидко включити студента у пізнавальну діяльність. Для здійснення цього етапу підбираються завдання, показуються шляхи

розв'язання на основі знань із даної предметної галузі.

Загальний принцип відбору та побудови системи мотиваційних завдань полягає у їх професійній спрямованості, що дозволяє утримувати увагу студентів на протязі заняття та всього терміну вивчення предмету.

Одна з універсальних методик мотивації – показ студентам тих реальних виробничих умов і задач, в яких їм необхідно буде використовувати знання, одержані з даного предмету.

Прикладом може бути завдання, яке полягає у побудові складних таблиць та використання надписів та рисунків при створенні об'яв (рис. 1).



2-35-57	2-35-57	2-35-57	2-35-57	2-35-57	2-35-57	2-35-57	2-35-57
---------	---------	---------	---------	---------	---------	---------	---------

Рис. 1 Приклад ілюстрації можливостей редактора Microsoft Word з метою формування навчальної мотивації

Пізнавальний інтерес збуджується в процесі навчання, створює найкращі умови для повного розкриття індивідуальності і є основою для розвитку творчих здібностей студентів й формування в них професійних схильностей [5]. Формування навчаль-

ної діяльності та позиції студента в ній допомагає встановити необхідний рівень інтересу до навчання на основі виникнення пізнавального мотиву, під яким розуміють як кращу орієнтацію студента на оволодіння способами дій [3].

До мотиваційного компоненту відноситься пізнавальна ініціатива (прагнення знати те чи інше явище за власним бажанням, більш в ньому розбиратися) навчання, яке повинно бути організовано так, щоб були утворені умови, які спонукають, сприяють, стимулюють, активізують розвиток студента. Викладачі повинні знати, що розвиток мотивації навчання не самоціль, а лише один з засобів розвитку діяльності студента. Можна виділити 6 етапів актуалізації та засвоювання мотиву [3].

1. Привертання уваги студентів до змісту курсу, розвитку індивідуальних творчих здібностей, самоосвіти, визначення та уточнення змісту мотивації навчання.

2. Виявлення можливостей мотивації, міркування студентів в ході виконання спеціальних вправ набуття досвіду мислення, оцінювання емоційного реагування, що відповідає даному мотиву.

3. Початок і хід формування мотивації під час організації навчальної діяльності здійснюється викладачем шляхом переключення з одного виду діяльності на інший, навчання студентів прийомам самостійної постановки мети та її досягнення.

4. Аналіз причин зниження мотивації навчання, співставлення кожним студентом даного мотиву зі своїми ідеалами та особистісне зняття мотиву, вмотивованого використання студентами знань, одержаних в ході навчання під керівництвом викладача.

5. Вивчення індивідуальних особливостей мотивації, професійної поведінки студента у відповідності з мотивом.

6. Контроль викладача за індивідуальними особливостями мотивації. Одним із результатів вивчення мотивації може бути характеристика мотивації навчання студентів, що включає в собі стан мотивів та мети, знань студентів, рівень сформованості навчальної діяльності.

Дослідження психологів [5] щодо взаємообумовленості мотивів та мети свідчать про те, що чим більше віддалені цілі ставить перед собою студент, тим ширші і чіткіші мотиви у нього

проявляються. У студентів, які бачать перед собою абсолютно визначену перспективу, глибші та усвідомленіші мотиви навчання. Для діяльності суб'єкта навчання характерним є осмислення мети та бажаних результатів діяльності, виявлення та з'ясування причин відхилення від мети; уточнення та коригування структури діяльності. Усвідомлення кожного етапу структури є одним із напрямків формування мотивів професійного становлення. А в тих студентів, у яких не сформовано чітке уявлення про перспективу у професійній діяльності, мотиви навчання є нестійкими.

Особлива роль у підвищенні рівня викладення інформатики у вищих технічних навчальних закладах, на наш погляд, повинна належати теорії і практиці формування та розвитку мотивації навчальної діяльності студентів. Саме із змінами у мотиваційній сфері студента пов'язано розвиток пізнавального інтересу до дисципліни, зокрема до інформатики.

Поняття мотивації включає в себе всю сукупність факторів, механізмів, процесів, що забезпечують виникнення спонукань до предметної або професійно спрямованої діяльності.

За С.Л. Рубінштейном [6] кожне явище набуває мотиваційного значення, тобто співвідноситься із системою уявлень суб'єкта навчання.

Особлива роль в цьому процесі повинна належати, на наш погляд, теорії і практиці формування та розвитку мотивації навчальної діяльності студентів. Саме із змінами мотиваційної сфери особи пов'язані пробудження і розвиток пізнавального процесу, в тому числі і до гуманітарних дисциплін.

Поняття мотивації взагалі включає в себе всю сукупність факторів, механізмів, процесів, що забезпечують виникнення спонукань до реальної або потенційної предметної, конкретно-направленої активності. При цьому розділяють внутрішні фактори мотивації і так звану “зовнішню мотивацію” – сукупність зовнішніх умов, що необхідно детермінують певну діяльність людини. Внутрішні фактори являють собою складну систему оціночних відносин потребнісного походження відносно усяких психічно відображуваних явищ, або систему мотивів.

Мотиваційного значення, згідно С.Л. Рубінштейну, набуває кожне відображуване, тобто співвіднесене із системою уявлень (“образом світу”) суб'єкту явище. Тому можна вести мову про

всезагальну мотиваційну значимість явищ. Це означає, що предмети технічних та гуманітарних областей за умов їх включення в “образ світу” – інтегроване на основі досвіду уявлення про світ – можуть з однаковим успіхом стати потенційними спонуками навчальної діяльності.

Оскільки подібних відображуваних предметів у досвіді будь-якої людини незліченно багато, мотивація в роботах ряду авторів уявляється в якості множини взаємодіючих відносин, предмети яких можуть виступати у найрізноманітнішому ступені узагальнення та усвідомлення (починаючи з неусвідомлюваних відчуттів). Таким чином утворюється своєрідне мотиваційне поле. За умов об’єктивної можливості та суб’єктивної необхідності реалізації мотивів у відповідній діяльності останні набувають форми спонукання. Актуалізовані мотиви у своїй взаємодії виявляють різну спонукальну здатність, або силу мотиву, доступну для трансформуючих впливів викладача впродовж спільної навчальної діяльності. У психічному плані такі перетворення реалізуються в явищах сумачії (“полімотивація”), зсуву мотиву на нові предмети. В процесі викладання гуманітарних дисциплін врахування цих можливостей може виглядати, як очікуваний перехід від технічного змісту до гуманітарного, наприклад, від техніки до технічної естетики, від вивчення певних технологічних процесів до супідрядної екологічної проблематики тощо. Це у свою чергу це дозволить адекватніше складати та використовувати процесуально-методичний блок навчальної дисципліни.

Формування мотивації певної, наприклад, навчально-пізнавальної, діяльності таким чином, означатиме створення первинного малюнку і функціонального стану індивідуального мотиваційного поля. Останнє у своїй взаємоузгодженій з іншими індивідуальними полями частини складає з ними колективне, або групове мотиваційне поле. Ця обставина визначає можливість застосування групових форм впливу на мотивацію в процесі спеціально організованої навчальної діяльності. Мається на увазі формування мотивації спільної навчальної діяльності, тобто створення загального позитивного відношення (та інтересу) до виучуваного предмету, наприклад, гуманітарної дисципліни.

Основний принцип формування змістовного аспекту мотивації полягає в переносі мотиваційного значення з одних предме-

тів на іншій шляхом встановлення між ними когнітивних або чуттєвих зв'язків. Перший з механізмів, що називають мотиваційним опосередкуванням, розуміє під собою формування мотиваційного відношення до подій, явищ, що на основі вербальної інформації подаються через встановлення когнітивних зв'язків з такими ж досить абстрактними, але вже маючими власне мотиваційне значення, явищами. Другий – мотиваційне обумовлювання, передбачає розміщення двох явищ, одне з яких має певне мотиваційне значення і завдяки цьому викликає відповідну реакцію суб'єкта у вигляді емоцій, в єдиний простір генерованої таким чином емоції. Результатом є перемикання емоції на новий предмет і набування ним, як наслідок, власного мотиваційного значення.

Передбачається також поряд з першими двома і наявність деякого третього загального механізму, в якому агентами процесу вбачаються специфічні переживання, що визначаються не реальними властивостями предметів як пізнавальної структури, характеристиками відображення, а відношенням реальних властивостей речей до досвіду індивіда. Вони відрізняються і від емоцій, які виникають, як відомо, на основі, відношення реальних властивостей предметів до потреб. До таких переживань відносять, наприклад, сприйняття музики, відчуття міри реальності, відчуття просторового положення предметів, що оточують тощо.

Вочевидь, активне включення цього механізму в педагогічну практику обіцяє нові можливості до успіху.

Не менш важливим виявляється формування мотивації у динамічному аспекті. Так, онтогенетичний розвиток людської мотивації, пов'язаний із зменшенням детермінованості активних дій ситуаційними спонуканнями. Таке можливе лише за умов участі свідомо-вольової регуляції поведінки індивіда. Вона забезпечує впорядкування, змістовне та часове, особистого досвіду та внутрішніх потягів, введення в дії індивіда елемента вибіркової та доцільності. Як наслідок, дії все більш визначаються позаситуаційними цілями і задачами, які, в свою чергу, централізуються та утворюють ієрархію.

Адекватне формування свідомо-вольового рівня регуляції мотиваційних відношень досягається оптимальним поєднанням

двох основних видів виховних впливів. Перший з них полягає в актуалізації і закріпленні певних ситуаційних спонукань відповідними умовами діяльності і взаємостосунків “знизу-вгору”.

Другий – у засвоєнні формальних спонук, цілей, ідеалів, тобто зовнішньої мотивації (згори-донизу). Основним критерієм оптимізації виховних впливів вбачається енергетична забезпеченість спонукання, яка залежить від ступеню зв’язку абстрагованих одиниць індивідуального “образу світу” з чуттєво-сенсорним світом, від широти і ступеню когнітивного протиріччя у сфері емоційного перемикавання, від співвідношення збагачення та переносу певного емоційного відношення тощо. Педагогічні наслідки цих теоретичних положень, вочевидь, слід вбачати у введенні одного з найголовніших критеріїв визначення змісту навчального матеріалу на кожному окремому етапі процесу навчання та організації змістовного компоненту певної навчальної гуманітарної дисципліни.

Література

1. Возрастная и педагогическая психология. – М.: Просвещение, 1973. – С. 232.
2. Жалдак М.І., Морзе Н.В. Інформатика–7. Експериментальний навч. посібн. для учнів 7 кл. ЗОШ. – К.: ДіаСофт, 2000. – 208 с.
3. Формирование учебной деятельности студентов / Под ред. В.Я. Ляудис. – М.: Изд-во Моск. ун-та, 1989. – 240 с.
4. Когнитивная психология. Учебник для вузов / Под ред. В.Н. Дружинина, Д.В. Ушакова. – М.: ПЕР СЭ, 2002. – 480 с.
5. Загальна психологія / О. Скрипченко, Л. Долинська, З. Огороднійчук та ін. – К.: А.П.Н., 1999. – 463 с.
6. Рубинштейн С.Л. Основы общей психологии. – М., 1946. – 704 с.

ПРО ОСНОВНІ ПИТАННЯ КОМП'ЮТЕРНОГО КОНТРОЛЮ ЗНАНЬ, УМІНЬ І НАВИЧОК

П.А. Ротаєнко, Н.І. Самойленко, Л.П. Семко
м. Київ, Інститут педагогіки АПН України
lsemko@mail.ru

Невід'ємною складовою навчального процесу завжди були і є перевірка та контроль знань, умінь і навичок студентів та учнів. Саме із цим завданням пов'язувалось спочатку використання комп'ютерів у навчальних закладах. І цілком природно, що машинному контролю присвячено чимало розробок і публікацій як у вітчизняній, так і зарубіжній літературі. З появою сучасних комп'ютерно-орієнтованих навчальних систем питання дістало свій подальший розвиток як з точки зору його вивчення і дослідження, так і з точки зору безпосереднього впровадження в життя. Саме цьому питанню присвячується і дана стаття.

Контроль потрібний не тільки для діагностування та підготовки міцної і надійної опори, якою є попередні знання, але й для коригування знань, як тих, що набуваються в даний час, так і попередніх. Завдяки «блискавичному» реагуванню комп'ютера на дії (зокрема на запитання і відповіді) користувача, цей тип контролю, який іде паралельно з процесом засвоєння нових знань, набуває все більшого застосування у комп'ютерно-орієнтованих системах навчання.

Використання комп'ютера для перевірки і контролю знань, умінь і навичок, з одного боку, полегшує розв'язання цієї важливої дидактичної задачі, але, з іншого, вимагає додаткових досліджень і розробок відповідних методів і методик. Методів, що тісно пов'язані не тільки з конкретними методиками викладання різних предметів, їх загальними закономірностями, але й цілою низкою загальних положень життєдіяльності людського організму, особливо молодого. Тут не можна обійтися прямолінійним підходом: до відомих, скажімо, принципів і методів додати кілька положень, що впливають з особливостей і переваг комп'ютерних систем, чи дещо модифікувати старі.

Нині здійснюється перехід від навчання в умовах обмежено-

го доступу до інформації до навчання в умовах необмеженого доступу до інформації, відчувається необхідність методичних розробок курсів нового покоління. Вони можуть відрізнятись від традиційних значним збільшенням обсягу матеріалу (можливо й надлишкового» і такого, що безперервно поповнюється), модульними принципами побудови, і вимагають великої уваги до самостійної роботи, самонавчання та самоконтролю. Ми неодмінно тут стикаємося з проблемою контролю знань тих, хто навчається.

З дидактичної точки зору основними питаннями комп'ютерного контролю ми вважаємо такі.

1. Спільне у традиційних формах контролю та комп'ютерному, їх відмінності.
2. Загальні вимоги до комп'ютерного контролю.
3. Різні форми та прояви комп'ютерного контролю, їх внутрішня логіка.
4. Поєднання традиційних форм контролю з комп'ютерним. Використання результатів комп'ютерного контролю.

Зупинимось коротко на кожному з даних питань.

1. Будь-який контроль знань, умінь і навичок учнів ґрунтується на формулюванні та виконанні певних завдань, зокрема формулюванні запитань та відповідей до них. Система таких завдань повинна організовувати, стимулювати і спрямовувати діяльність учнів у потрібне русло. Запитання мусять бути короткими, чіткими і посильними для учнів, у невимушеній формі спонукати їх до розв'язання навчальних завдань. Запрошення до діалогу, що виражається у вигляді запитань та відповідей, або постановка і виконання тих чи інших завдань, послуг та відповідної реакції на них, – це своєрідний регулятор діяльності учнів, активний засіб керівництва нею. Необхідно слідкувати за тим, щоб цей діалог розвивався з наростаючою складністю завдань, мав евристичний заряд, проводився з урахуванням індивідуальних особливостей учнів та студентів.

Усе це безпосередньо стосується перевірки і контролю знань, умінь і навичок як традиційними методами, так і методами з використанням комп'ютера. Але виходити повністю з тих же критеріїв до запитань у комп'ютерних програмах, з яких ми виходимо у ході оцінювання запитань викладачем, не можна, тому що першим значно важче надати тієї гнучкості і різномані-

тності, що притаманні діалогу між людьми. Вони передбачають коротку, наперед визначену варіативну відповідь. З іншого боку, вони краще вивірені, продумані (без експромту) і не несуть у собі психологічного чи морального тиску на опитуваного.

Як у першому, так і в другому випадку запитання мусять не тільки узгоджуватися з навчальним матеріалом, вони повинні охоплювати відповідний програмний матеріал і виражати рівень вимог, передбачений державним стандартом.

Необхідно ретельно проаналізувати накопичений щодо означеного досвід, виділити у традиційних методах контролю саме характерне і перспективне з точки зору придатності їх для використання в нових умовах, визначити, що в них можна вважати інваріантним, а що варіативним, що потребує певної модернізації, а про що взагалі варто забути. Водночас доцільно врахувати чималі напрацювання з програмованого навчання і машинного контролю, про які було багато мови в 70-і роки минулого століття.

Оскільки, уся організація контролю та перевірки знань традиційними методами визначається викладачем на основі його педагогічного досвіду та інтуїції, то настільки при цьому різноманітними, неоднозначними і суб'єктивними є відношення його до відповідей учнів та студентів, критерії виставлення оцінок. Комп'ютерний контроль не допускає такої розмаїтості і тому не викликає сумнівів чи недовіри з боку опитуваних.

Завдяки вибору темпу, а то й часу та місця роботи за комп'ютером, комп'ютерний контроль сприяє кращій самоорганізації учнів та студентів, підвищує їхню відповідальність, економить час усіх учасників навчального процесу.

2. Питання діалогу між опитуваним і комп'ютером, зокрема аналіз різних формулювань запитань і відповідей, якими б методами він не проводився, – змістовими чи формальними, – виступає на перший план у вивченні та впровадженні комп'ютерного контролю.

Окремої розмови потребують питання побудови та якості тестів, що виражається у їх об'єктивності, валідності, несуперечливості, адаптивності та ін.

Для комп'ютерного контролю має значення логікограматична форма запитань (відповідей). Її дослідження здійс-

нюються формальними методами, а звідси уже впливає відповідність таких форм певним, спеціальним вимогам. При такому розгляді поняття запитання останнє має різні аспекти, серед яких найважливішим для нас є дидактичний. Семіотичне вивчення дидактичного аспекту може здійснюватись як з точки зору його синтаксичної побудови, так і з точки зору семантики, а можна його розглядати просто з точки зору педагогічної ефективності, тобто прагматичної точки зору. Завдання полягає у відшуканні методів підвищення та пониження ентропії запитання.

Велике значення мають вимоги до способів та методів подання запитань і відповідей, які часто можуть мати психологічну, а то й фізіологічну основу. Має значення, наприклад, належний зворотний зв'язок між опитуваним та комп'ютером, зокрема, швидкість і частота видачі комп'ютером запитань та реакцій на ті чи інші дії учня чи студента, можливість вибору різних варіантів програми, а, отже, і запитань у них, керування її виконанням, зокрема прямого переходу чи повернення до відповідних її ланок, фрагментів, і т.д. Сюди, звичайно, можна віднести вимоги до реакції комп'ютера на помилки, зокрема вимоги, що стосуються заохочення учня за правильні відповіді, фіксації уваги на неправильних відповідях, етичних норм і т.ін.

3. Комп'ютерний контроль, як і традиційний, може проводитись у різних формах, індивідуально та групами учнів.

Він може здійснюватись за допомогою спеціально для того призначених програмних засобів (комп'ютерних або тестуючих та опитувальних комп'ютерних навчальних систем); паралельно (майже одночасно) з навчанням у тренувальних, наставницьких, імітаційно-моделюючих і, особливо, інтелектуальних навчальних системах. В останніх комп'ютер не тільки вказує на допущені учнем чи студентом помилки, аналізує їх і підказує можливі напрямки його подальших дій, а й спонукає їх до самостійного розв'язування задачі методом, запропонованим системою або самими опитуваними. Водночас, якщо запропонований ними метод, за певними критеріями не програє тим, які система має в своїй стратегії, вона його включає у свій арсенал (стратегію) і в подальшому пропонує уже від себе.

Таким чином, тут уже йде мова не тільки про перевірку простих відповідей на ті чи інші запитання комп'ютера, а й про пе-

ревірку методу, стратегії (плану) розв'язування навчальної задачі. До речі, запитання досить часто в таких системах задає і сам опитуваний, тобто відбувається діалог між ними у формі спілкування, яке, в принципі, мало чим відрізняється від людського спілкування.

Найпростіша контролююча програма чи тест реалізуються за лінійною схемою, коли контроль ведеться за кількістю правильних і неправильних відповідей.

Такий же простий вигляд має умовно лінійна схема, за якою контроль реалізується поряд з навчанням і ведеться також за кількістю правильних і неправильних відповідей.

Дещо складніший вигляд має нелінійна схема навчання з простим зворотним зв'язком – поверненням учня до того матеріалу, на базі якого сформульовані запитання і відповідь.

Зворотний зв'язок базується тут на контролі відповідей, який ведеться з метою повного засвоєння матеріалу, що вивчається. Він може додатково вестися для оцінки швидкості такого засвоєння.

Зворотний зв'язок може бути доповнений порцією допоміжного матеріалу.

Схему можна і далі удосконалювати, її нелінійність при цьому буде зростати, зростатиме відповідно й ефективність побудованої на цій основі програми.

Завдання комп'ютерного тесту можна будувати так, щоб вони перебували у певній залежності. Вони можуть підбиратися і розташовуватися, так, що правильна відповідь на яке-небудь запитання означатиме також правильну відповідь на кожне із запитань, розташованих перед ним. Або ж навпаки. У першому випадку завдання розташовуються з наростанням складності, у другому – з її пониженням. Така структура тесту дає можливість, поперше, швидше здійснити перевірку, а, по-друге, задіяти просту формулу для оцінювання навчальних досягнень з відповідної теми.

Відомо, що “відкритого” типу завдання мають більшу ентропію ніж “закритого”, а значить у повнішій мірі дають можливість реалізувати індивідуальний, творчий підхід до формування відповіді, більше імпровізувати. Але вони вимагають глибшого опрацювання методики їх подання.

Про відносність закритої та відкритої форм тестових завдань і доцільність такої термінології можна говорити окремо. Очевидно, що першочерговим завданням має бути пошук критеріїв правильності відповідей у будь-якій формі (скажімо, у “довільнішій”) та побудові програмних оболонок, що мають використовувати такі критерії і бути зручними для наповнення їх різними за змістом завданнями.

4. Одного комп’ютерного контролю недостатньо для оцінювання знань, умінь і навичок, він у такому разі є лише складовою загальної перевірки, яка здійснюється викладачем. Будучи органічно вбудованим у комп’ютерні навчальні системи, такий контроль за своєю суттю наближається до контролю педагога. Але він не може бути повністю самодостатнім засобом, тому що реалізується з наперед заданими певними обмеженнями.

Очевидно, найкращий результат дає поєднання традиційних методів контролю з комп’ютерними. Звичайно, це не означає, що достатньо механічно поєднати ці методи. Щось із елементів того чи іншого традиційного контролю може просто відпасти, виявитись непотрібним, зайвим, або ж навіть перешкоджати ефективній роботі з відповідного завдання. У першу чергу це стосується рутинних елементів, або ж якихось повторень чи шаблонних процедур-завдань.

З іншого боку, педагогові завжди видніше, засвоєння яких саме частин і/чи аспектів розглядуваного або уже вивченого матеріалу потрібно саме в даний час перевірити. І які саме програми чи фрагменти програм підходять найкращим чином для цієї мети, чого в них не вистачає, як їх скомпонувати і т.д.

Принципове значення має, враховуючи багатобальну систему оцінювання знань, підсумковий комп’ютерний контроль знань. Тут важливо, по-перше, виробити формулу, за якою обчислюється остаточний бал, і, по-друге, вбудувати цю формулу (використати цей бал) у методику (нехай це також буде формула) остаточного оцінювання навчальних досягнень учня.

Тематичний та підсумковий контроль можна здійснювати шляхом проведення контрольних робіт з виконанням деякої їх частини на комп’ютері. Якщо контроль стосується теми (розділу) з великим обсягом матеріалу, його можна провести у два етапи: спочатку провести тестування із завданнями відкритого типу

(його можна проводити як традиційним способом, так і з використанням комп'ютера), а потім, відібравши найтипівіші відповіді на конкретні запитання, вбудувати їх у програму із завданнями закритого типу і скористатися цією програмою на другому етапі контролю.

Процес генерації контрольних варіантів повинен реалізуватися програмою з урахуванням побажань педагога, що знає конкретну навчальну ситуацію, його можливостей вибрати належний рівень комп'ютерної підтримки для кожного конкретного завдання. Викладач повинен чітко уявляти процедуру поповнення банку завдань.

Досягнення позитивних результатів в оцінюванні знань, умінь і навичок за допомогою комп'ютерного контролю – важлива дидактична задача. Вона вимагає серйозної, ретельної підготовки викладача. Для цього, зокрема, він мусить вибирати між різними комп'ютерними засобами, одного такого засобу часто буває замало.

МЕТОДОЛОГІЯ ЗАСТОСУВАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ У СТУДЕНТСЬКІЙ НАУКОВІЙ РОБОТІ З ІНОЗЕМНОЇ МОВИ

О.Ю. Рудик¹, К.М. Скиба¹, Т.О. Рудик¹, Н.В. Райко²

¹ м. Хмельницький, Хмельницький державний університет

² м. Хмельницький, Національна академія
Державної Прикордонної служби України
arudyk@rambler.ru

Розвиток інформаційних технологій призвів до того, що сучасні освітні технології використовують комп'ютерно-інформаційні технології, які привносять нове в традиційні методи викладання, перетворюючи й удосконалюючи основні поняття навчання [1].

Ефективність застосування інформаційних технологій при вивченні іноземної мови не підлягає сумніву. Учені відзначають, що чим більше систем сприйняття задіяно в навчанні, тим краще і міцніше засвоюється матеріал. Активне ж впровадження інформаційних технологій у навчальний процес збільшує дидактичні можливості, забезпечуючи наочність, аудіо і відеопідтримку і контроль, що в цілому сприяє підвищенню рівня викладання [2].

Педагогічне програмне забезпечення класифікують у такий спосіб [3]:

- інтерактивні мультимедійні електронні підручники-посібники;
- комп'ютерні навчальні програми;
- мультимедійні енциклопедії, електронні бібліотеки;
- тестово-контролюючі програми.

Інтерактивні мультимедійні електронні підручники-посібники – це повний методичний комплекс текстової, графічної, звукової та відеоінформації з декількох тем або за всім курсом. У них, разом з лекційним матеріалом, вправами для практичних занять і лабораторних робіт, тест-екзаменаційними матеріалами і навчальними відеороликами, наведені і доступні (freeware) програмні середовища для виконання домашніх завдань і лабораторних робіт.

Комп'ютерні навчальні програми – це комплекс текстової,

візуальної, графічної аудіоінформації. Вони дозволяють тренувати види мовної діяльності і сполучати їх у різних комбінаціях, допомагають усвідомити мовні явища, сформувані лінгвістичні здібності, створювати комунікативні ситуації, автоматизувати мовні дії, а також забезпечують реалізацію індивідуального підходу й інтенсифікацію самостійної роботи студента. За методичним призначенням виділяють наступні їхні типи [1]:

– граматичні, тобто програми, спрямовані на оволодіння граматичною системою мови (наприклад, категоріями роду, числа, відмінку, особи, виду тощо);

– лексичні (знаходження родинних слів, синонімів, антонімів, паронімів тощо) і лінгвокраїнознавчі програми, присвячені вивченню лінгвокраїнознавчих одиниць і зв'язаних з ними країнознавчих відомостей;

– програми, спрямовані на навчання таким видам мовної діяльності, як читання і письмо (програми з правопису, пунктуації), комунікативні, фонетичні.

Мультимедійні енциклопедії та електронні бібліотеки непродуктивні для використання на уроках, але важливі для самостійної підготовки студентів. Викладач, відсилаючи студентів, повинен сам знати ці матеріали, при цьому даючи чіткі вказівки для швидкого пошуку інформації.

Тестово-контролюючі програми – найпростіші та найефективніші і не займають багато часу на розробку. Різні види тестування залишаються у даний час найбільш розповсюдженою формою контролю при навчанні іноземним мовам. Різноманітні форми тестових завдань дозволяють здійснювати перевірку знань граматики, лексики, текстового матеріалу тощо. Розширення області застосування тестів визначається такими перевагами, як можливість роботи з великим обсягом матеріалу і швидке виявлення «пробілів» у знаннях.

Багато видів педагогічного програмного забезпечення розробляється самими викладачами (в основному тестово-контролюючі програми). Останнім часом для цього задіюються студенти, які під керівництвом викладача створюють сучасні дидактичні і методичні матеріали. Однак, такі розробки одиничні, не системного характеру. У Хмельницькому державному університеті (ХДУ) і Національній академії Державної Прикордонної

служби України (НАДПСУ) співробітництво викладачів, студентів і курсантів організовано у вигляді науково-дослідної роботи з виконанням комплексних курсових і дипломних проєктів. Для цього для роботи над проєктами залучені студенти-комп'ютершики (спеціальність «Інформаційні технології програмування») і лінгвісти (спеціальність «Переклад»). Спільні розробки формують у студентів і курсантів звичку до пізнавальної діяльності, прилучають їх до самостійного придбання знань, стають засобом підготовки до безперервної освіти.

Одним з можливих застосувань комп'ютерно-інформаційних технологій при вивченні іноземних мов є комп'ютерна лінгвістика, яка народилася на стику інформатики і мовознавства з потреби у спілкуванні з комп'ютером природною мовою. Лінгвістичні технології знаходять багато областей застосування, починаючи з нескладних програм перевірки орфографії, до більш витончених алгоритмів, використовуваних, наприклад, у системах машинного перекладу (СМП).

Сучасний стан машинного перекладу дозволяє одержувати відносно коректний переклад з більшості мов. І хоча цілком автоматичний високоякісний переклад неможливий, існує програмне забезпечення, яке полегшує сам процес перекладу [4]. У результаті настроювання на предметну область та інтеграції з іншими програмами обробки документів машинний переклад дозволяє автоматизувати одержання перекладу.

В Україні розробкою СМП займається компанія Trident Software (trident@kagi.com), яка представлена перекладачем Pragma 2.0. І хоча якість перекладів даною системою досить висока [5], однак існують наступні проблеми, з якими вона зіштовхується: неоднозначність, омоформія на рівні морфологічного розбору, переклад слів зі складу фразеологізмів, вибір тематичної ознаки, словотворення і багато інших. У даний час студентсько-курсантський колектив займається питаннями дієслівної форми з закінченням *-ed*, *ing*-вої форми дієслова, створенням польсько-російсько-українського модуля для СМП Pragma 2.0, розробкою тестово-контролюючих програм польсько-українського спрямування.

Співробітництво ХДУ і НАГПСУ з компанією Trident Software організовано як у виді тестування нових версій програ-

ми-перекладача Pragma з наступним спільним усуненням характерних помилок, так і зі створення нових напрямків перекладу. Пропоновані компанією евристичні типи завдань, що вимагають експерименту й елементів дослідження, сприяють самостійному розвитку у студентів і курсантів мовних і професійних навичок.

Література

1. <http://ito.bitpro.ru/2003>
2. http://www.dlab.kiev.ua/conf_2003/reports_u.html
3. <http://www.nkkep.dp.ua/Methodi/IT-Conferenc-2.htm>
4. Рудик О.Ю., Скиба К.М., Сторожук І.Ф. Засоби машинного перекладу для роботи в Internet / Вісник ТУП, Ч. 1. Технічні науки. – Хмельницький: ТУП, 2003. – С. 216–219.
5. Рудик О.Ю., Скиба К.М., Сторожук І.Ф. Системи машинного перекладу (українсько- та російсько-англійська підсистеми). / Зб. наук. праць №21, Ч. 2 (спец. випуск). – Хмельницький: НАПВУ, 2002. – С. 346–353.

ФОРМИРОВАНИЕ ЭРГОНОМИЧЕСКИХ ЗНАНИЙ У СТУДЕНТОВ ИНЖЕНЕРНО-ПЕДАГОГИЧЕСКИХ СПЕЦИАЛЬНОСТЕЙ

Г.И. Сажко, А.Н. Майданович

г. Харьков, Украинская инженерно-педагогическая академия
ikt@uipa.kharkov.org

Эргономика – научная дисциплина, комплексно изучающая трудовую деятельность человека в системах «человек – техника – среда» (СЧТС) с целью обеспечения ее эффективности, безопасности и комфорта. Аналогичную область знаний в США называют «человеческими факторами» (human factors) [1, 2].

Эргономика относится к группе наук неклассического типа, сочетающих в себе черты научной дисциплины и средства практической деятельности [2]. Она взаимосвязана со всеми науками, предметом изучения которых является трудовая деятельность человека: инженерной психологией, психологией, физиологией, гигиеной, социологией труда, безопасностью труда и др. Она возникла на «стыках» наук о человеке и призвана устранить нарушения единства оптимизации всех компонентов деятельности, обусловленных множеством дисциплинарных подходов к созданию СЧТС.

Значение эргономики для народного хозяйства состоит в усилении социальной ориентации технического прогресса, в повышении качества СЧТС, сокращении сроков подготовки человека к обслуживанию этих систем, сведению к минимуму количества аварий и катастроф по причинам несовершенной организации взаимодействия человека с техникой, снижении напряженности и повышении привлекательности труда [3].

В целом, эргономика способствует разработке оптимального рабочего места, проектированию компьютерных средств и программного обеспечения, товаров народного потребления и сельскохозяйственных машин и т.д. Это подчеркивает как важность, так и успех эргономики, область применения которой стремительно расширяется, охватывая фактически все аспекты человеческой деятельности на работе, дома и во время досуга.

На основании анализа нормативных документов, научно-

технической литературы, международного опыта развития эргономики, современных концепций эргономического образования нами была поставлена цель выявить и обосновать целесообразность эргономической подготовки студентов инженерно-педагогических специальностей компьютерных специализаций.

Среди нормативных документов наиболее важным с позиции рассматриваемой темы является Постановление №37 Кабинета Министров Украины от 20 января 1997 г. *«Про першочергові заходи щодо розвитку національної системи дизайну та ергономіки і впровадження їх досягнень у промисловому комплексі, об'єктах житлової, виробничої і соціально-культурної сфер»*. Учитывая значение дизайна и эргономики в улучшении качества промышленной продукции и ее конкурентоспособности на мировом рынке, в создании современных условий труда и жизнедеятельности людей, в п. 5 этого постановления Министерству образования, Всеукраинской эргономической ассоциации и другим министерствам предписывалось в первом полугодии 1997 г. *«розробити пропозиції щодо формування системи підготовки та перепідготовки фахівців (у тому числі вищої кваліфікації) у галузі дизайну та ергономіки; затвердити перелік базових навчальних закладів з підготовки та перепідготовки фахівців в галузі дизайну та ергономіки»*.

В соответствии с этим постановлением Всеукраинская эргономическая ассоциация разработала концепцию эргономического образования в Украине [4]. Исходя из предмета и задач эргономики, были выделены следующие группы специалистов по эргономике:

1. А. По степени охвата задач эргономического обеспечения:
 - A1. Эргономисты-системотехники
 - A2. Эргономисты, решающие частные задачи эргономического обеспечения
 2. Б. По месту в процессе создания СЧТС (по отраслям)
 - B1. Эргономисты-исследователи
 - B2. Эргономисты-проектировщики
 - B3. Эргономисты-дизайнеры
 - B4. Эргономисты-эксплуатационники.
- В концепции была предложена схема сквозной подготовки

специалистов по эргономике. Из этой схемы видно, что ещё 5 лет назад для учащихся всех учебных заведений рекомендовался учебный курс «эргономика», и что в УИПА предполагалась подготовка специалистов по эргономике для преподавательской работы в учебных заведениях. Эргономическая подготовка студентов инженерно-педагогических специальностей по содержанию и по объёму ближе всего к подготовке эргономистов группы Б4 – эргономистов-эксплуатационников.

В настоящее время в мире издается более десятка специализированных журналов по эргономике, ежегодно проводятся национальные и международные конференции и симпозиумы. Среди них следует выделить ежегодные международные семинары для преподавателей эргономики, в частности, 16-й семинар «Сертификация и аккредитация обучения эргономике, охране труда и безопасной работе» в Лешно, Польша (1999 г.) и 14-й Конгресс Международной эргономической ассоциации – IEA в Сан-Диего, США (2000 г.) [5]. Последний конгресс стал значительной вехой в истории эргономики. Конгресс не только организационно указал на рост интереса к этой дисциплине во всем мире (количество федеративных членов IEA возросло за 3 года с 29 до 38), но и внес значительный вклад в содержательной области: зафиксировал в новом определении признание эргономики как самостоятельной научной дисциплины; констатировал появление новых приоритетов дисциплины, связанных с кардинальным изменением характера трудовой деятельности человека – переход от физического труда к преимущественно умственному; признал когнитивные и психофизиологические подходы к решению проблем безопасности и эффективности труда наиболее быстро развивающимися; подтвердил рост важности учета влияния окружающей среды, в том числе организационной, при проектировании и эксплуатации изделий и процессов.

Анализ наименований и аннотаций 1513 диссертаций, выполненных в Украине за период 1993–2001 г.г., а также объявлений о защите кандидатских диссертаций, опубликованных в журнале «Науковий світ» за 2003 год, показал, что компьютерная и эргономическая подготовка студентов инженерно-педагогических специальностей компьютерных специализаций является новым объектом педагогических исследований.

Выводы

1. В современном мире эргономика становится одной из движущих сил экономического и социального прогресса. Значение эргономики для народного хозяйства состоит в усилении социальной ориентации технического прогресса, в повышении качества СЧТС, сокращении сроков подготовки человека к обслуживанию этих систем, сведению к минимуму количества аварий и катастроф по причинам несовершенной организации взаимодействия человека с техникой, снижении напряженности и повышении привлекательности труда.

2. В связи с кардинальным изменением характера трудовой деятельности человека – повсеместным переходом от физического труда к преимущественно умственному – у эргономики появились новые приоритеты: когнитивные процессы, образование, управление.

3. В Украине есть нормативная база и социальные предпосылки для развития и распространения эргономического образования.

4. Мировой опыт говорит о необходимости эргономической подготовки будущих инженеров – педагогов компьютерных дисциплин, которые будут готовить рабочие кадры компьютеризированного производства.

5. Эргономическая подготовка будущих инженеров – педагогов компьютерных дисциплин являются новым объектом и предметом исследования в педагогике.

Литература

1. Человеческий фактор. В 6-ти томах. Т. 1 Эргономика – комплексная научно-техническая дисциплина: Пер. с англ./ Ж. Кристенсен, Д. Мейстер, П. Фоули и др. – М.: Мир, 1991. – 599 с.
2. Эргономика в вопросах и ответах: Материалы понятийной базы эргономики / Г.М. Зараковский, В.М. Мунипов, П.Я. Шлаен / Под ред. Е.Н. Куличкова и А.А. Польского. – Тверь: Эргоцентр, 1993. – 68 с.
3. ДСТУ 3899-99. Дизайн та ергономіка. Терміни та визначення. – Київ: Держстандарт України, 1999. – 33 с.

4. Ашероv А.Т., Лавров Е.А., Людвичек К.В. Концепция эргономического образования в Украине // Новый коллегийум (Науковий інформаційний журнал. Проблеми вищої освіти). – 1999. – №1. – С. 14-17.
5. Proceedings of the IEA 2000 \ HFES 2000 Congress. July 29 - August 4, 2000. - San Diego, California USA – (CD)

ОСОБЕННОСТИ ИЗУЧЕНИЯ АППАРАТНЫХ СРЕДСТВ ПЕРСОНАЛЬНЫХ КОМПЬЮТЕРОВ

В.Д. Сквирский, Н.А. Семенов
г. Луганск, Луганский национальный педагогический
университет имени Тараса Шевченко
svd_4646@mail.ru

При подготовке студентов по специальности «Информатика» особое место занимает курс «Элементы и узлы персональных компьютеров».

Современные персональные компьютеры (ПК) имеют высокую степень интеграции элементов и узлов, что затрудняет их поблочное изучение традиционными физическими методами с помощью входных и выходных сигналов. Многие вопросы, позволяющие анализировать функционирование аппаратных средств ПК, могут быть решены с помощью специальных программных средств, к которым можно отнести всевозможные тестирующие программы, а также программы-отладчики.

Существующие программы-отладчики решают практически одинаковые задачи и отличаются, в основном, видом интерфейса и сервисными функциями. Отладчик *DEBUG* [1] является наиболее простым (всего один файл – *debug.exe*), а главное – он всегда присутствует в наборе утилит широко применяемых операционных систем (*MS-DOS*, *Windows* и др.), что и предопределило его использование в учебных целях.

С помощью отладчика *DEBUG* можно решать широкий круг задач, например, следующие:

- изучение текущего содержимого оперативной памяти, регистров процессора;
- редактирование отдельных секторов на флоппи-дисках и на винчестере;
- загрузка и дизассемблирование *.com* и *.exe*-файлов;
- разработка и отладка программ на языке ассемблера;
- загрузка, модификация, вывод на экран содержимого произвольного файла;
- тестирование периферийного оборудования, для работы с портами ввода/вывода напрямую (в диалоговом режиме);

- изучение системы команд процессора, прерываний *BIOS* и *MS-DOS*.

На кафедре экономической информатики Луганского национального педагогического университета имени Тараса Шевченко нами разработаны методические материалы и проводятся лабораторно-практические работы по применению системного отладчика *DEBUG* для изучения ряда узлов ПК.

Лабораторно-практический цикл представлен в виде следующих 8 работ.

Работа 1. Работа с файлами в среде *DEBUG* (часть 1).

Здесь исследуются способы загрузки содержимого файлов в машинных кодах с различных носителей в оперативную память компьютера и просмотра этого содержимого.

Работа 2. Работа с файлами в среде *DEBUG* (часть 2).

Изучаются возможности изменения содержимого файлов в ОЗУ, копированию в другую область памяти, связь параметров файла с состоянием регистров процессора.

Работа 3. Исследование системной области *BIOS* в оперативной памяти (часть 1).

Исследуются области памяти, в которых размещены некоторые данные операционных систем.

Работа 4. Исследование системной области *BIOS* в оперативной памяти (часть 2).

Изучается область векторов прерываний и организация прерываний. Анализ основных признаков программы *POST*.

Работа 5. Работа с программами в среде *DEBUG*.

Изучаются возможности размещения в памяти, редактирования и запуска на исполнение программ на языке ассемблера и в машинных кодах.

Работа 6. Работа с видеопамятью.

Предлагается получить навыки в работе с текстовым дисплейным буфером памяти.

Работа 7. Работа с портами ввода/вывода.

Изучаются приемы работы с внешними устройствами (дискетой, динамиком, *CMOS*-памятью) через порты ввода/вывода.

Работа 8. Работа с дисками.

Предложены задания по «жесткой» загрузке в ОЗУ инфор-

мации с гибких и жестких дисков, записи на них иной информации. Рассмотрена работа с информацией, расположенной в *BOOT*-секторе диска.

Для поддержки указанного курса создано учебное пособие [2], в котором содержатся теоретические сведения о системе команд и приемы работы с отладчиком.

Ниже в качестве примера приведено описание порядка выполнения одной из лабораторных работ цикла.

ЛАБОРАТОРНАЯ РАБОТА № 7

РАБОТА С ПОРТАМИ ВВОДА/ВЫВОДА

Цель работы: научиться работать с внешними устройствами через порты ввода/вывода.

Порядок выполнения работы

При подготовке к работе необходимо:

- 1) просмотреть задание, после чего найти и изучить соответствующие теоретические положения в разделе 2 учебного пособия [2], а также в материалах курса «Элементы и узлы персональных компьютеров»;
- 2) ответить на контрольные вопросы;
- 3) в тетради для лабораторных работ заготовить отчет, в который предварительно занести название и цель работы, требуемые для выполнения теоретические данные, команды и последовательность действий в соответствии с заданием.

В лаборатории:

- 1) по указанию преподавателя включить компьютер и загрузить операционную систему *Windows*;
- 2) открыть окно <Панель управления> (<*Control Panel*>), а в нем – содержимое пиктограммы <Система> (<*System*>). В окне <Система> на вкладке <Устройства> (<*Device Manager*>) установить курсор на пиктограмме <Компьютер> (<*Computer*>) и щелкнуть кнопку <Свойства> (<*Properties*>) и в открывшемся окне на вкладке <Просмотр ресурсов> (<*View Resources*>) выбрать опцию <Ввод/вывод> (<*Input/output (I/O)*>);
- 3) записать в тетрадь принадлежность к системным ресурсам портов с адресами 40H, 61H, 70H, 71H, 3F2H;

- 4) выйти в режим *MS-DOS Prompt* и вызвать среду *DEBUG*;
- 5) набрать и выполнить команду ввода из порта с адресом 40H. Записать содержимое порта;
- 6) повторить несколько раз выполнение команды по п. 5. Объяснить полученные результаты;
- 7) выполнить по очереди две команды:
 - О 3F2 10
 - О 3F2 00
- 8) Отметить, что происходит с дисководом А: для флоппи-дисков после выполнения каждой команды;
- 9) включить звуковой сигнал путем обращения к порту 061H. Для этого сначала прочитать содержимое порта и записать полученное число, определяющее выключенное состояние динамика. Далее добавить к этому числу константу 3 и вывести в порт 061H число, полученное в результате сложения;
- 10) выключить звуковой сигнал командой вывода;
- 11) прочесть некоторую информацию из *CMOS*-памяти.

Указания. Программно обмен с *CMOS*-памятью осуществляется через адресный порт записи 70H и порт чтения/записи 71H. Сначала в порт 70H заносится адрес нужного байта, а затем через порт 71H осуществляется доступ к этому байту *CMOS*-памяти. В таблице 1 приведено описание некоторых байтов этой памяти.

Таблица 1

Адрес байта	Функции
00	Текущее значение секунд показаний часов реального времени
02	Текущее значение минут
04	Текущее значение часов
06	Текущий день недели
07	День месяца
08	Номер текущего месяца
09	Год (две последние цифры)

Выполнить команды:

О 70 02 ; адресация к байту 02

I 71 ; чтение текущего значения минут

- 12) Сверить значение с показанием часов ПК (по команде *time*

- либо по показаниям табло в панели задач *Windows*).
- 13) Обратиться по очереди к другим байтам *CMOS*-памяти и вывести их значения на экран. Сравнить данные с показаниями часов реального времени.
- 14) выйти из программы-отладчика в среду *Windows*;

В Указаниях к п. 11 справедливо говорится о возможностях чтения и записи в ячейки *CMOS*-памяти. В задании предусмотрен только процесс чтения информации. Попытки изменить содержимое *CMOS* ни к чему не приведут, так как для изменения необходима дополнительная (разрешающая) команда.

В описании опущены сведения об оборудовании и программном обеспечении, дополнительные указания, контрольные вопросы и другие пункты.

Следует сказать, что в условиях отсутствия специальных обучающих стендов для физических исследований работы аппаратных средств компьютеров, предложенная методика анализа работы узлов ПК посредством программных средств нижнего уровня показала себя достаточно эффективной.

По результатам выполнения лабораторных работ и опроса студентов можно отметить повышенный интерес студентов к данному курсу, и даже тех, кто относит себя к «чистым» программистам. Это объясняется получением навыков в решении задач, связанных с ограничением ресурсов при программировании, рациональным использованием драйверов, системы прерываний и др.

ЛИТЕРАТУРА

1. Левкин Г.Н., Левкина В.Е. Введение в схемотехнику ПЭВМ РС/АТ. – М.: Изд-во МПИ, 1991. – 96 с.
2. Сквирский В.Д. Внутренняя память персональных компьютеров. Учебное пособие. – Луганск: Альма матер, 2001. – 118 с.

РЕШЕНИЕ ТРАНСЦЕНДЕНТНЫХ УРАВНЕНИЙ МЕТОДОМ НЬЮТОНА ПРИ ИЗУЧЕНИИ ТАБЛИЧНОГО ПРОЦЕССОРА EXCEL

Е.И. Скворцова

г. Керчь, Керченский морской технологический институт
ginaskvorz@mail.ru

Современные информационные технологии представляют студенту очень широкий спектр возможностей. Это и быстрая передача огромных объемов информации, и возможность решения сложнейших вычислительных задач, построение адекватных моделей, проведения прямых исследований на виртуальном оборудовании и т.п. [1].

В современных условиях общего образования в институте, где обучают морским специальностям, когда объема аудиторной нагрузки явно не хватает для полноценного раскрытия всех тем рабочей программы, особая роль отводится межпредметным связям и повышению качества подготовки морских специалистов.

А.И. Ефремова пишет, что межпредметные связи – это методическая система, вне которой невозможно достичь высокого уровня качества обучения [2].

Мы можем отметить, что межпредметные связи – это методическая система, в которой используются знания, полученные ранее по другим предметам, для изучения данной дисциплины, и которая направлена на достижение высокого уровня качества обучения.

Проблема развития межпредметных связей уже давно существует перед преподавателями и методистами: научить будущих морских специалистов использовать в своей деятельности знания, приобретенные при изучении не одного предмета, а полученные в целостной системе обучения.

Осуществление межпредметных связей предлагают современные исследователи в области методики физики и математики В.М. Барановский, В.Ф. Дмитриева, О.А. Марченко, А.С. Нисимчук, О.С. Падалка, С.Н. Прийма, Е.Х. Свергун, А.В. Сергеев, И.А. Смолук, О.М. Царенко, А.Т. Шпак и другие специалисты в области методики, с помощью объединения теоретических и

практических курсов общеобразовательных и специальных предметов как в средней, так и высшей школе.

Целью данной статьи является показать возможность осуществления межпредметной связи двух общеобразовательных дисциплин, которые изучаются в морском институте – математики и информатики на примере решения трансцендентного уравнения методом Ньютона при изучении работы в табличном процессоре Excel.

Лабораторная работа

«Решение трансцендентных уравнений методом Ньютона»

Задание.

Найти корни алгебраического уравнения $F(x)=0$ на промежутке $[x_n, x_k]$ методом Ньютона и модифицированным методом Ньютона. Предварительное отделение корней уравнения произвести путем табулирования функции $F(x)=0$ на заданном промежутке с шагом, равным $1/20$ от длины промежутка.

В отчете представить:

- результат отделения корней;
- значение корня уравнения и количество итераций для его достижения для значений точности $\varepsilon=0.00, 0.001, 0.0001$, полученные методом Ньютона и модифицированным методом Ньютона.

Порядок выполнения работы:

Создать электронную таблицу для определения одного из корней уравнения $2\ln x - 0,5x + 1 = 0$ на промежутке $[1, 20]$.

Этап 1. Отделение корней

Создадим таблицу значений функции $F(x)=2\ln x - 0,5x + 1$ на промежутке $[1, 20]$ в виде последовательного значения аргумента $x=x_n+i\cdot h$ ($i=0, 1, \dots, 20$; $h=(20-1)/20=0.95$) и соответствующих значений функции $F(x)$.

Сначала введем заголовки таблицы в соответствующие строки:

A1← 'Лабораторная работа

A2← 'Отделение корней уравнения $2\ln x - 0,5x + 1 = 0$

Введем исходные данные для построения таблицы:

A4← 'Xнач= ;

B4← 1;

C4← 'Xкон= ;

D4← 20 ;

E4← ' h= ;

F4← =(D4-B4)/20

Создадим начальные клетки таблицы значений x и $F(x)$:

A5← 'X ;

B5← 'F ;

A6← =B4;

B6← =2*ln(A6)-0.5*A6+1;

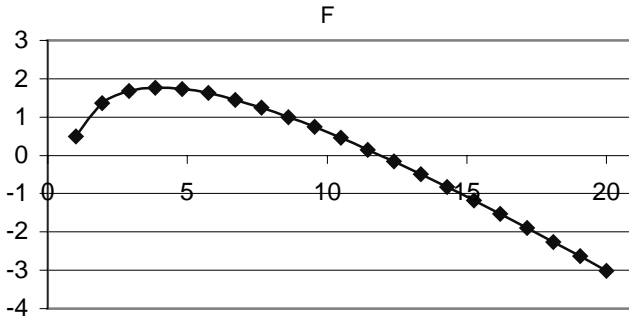
A7← =A6+\$F\$4

Скопируем клетку **A7** в диапазон клеток **A8:A26** и клетку **B6** в диапазон клеток **B7:B26**.

	A	B	C	D	E	F
1	Лабораторная работа № 1					
2	Отделение корней уравнения $2\ln x - 0,5x + 1 = 0$					
3						
4	Xнач= 1		Xкон= 20			h= 0,95
5	X	F				
6	1	0,5				
7	1,95	1,36066				
8	2,9	1,67942				
9	3,85	1,77115				
10	4,8	1,73723				
11	5,75	1,6234				
12	6,7	1,45422				
13	7,65	1,24441				
14	8,6	1,00352				
15	9,55	0,73808				
16	10,5	0,45275				
17	11,45	0,15098				
18	12,4	-0,1646				
19	13,35	-0,492				
20	14,3	-0,8295				
21	15,25	-1,1758				
22	16,2	-1,53				
23	17,15	-1,891				
24	18,1	-2,2582				
25	19,05	-2,6309				
26	20	-3,0085				

Построим график функции $F(x)$ по значениям, полученным в таблице.

В результате получим следующую картинку:



Наблюдая полученные таблицы и график, можно сделать следующие выводы:

- а) на отрезке $[1, 20]$ имеется один корень;
- б) корень расположен в интервале $(11.45, 12.4)$.

Этап 2. Уточнение корней

Проведем уточнение корней уравнения $2\ln x - 0.5x + 1 = 0$ на интервале $11.45 < x < 12.4$, воспользовавшись алгоритмом метода Ньютона.

1) Введем заголовки новой таблицы:

D5 ← 'Уточнение корней методом Ньютона

D6 ← 'Первая производная в точке X_0

G6 ← ' $f'(x_0) =$;

H6 ← $2/E8 - 0.5$;

E7 ← 'X ;

F7 ← ' $F(x)$;

G7 ← $|X_k - X_{k-1}|$

H7 ← 'Итерации

2) Введем начальные значения отрезка, на котором находится корень, значение функции $F(x)$ и интервала неопределенности $|X_k - X_{k-1}|$:

E8 ← 11.45

F8 ← $= 2 * \text{LN}(E8) - 0.5 * E8 + 1$

G9 ← $= \text{ABS}(E9 - E8)$

3) Скопируем диапазон **E8:F8** в диапазон **E18:F18**.

4) Скопируем **G9** в диапазон **G10:G18**.

В результате выполнения этих действий на экране будет представлено следующее:

	A	B	C	D	E	F	G	H	I
1	Лабораторная работа № 1								
2	Отделение корня уравнения $2\sin x - 0,5x - 1 = 0$								
3									
4	Хзад=	1	Хкон=	20	h= 0,95				
5	X	F	Уточнение корней модифицированным методом Ньютона						
6	1	0,5	Первая производная в точке X_0 $f'(x_0)$ $-0,3253$						
7	1,05	1,36068	X	f(x)	$f'(x_0)$	Итерации			
8	2,9	1,67942		11,45	0,150979		1		
9	3,85	1,77115		11,014085	-0,0016	0,464065	2		
10	4,8	1,73723		11,91	3,33E-05	0,004917	3		
11	5,75	1,6234		11,90927	-6,89E-07	0,0001023	4		
12	6,7	1,46422		11,909267	1,43E-08	0,0000021	5		
13	7,65	1,24441		11,909268	-2,95E-10	4,39E-06	6		
14	8,6	1,00352		11,909268	6,12E-12	9,08E-10	7		
15	9,55	0,73808		11,909268	-1,27E-13	1,88E-11	8		
16	10,5	0,46275		11,909268	2,66E-15	3,91E-13	9		
17	11,45	0,15098		11,909268	0	8,88E-15	10		
18	12,4	-0,1646							
19	13,35	-0,402							
20	14,3	-0,8295							
21	15,25	-1,1758							
22	16,2	-1,53							
23	17,15	-1,891							
24	18,1	-2,2582							
25	19,05	-2,6309							
26	20	-3,0085							

Аналогично проводим уточнение корней модифицированным методом Ньютона, который отличается от метода Ньютона тем, что первая производная от $F(x)$ вычисляется в каждой точке приближения к корню уравнения. Таким образом, у нас будет один дополнительный столбец, в котором мы будем вычислять первую производную $F(x)$ в каждой точке приближения к корню уравнения.

После выполнения этих действий в клетках столбца E и J будут находиться последовательные приближения к точному значению корня с интервалом неопределенности, который отражается в столбце G и M. Величина интервала неопределенности на каждом шаге уменьшается. Как только интервал станет меньше заданной точности ε , то мы можем считать задачу выполненной.

Таким образом, из полученного результата видно, что метод Ньютона обладает более высокой скоростью сходимости, чем модифицированный метод Ньютона.

Если же сравнить время, затраченное на решение трансцендентного уравнения численными методами вручную и средствами табличного процессора Excel, то выигрыш во времени оказывается значительным.

Литература

1. Падалка О.С., Нисимчук А.С., Смолюк И.А., Шпак А.Т. Педагогические технологии: Учебн. пособие. – Киев: Изд-во “Украинская энциклопедия” им. М.П. Бажана, 1995. – С. 65-78.
2. Ефремова А.И. Межпредметные связи физики и математики 9-11 классах средней общеобразовательной школы: Дис. ... канд. пед. наук / Южноукраинский гос. пед. ун-т им К.Д. Ушинского. – Одесса, 2001. – 223 с.

ИНФОРМАТИКА ДЛЯ ИНЖЕНЕРНЫХ СПЕЦИАЛЬНОСТЕЙ ТЕХНИЧЕСКИХ УНИВЕРСИТЕТОВ

А.Ю. Соколов^{1α}, Г.Н. Жолткевич^{2β}, И.Т. Зарецкая^{2γ}, О.В. Яровая¹

¹ г. Харьков, Национальный аэрокосмический университет
«ХАИ» им. Н.Е. Жуковского

² г. Харьков, Харьковский национальный университет
им. В.Н. Каразина

^α asokolov@xai.edu.ua

^β gregory.n.zholtkevych@univer.kharkov.ua

^γ irina.t.zaretskaya@univer.kharkov.ua

Трудно назвать другую сферу человеческой деятельности, которая развивалась бы столь стремительно и порождала бы такое разнообразие проблем, как информатизация и компьютеризация общества. История развития информационных технологий характеризуется быстрым изменением концептуальных направлений, технических средств, методов и сфер применения.

Еще десять лет назад бесспорным казался лозунг «Программирование – вторая грамотность», под которым подразумевалось умение каждого инженера создавать алгоритмы и программировать в своей предметной области на языках Fortran, Pascal и т.п. В современных условиях весьма актуальным для большинства людей стало не столько программирование (в старом смысле слова), сколько умение пользоваться промышленными информационными технологиями. Проникновение компьютеров во все сферы жизни общества убеждает в том, что культура общения с компьютером становится частью общей культуры человека – термины «Word», «Excel», «MathCAD» стали обыденными.

Возрастание роли информационных технологий в различных сферах инженерной деятельности стало сегодня очевидным фактом. Компьютерное моделирование, использование систем автоматизации проектных работ, применение экспертных систем при принятии технических решений стало повседневной инженерной практикой. Эффективное владение этими технологиями стало одной из основных квалификационных характеристик современного инженера. Следует подчеркнуть, что целью компьютерной

подготовки такого специалиста является формирование у студента понимания возможностей применения компьютера в своей профессиональной деятельности, а не навыков использования компьютера вообще. Такое понимание должно базироваться на знаниях взаимозависимостей между фундаментальными математическими и естественно-научными курсами, общетехническими и специальными инженерными дисциплинами. Именно информационные технологии играют здесь интегрирующую роль. В связи с этим, курс информатики, гармонизованный с фундаментальной и специальной подготовкой инженера, должен начинаться с первого года обучения в высших технических учебных заведениях и продолжаться весь период бакалаврской подготовки.

К сожалению, существующая учебная литература по информатике ориентирована в первую очередь либо на подготовку специалистов в области информационных технологий, либо на формирование первичных навыков работы с компьютером. Таким образом, можно констатировать дефицит учебников, которые излагают проблемы информатики не сами по себе, а в контексте решения важных прикладных инженерных задач. Предлагаемый учебник представляет собой попытку авторов ликвидировать указанный пробел в учебной литературе.

Учитывая уровень и характер первоначальной (школьной) подготовки студентов (авторы рассчитывают на знакомство читателя с курсом школьной информатики в объеме [1–3]), а также опыт преподавания информатики в ведущих технических университетах мира, авторы пришли к следующей структуре учебника.

Первая глава посвящена общему обзору методологии создания программ, базовым алгоритмам и структурам данных, а также методам их реализации на основании структурной парадигмы программирования.

В нашу задачу входило доступным для студентов общеинженерных специальностей языком описать полный цикл разработки программы, включающий:

- построение модели;
- разработка алгоритма;
- написание кода программы и его отладка;

- документирование программы;
- ее тестирование.

В этой главе изложены основы структурного подхода к проектированию программного обеспечения, включая базовые структуры и методы описания алгоритмов, методы и шаблоны структурного проектирования, этапы разработки и порядок документирования программ. Рассмотрены также основные структуры данных и методы их реализации, а также ставшие классическими алгоритмы сортировки, поиска, кодирования и шифрования информации. Изложение сопровождается тщательно подобранными примерами стандартных алгоритмов и структур данных, входящих в базовые курсы информационной подготовки как в нашей стране, так и за рубежом.

Вторая глава посвящена перспективной современной методологии разработки программных систем на основе объектно-ориентированного подхода (ООП). В ней определены и проиллюстрированы концептуальные понятия ООП:

- объекты, классы, поля и методы;
- инкапсуляция, наследование, полиморфизм;
- итерационная технология «анализ – проектирование – реализация – ввод в эксплуатацию»;
- базовые шаблоны объектного проектирования.

Изложение сложной концепции ООП поясняется на двух примерах моделирования технических систем: простой оптической системы и часов Гюйгенса. На этих примерах рассмотрен полный цикл разработки программного продукта: от построения математических моделей физических процессов, лежащих в основе функционирования соответствующих технических систем до отлаженного кода и документации. С целью обеспечения полноты указанного цикла разработки программной системы изложены:

- основы объектно-ориентированного программирования на языке Object Pascal;
- принципы программирования в операционных средах передачи сообщений;
- основы программирования в среде Delphi.

Третья глава носит теоретический характер. Ее задача, с точки зрения авторов, состоит в формировании у студента цело-

стного представления о математических методах, лежащих в основе подавляющего большинства программных моделей технических систем, а также взаимосвязей между этими методами. Подбор методов осуществлялся с учетом их практической ценности для современной инженерной деятельности. Особенностью изложения материала является отход от формулировки и доказательства теорем. Изложение в этом стиле дублировало бы стандартные курсы математической подготовки инженера. Вместо этого авторы приводят формулировки основных результатов, делают акцент на алгоритмических проблемах, которые эти результаты позволяют решить, демонстрируют возможности их применения при разработке алгоритмов и программ. Изложение сопровождается примерами использования основных результатов и полученных на их основе алгоритмов в информатике. Основные параграфы этой главы посвящены:

- элементам дискретной математики, включая вопросы связанные с алгеброй отношений;
- проблемам обеспечения точности арифметических вычислений;
- вопросам кодирования информации;
- алгоритмам вычислительной геометрии;
- базовым численным методам решения линейных и нелинейных алгебраических уравнений;
- численным алгоритмам получения решений дифференциальных уравнений;
- методам решения задач линейного и целочисленного программирования;
- алгоритмам решения потоковых задач и задач на графах;
- методам построения и анализа математических и имитационных моделей при решении инженерных задач: аппроксимация и интерполяция функций, включая методы теории сплайнов, обработка результатов эксперимента методом наименьших квадратов и несколько типичных оптимизационных задач.

Четвертая глава посвящена современным инструментальным программным средствам, обеспечивающим решение перечисленных выше задач. Изложение включает иллюстрации применения этих средств для решения типичных задач на базе:

- применения MS Excel и технологии VBA для решения простейших инженерных задач;
- использования языка Object Pascal и интегрированной среды разработки Delphi для создания законченных инженерных программных приложений;
- применения пакетов MathCad и MatLab для моделирования процессов и постановки компьютерных экспериментов;
- методологии документирования разработок при помощи современного инструментального средства Visio.

Важно подчеркнуть, что примеры решения задач с использованием конкретных инструментальных средств, рассматриваемые в этой главе, органически связаны с задачами, методы решения которых приведены в третьей главе.

Таким образом, предлагаемый учебник, позволяя читателю увидеть тенденции развития индустрии создания программного обеспечения, на протяжении всего изложения фокусирует его внимание на прикладном, инженерном аспекте математики. С нашей точки зрения он создает основы для более специальных информационных курсов для инженеров, которые могут предлагаться при подготовке специалистов квалификационного уровня специалист или магистр.

Обилие и разноплановость излагаемого материала не являются случайными. Такой характер изложения продиктован стремлением дать возможность преподавателям и студентам различных инженерных специальностей выбрать оптимальный с их точки зрения порядок изучения, набор инструментальных средств и исследуемых с их помощью моделей. Мы надеемся, что нам удалось достигнуть поставленной перед собой цели.

Литература

1. Зарецкая И.Т., Гуржий А.Н., Соколов А.Ю. Информатика – 10. – Киев: Форум, 2004. – 396 с.
2. Зарецкая И.Т., Гуржий А.Н., Соколов А.Ю. Информатика – 11. – Киев: Форум, 2004. – 288 с.
3. Методичний посібник з інформатики / Зарецька І.Т., Семенова Т.В., Соколов О.Ю. – Харків: Факт, 2004. – 192 с.

ТЕСТИРУЮЩИЙ ПРОГРАММНЫЙ КОМПЛЕКС

А.Ю. Соколов, И.А. Трофимова

г. Харьков, Национальный аэрокосмический университет «ХАИ»
им. Н.Е. Жуковского
asokolov@xai.edu.ua

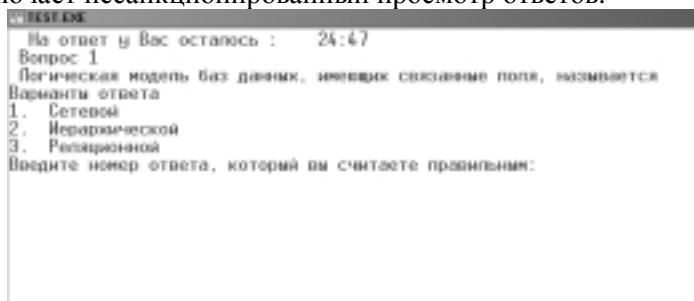
В ходе работы по внедрению компьютерных технологий в учебный процесс кафедрой информатики разработан и используется **универсальный тестирующий программный комплекс**. Он *предназначен* для оценки стартовых знаний студентов, их текущей успеваемости и итогового контроля по темам. Кроме того, кафедра рекомендует данный программный комплекс для использования в средних учебных заведениях с целью компьютеризации контроля знаний учащихся по различным дисциплинам.

Текущая версия комплекса *состоит* из:

- ✓ Базы данных вопросов и ответов
- ✓ Программы шифрования файлов вопросов и ответов
- ✓ Интерактивной тестирующей оболочки
- ✓ Генератора отчетов

Функционирование комплекса заключается в следующем:

1. В зависимости от целей тестирования создается база вопросов и ответов в виде текстового файла специального формата.
2. Файл вопросов шифруется специальной программой, что исключает несанкционированный просмотр ответов.



3. Тестирующая оболочка случайным образом выбирает заданное количество вопросов из базы, выдает их на экран и запоминает номер варианта ответа. Поскольку база вопросов значительно больше числа задаваемых при тестировании, набор вопросов при каждом запуске программы будет отличаться, что

повысит достоверность оценки знаний. Суммарное время на ответы ограничено настройками программы. По его истечении подается звуковой сигнал и выводится запрос личных данных тестируемого.

4. По окончании тестирования запускается генератор отчетов, который выдает на экран количество правильных ответов. Полная версия отчета сохраняется на открытый только для записи сетевой ресурс, что исключает искажение результатов тестирования.

5. Настройки тестирования задаются в специальном текстовом файле, читаемом при запуске тестирующей оболочки. В данном файле можно задать:

- Имя и расположение файла, из которого будут выбираться вопросы;
 - Количество вопросов;
 - Время, выделяемое на тестирование.
- При наличии локальной сети программа может сгруппировать отчеты о тестировании с различных компьютеров в единый отчет.

В настоящее время многие вузы (и ХАИ в том числе) проводят работу по довузовской подготовке учащихся к поступлению и обучению в высшей школе. При разработке данного программного пакета предполагалось создать некий универсальный продукт, который мог бы использоваться для оценки знаний по информатике как в высшей, так и в средней школе. Учитывая то, что уровень технической базы в средних школах, где обучались студенты, может быть разным, данный программный комплекс предъявляет минимальные системные требования (PC 386 и выше, MS DOS версии 6 и выше, MS DOS-сессия под WINDOWS) и рассчитан на минимальные пользовательские навыки. Эти особенности позволяют внедрить данный программный продукт в средних школах с практически любым уровнем оснащенности компьютерных классов для оценки текущей успеваемости и итогового контроля знаний учащихся.

Кафедрой информатики данный программный комплекс используется:

- для стартового среза уровня знаний студентов, начинающих обучение на курсах, читаемых кафедрой. Результаты сводно-

- го отчета по тестированию анализируются на методическом семинаре кафедры для выявления тем, нуждающихся в дополнительной проработке в течение адаптационного курса;
- по окончании адаптационного курса для оценки его эффективности;
 - для проверки уровня подготовки к практическим занятиям;
 - при проведении зачетов по отдельным темам.

ВИКОРИСТАННЯ ВІРТУАЛЬНИХ ЛАБОРАТОРІЙ ПРИ ВИВЧЕННІ КУРСУ “ТЕОРІЯ ОПЕРАЦІЙНИХ СИСТЕМ”

А.М. Стрюк^α, М.С. Стрюк^β

м. Кривий Ріг, Криворізький технічний університет

^α andrey_stryuk@mail.ru

^β marina_novak@mail.ru

Традиційно вважається, що використання віртуальних лабораторій не доцільне при вивченні комп'ютерних дисциплін. Віртуальні лабораторії, в яких широко застосовуються моделювання фізичних процесів, використовуються тоді, коли створення фізичної лабораторії ускладнене великими матеріальними затратами, часом або коли необхідно організувати проведення дистанційних лабораторних робіт. І комп'ютер стає добрим помічником при вивченні багатьох розділів фізики, хімії, електроніки тощо. Коли ж мова йде про вивчення комп'ютерних дисциплін, вважається що сама обчислювальна машина вже є готовою фізичною лабораторією і моделювання процесів, що відбуваються в комп'ютері не потрібне.

Але так буває далеко не завжди. При вивченні комп'ютерних наук ми також можемо опинитися в ситуації, коли прослідкувати фізичні процеси лише за допомогою персонального комп'ютеру неможливо і створення фізичної лабораторії ускладнене різними причинами. Тоді і виникає необхідність створювати комп'ютерну модель. Прикладами таких моделей можуть слугувати віртуальні машини і емулятори, які дозволяють нам вивчати архітектурні особливості різних обчислювальних машин, експериментувати з різними мікропроцесорними засобами, не витрачаючи кошти на придбання коштовного обладнання та таке інше.

Під час своїх досліджень ми звернули увагу на використання віртуальних лабораторій при вивченні курсу “Теорія операційних систем”. Вивчення цієї дисципліни передбачає ознайомлення студентів з архітектурними особливостями різних операційних систем, функціонуванням окремих механізмів в різних системах, та вивчення фундаментальних теоретичних доробок в цій галузі. Лабораторією для вивчення теорії операційних систем слугував звичайний комп'ютерний клас. Під час проведення практичних

та лабораторних занять студенти, працюючи з персональним комп'ютером, можуть познайомитись з функціонуваннями лише двох-трьох операційних систем, в той час як для створення повної картини потрібно не менше десяти. Забезпечити таку кількість різних операційних систем на кожній обчислювальній машині в комп'ютерному класі дуже складно навіть з широким використанням віртуальних машин (більшість з цих систем взагалі можуть не підтримувати архітектури IBM PC). І пов'язано це не лише з великою кількістю обчислювальних ресурсів, що потребують такі лабораторії, не лише з тим, що більша частина практичних занять може бути витрачена на вивчення студентами особливостей роботи користувача з новою операційною системою, а в основному з тим, що навіть фізично працюючи з тією чи іншою операційною системою, ми не маємо змоги безпосередньо прослідкувати функціонування того чи іншого її механізму, а можемо лише констатувати їх роботу по побічним явищам та подіям. Також важливим моментом є складність проведення експерименту. Внаслідок великої кількості параметрів, що впливають на роботу операційної системи та окремих її механізмів, нам важко буде повторити той чи інший дослід, або навіть повністю дотриматися початкових даних експерименту.

Зважаючи на все вище перераховане, нами було прийняте рішення створити програмний комплекс віртуальної лабораторії з дисципліни “Теорія операційних систем”. Перш за все ми звернули увагу на моделювання наступних механізмів операційних систем:

- планування процесорного часу в системах з одним процесором;
- планування в багатопроцесорних системах;
- механізми розподілення пам'яті;
- стратегії заміщення в віртуальній пам'яті;
- планування дискової підсистеми.

Цей комплекс, хоча і є повністю самостійним пакетом програм, став продовженням і доповненням програми навчання та контролю знань ProTOS, що була розроблена минулого року, і отримала широке використання під час викладання теорії операційних систем. Але система ProTOS мала суттєвий недолік – теоретичний матеріал був позбавлений наочних демонстрацій, бу-

ли відсутні можливості проведення практичних та лабораторних занять, які б керувалися самою системою. Ці недоліки повинна компенсувати система віртуальної лабораторії, даючи, з одного боку можливість продемонструвати роботу різних механізмів при викладанні нового матеріалу, так і організувати практичну і самостійну роботу студентів по вивченню тих чи інших механізмів операційної системи.



Рис. 1. Вікно лабораторної роботи “Планування процесорного часу в системах з одним процесором”

Прикладом роботи програми може слугувати вікно модуля “Планування процесорного часу в системах з одним процесором” (рис. 1). В головному вікні програми перераховані всі стратегії планування, що програма дозволяє продемонструвати та проаналізувати. Ми можемо вибрати будь-які з них для проведення загального тесту, зробивши активними прапорці зліва від назви алгоритму. Загальний тест виконає кожен з вибраних алгоритмів планування згідно з початковими тестовими даними і виведе на екран докладну статистику по роботі кожного алгоритму, яку можна буде згодом використати для порівняльного аналізу.

Початкові данні зберігаються в звичайному текстовому файлі, який можна редагувати прямо з програми (кнопка

“Редагування”). Для лабораторної роботи “Планування процесорного часу в системах з одним процесором” файл початкових даних має наступний вигляд:

```
;Файл початкових даних test1.txt
;Процес   Час запуску   Час обслуговування
p1        0             5
p2        2             12
p3        3             8
p4        8             5
;кінець.
```

В файлі початкових даних ми в кожному рядку вказуємо послідовно ім'я процесу, час його запуску та час виконання в умовних одиницях. Ця інформація буде використана при моделюванні роботи кожного з алгоритмів планування.

Крім загального тесту, можна переглянути роботу кожної з перерахованих стратегій, натиснувши кнопку “Старт!”, переглянути статистику по цій стратегії, отримати теоретичну інформацію по роботі кожного алгоритму планування. Також можна налаштовувати різні параметри кожного з алгоритмів, наприклад, змінювати розмір кванту часу для стратегій Round Robin та стратегії зі зменшенням пріоритету.

Основну увагу ми приділяли можливості порівняти між собою різні стратегії, перевірити ефективність кожної на однакових тестових даних. Такий підхід дозволяє студентам під час роботи з лабораторією отримати найбільш повну картину стосовно особливостей роботи кожного механізму операційної системи, еволюційних етапів розвитку цих механізмів, критеріях надійності та ефективності різних алгоритмів.

При вивченні різних стратегій у віртуальній лабораторії студент має змогу працювати одночасно з будь-яким набором моделей. Важливою особливістю є те, що в лабораторії реалізовані як і спрощені моделі, що мають суто теоретичний інтерес і призначені для початкового вивчення, так і моделі наближені по функціональності до роботи механізмів сучасних операційних систем. Це дозволяє студентам прослідкувати роботу того чи іншого механізму від простого до складного. Спочатку отримати повну уяву про саму проблему, про необхідність того чи іншого механізму, далі ознайомитись з теоретичними пошуками і доробками в цій сфері і нарешті проаналізувати реалізацію теоретичних доробок на практиці в сучасних операційних системах. Також кож-

ний розділ ми супроводили допоміжним теоретичним матеріалом, до якого студент може звернутися під час виконання лабораторної роботи.

Надання теоретичної інформації дає студенту можливість більш самостійної роботи з віртуальною лабораторією і розширяє можливості використання останньої. Так, комплекс віртуальної лабораторії з дисципліни “Теорія операційних систем” планується використовувати під час лекцій, для наочної демонстрації нового матеріалу, під лабораторних та практичних занять, пропонувати студентам для самостійної роботи і в майбутньому цей комплекс планується використати в підготовці дистанційного курсу з теорії операційних систем.

МЕРЕЖЕВИЙ НАВЧАЛЬНО-КОНТРОЛЮЮЧИЙ КОМПЛЕКС З МАТЕМАТИЧНИХ ОСНОВ ІНФОРМАТИКИ

Ю.В. Триус
м. Черкаси, Черкаський національний університет
імені Богдана Хмельницького
trius@cdu.edu.ua

Досягнення якісно нового рівня у підготовці фахівців з вищою освітою неможливе без забезпечення розвитку вищої школи на основі нових прогресивних концепцій, запровадження сучасних педагогічних та інформаційних технологій, широкого використання науково-методичних досягнень. Зазначені проблеми обумовлюють і стимулюють процес впровадження у навчальний процес вищої школи нових технологій навчання, які у вузькому розумінні являють собою системи дидактично організованих форм, засобів і методів опрацювання та контролю рівня засвоєння конкретного навчального матеріалу в рамках певної дисципліни, розділу, теми чи питання.

Серед сучасних технологій навчання чільне місце займає технологія модульно-рейтингового навчання, яка забезпечує на основі кількісного показника (рейтингу) комплексний підхід щодо контролю процесу професійної підготовки майбутніх фахівців з урахуванням їх навчальної, науково-дослідної, організаційної і громадської діяльності протягом всього періоду навчання у ВНЗ, стимулює і розвиває у студентів творче ставлення до отриманих знань, прагнення до їх систематичного, активного і самостійного поповнення, створює умови для саморозкриття, самореалізації і самостворення особистості, підвищення її відповідальності за результати своєї праці.

Іншою сучасною технологією навчання є дистанційне навчання, яке базується на принципі *самостійного навчання* студентів, які віддалені від викладача у просторі і (або) часі, але які мають можливість у будь-який час спілкуватися з ним за допомогою засобів телекомунікації. Можливість інтенсивного спілкування між тим хто навчається і тим хто навчає, підсилення діяльнійшої основи, індивідуалізація навчання (навчальних матері-

алів і темпів навчання), виводить цю форму на якісно новий рівень в системі освіти в цілому.

Використання зазначених прогресивних технологій навчання гостро ставить питання про ефективну організацію самостійної роботи, яка стає основним засобом засвоєння студентами навчального матеріалу в час, вільний від обов'язкових навчальних занять на денній формі, не говорячи вже про заочну, екстернатну та дистанційну форми навчання, де самостійна робота, самоосвіта, самоконтроль є домінуючими при професійній підготовці майбутніх фахівців.

В цих умовах значно зростає вага поточного і рубіжного (модульного) контролю та оцінювання навчальних досягнень студентів, які є невід'ємною і важливою частиною педагогічного процесу, що виконують ряд функцій: контрольну, навчальну, виховну, розвивальну, організаційну і методичну. Виконуючи ці функції в процесі підготовки висококваліфікованих фахівців, контроль вирішує також завдання здійснення зворотного зв'язку і керування процесом навчання, оскільки, одержуючи своєчасну і достатню інформацію про те, як відбувається процес засвоєння студентами знань, викладач може коректувати процес навчання і перебудовувати його відповідно до потреб даної конкретної ситуації, що є важливою умовою підвищення якості підготовки фахівців.

На сьогодні найбільш розповсюдженим способом для такого контролю і оцінювання вважають автоматизовані системи тестування (АСК). При цьому все частіше, особливо для поточного контролю, використовується комп'ютерна і телекомунікаційна техніка ([1–5, 7]), що дозволяє:

1. Збільшити обсяг самостійної роботи студентів над навчальними матеріалами дисципліни, що актуально з врахуванням збільшення долі самостійної роботи в загальному обсязі навчального навантаження.

2. Зменшити навантаження на викладачів у частині, що пов'язана з підготовкою контрольних тестів і завдань, при проведенні контрольних заходів і обробці їх результатів, зокрема перевірки робіт студентів, формуванню звітної документації, аналізі типових помилок, формуванні рекомендацій щодо їх усунення з урахування індивідуальних результатів кожного студен-

та.

3. Дати простір викладачу для його діяльності щодо організації гнучкої технології навчання в залежності від контингенту студентів і кількості навчального часу, що виділяється на вивчення курсу. Роль і можливості викладача зростають. Звільняючись від дріб'язкової опіки (цим займається комп'ютер), викладач має можливість індивідуально працювати з кожним студентом і вносити відповідну корекцію в організацію навчального процесу.

4. Підвищити мотивацію і зацікавленість в освоєнні дисципліни на протязі семестру, а не тільки перед підсумковим контролем.

5. Зробити процес оцінювання навчальних досягнень студентів більш об'єктивним.

Як правило автоматизовані системи тестування в локальних і глобальних мережах будуються за такою схемою: на основі одного потужного центрального серверу, який містить всі програми, бази даних з тестовими і контрольними завданнями, і невеликих програм-терміналів для робочих станцій, які виконують навчальні функції і функції перевірки знань, дані для яких приймаються з сервера.

Цей підхід до побудови мережевого навчально-контролюючого комплексу вимагає від програмного забезпечення певної універсальності, що є хоч і логічно доцільним, але невиправданим, оскільки практично неможливо побудувати надійну і однаково зручну для різних дисциплін інтерфейсну систему, яка б використовувалась для демонстрації навчального матеріалу з можливістю практичного закріплення знань на основі інструментальних засобів та проведення адекватної перевірки знань, вмінь і навичок з конкретних тем з урахуванням їх специфіки. Також такий підхід ставить під загрозу розширюваність комплексу, оскільки при підключенні нового тематичного модуля може виникнути проблема несумісності її з центральним сервером.

Крім того, при віддаленому контролі, в умовах України з ненадійними і повільними лініями комунікацій, використання традиційної клієнт-серверної технології контролю, коли база завдань і тестів, які містять мультимедійні матеріали, знаходиться на сервері, а клієнти в режимі реального часу проходять тесту-

вання є не завжди ефективним. На жаль автоматизовані системи тестування не завжди дають можливість досить об'єктивно здійснити таке оцінювання, особливо це стосується математичних і природничих дисциплін. Це пов'язано з тим, що поряд зі знанням певного теоретичного матеріалу, студенти повинні вміти будувати моделі різноманітних об'єктів, процесів і явищ, вміти їх аналізувати, реалізовувати ефективні методи їх розв'язування, використовуючи при цьому логічний і алгоритмічний підходи тощо. За допомогою традиційного тестування це зробити практично неможливо. Хоча є приклади автоматизованих систем контролю, які містять досить складні механізми ідентифікації так званих відкритих відповідей, зокрема, математичних виразів і дозволяють певною мірою вирішувати цю проблему (див., наприклад, [1, 3]). Тому необхідно шукати альтернативні способи використання інформаційно-комунікаційних технологій для дистанційного (віддаленого) вимірювання навчальних досягнень студентів в режимі реального часу.

В цій ситуації одним з можливих напрямів створення АСК, крім інтелектуалізації цих засобів, при контролі та оцінюванні навчальних досягнень студентів з математичних дисциплін доцільно використовувати інструментально-контролюючі програми (ІКП). Найчастіше такі програми являють собою компактні спеціалізовані середовища, які призначені для розв'язування досить вузького класу задач, які є найбільш актуальними при вивченні певної теми або розділу, а також, використовуючи реалізований в них математичний апарат, формувати відповідні контрольні завдання і здійснювати досить ефективний контроль навчальних досягнень студентів з відповідного матеріалу, як в автономному режимі, так і в режимі роботи з локальною та глобальною мережами.

У Черкаському національному університеті силами викладачів, аспірантів і студентів зазначений підхід реалізується у вигляді мереженого комплексу інструментально-контролюючих програм з математичних основ інформатики, до складу якого увійшли програма “Сервер контролю” і навчально-контролюючі програми з математичної логіки (Master of logic, Draw Shem), теорії алгоритмів (AlgoMachines), теорії обчислювальних процесів (SysCalc2, BinOp, Основи теорії похибок) [6].

Концепція створення мереженого навчально-контролюючого комплексу ґрунтується на основі логічно-завершених навчальних інструментально-контролюючих програм з конкретних розділів або тем, які розробляються з врахуванням особливостей навчального матеріалу, та невеликого сервера з підтримкою повного набору засобів для централізованого проведення мережевого контролю з будь-яких тем, що реалізовані в навчально-контролюючих програмах.

Такий підхід до створення систем контролю з математичних дисциплін має певні переваги оскільки така система складається по суті з незалежних програм, на розробку яких не накладається ніяких обмежень, якщо порівняти із розробкою таких програм для центрального сервера. Це надає можливість розробникам створювати оригінальні навчально-контролюючі програми, турбуючись лише про підтримку мережевого сервера контролю. У даному випадку програми можуть використовуватись як автономно, так і в складі мережевого комплексу. Саме цей факт робить недоцільною централізацію всіх програм на функціонуючому сервері, оскільки програми для даної структури комплексу не потрібні всі одночасно, їх можна в будь-який час швидко встановити, а при необхідності деінсталиувати на робочих станціях. А сервер контролю займається виключно адмініструванням мережевого контролю приєднаних клієнтів за допомогою “моніторів” (рис. 1).

Ще однією перевагою такого підходу є менш інтенсивне використання трафіка локальної, а особливо, глобальної мережі для передачі даних із контролюючих програм на сервер і навпаки. При цьому завантаження файлів з контрольними завданнями відбувається по мережі, або генерується автоматично відповідною інструментально-контролюючою програмою.

Програмний комплекс “Сервер контролю” складається з трьох частин (рис. 2):

1. Блок спілкування з клієнтами (сервер);
2. Текстова база даних;
3. WEB-інтерфейс.

Адміністратор – це викладач, який використовує даний програмний комплекс. Для нього доступні усі можливості програми, включаючи перегляд підключень та адміністрування.

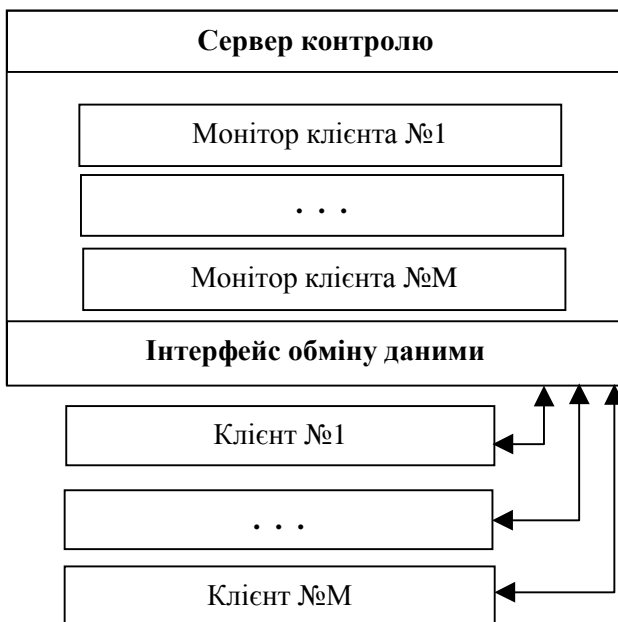


Рис. 1.

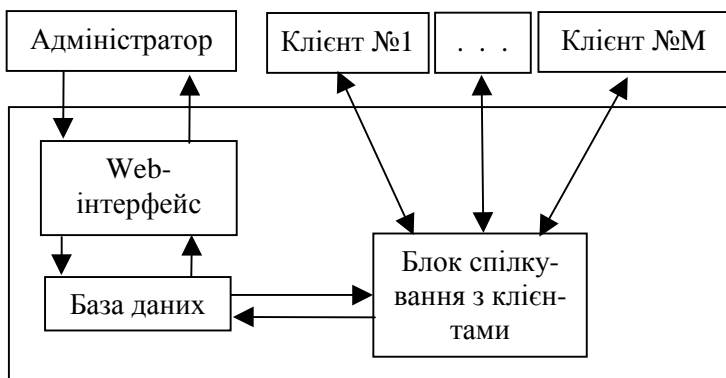


Рис. 2. Структурна схема програмного комплексу "Сервер контролю"

Клієнт – це програма, яка підключається до даного програмного комплексу. Йому “Сервер контролю” доступний лише на рівні повідомлень. Сервер відображає на екрані саме ту інформацію, яку надсилає йому клієнт.

Блок спілкування з клієнтами – цей блок відповідає за прийняття текстових повідомлень, їх розшифрування та подання веб-інтерфейсу в зрозумілому для нього вигляді. В цьому блоці відбувається з’єднання з клієнтами на рівні сокетів. При отриманні повідомлення, сервер розбиває його на дві частини: керуючу та інформаційну. В залежності від керуючої частини, інформаційна частина записується в ту чи іншу область текстового файлу певного формату. Сервер написаний мовою C під Unix і використовує мережевий демон *inetd* для встановлення та підтримки зв’язку з клієнтами.

WEB-інтерфейс – це програма, яка читає інформацію в базі даних, і формує її графічне відображення у вікні браузера. Веб-інтерфейс являє собою Perl-скрипт, який формує на основі отриманої з бази даних інформації html-сторінку, яка завантажується браузером. Інформація подається у вигляді таблиць. Для зручності, клієнти в різних стадіях виконання тесту (наприклад, завершено або не завершено), відображаються в таблицях, що мають різний колір.

При під’єднанні клієнта для нього динамічно створюється **монітор**. Монітор є представником клієнта, що проходить контроль на віддаленому комп’ютері, і відображає інформацію про стан виконання клієнтом контрольних завдань (рис. 3). Він містить:

- прізвище, ім’я та групу клієнта;
- тему, яку обрав клієнт для контролю;
- максимальну кількість балів, яку можна набрати за виконання контрольних завдань;
- час, що залишився на виконання всіх завдань для даного клієнта;
- кількість виконаних ним завдань;
- кількість правильних відповідей;
- набрану кількість балів;
- поточну оцінку знань клієнта;
- по завершенню контролю підсумкову оцінку за 4-х бальною

шкалою;

- повідомлення “Не завершено” при некоректному виході з клієнтської програми.

База даних – являє собою набір текстових файлів спеціального формату. Для зручності навігації по базі даних, вона має спеціальний файл, в якому знаходяться відомості про усі інші файли, що містяться у базі даних. Для кожного клієнта створюється унікальний файл, що зберігає поточну інформацію про стан проходження тесту цим клієнтом.

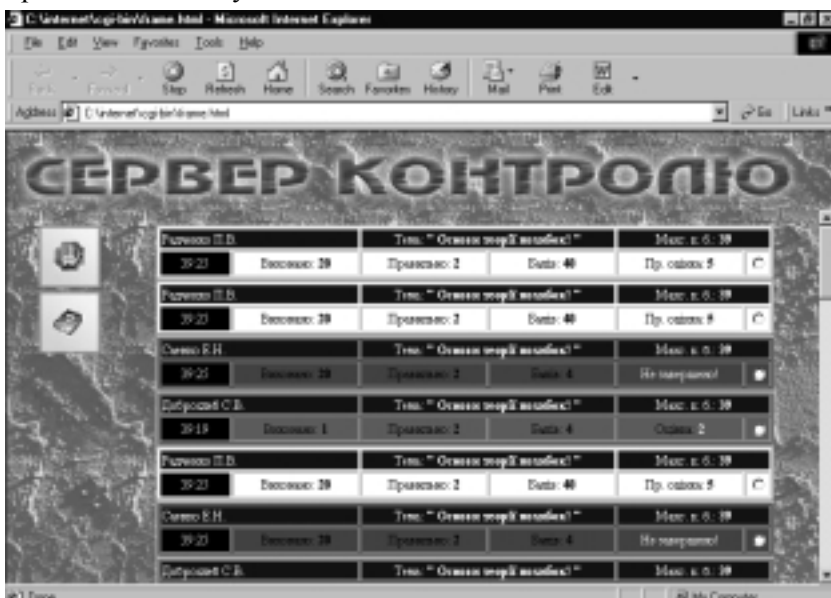


Рис. 4. Головне вікно “Сервера контролю”

Для функціонування серверу необхідний комп’ютер, що працює під операційною системою класу UNIX. Також необхідно, щоб цей комп’ютер був підключений до глобальної мережі Інтернет.

Для функціонування web-інтерфейсу, потрібен комп’ютер підключений до мережі Internet, на якому встановлений інтерпретатор з мови Perl. Для відображення графічної інформації необхідний будь-який браузер.

Серверна програма запускається на обраному адміністратором комп’ютері, який стає сервером для програм-клієнтів. Для

з'єднання з сервером необхідно ввести його IP адресу при входженні до режиму мережевого контролю.

Монітор з графічним відображенням результатів контролю у реальному часі підключених клієнтів запускається на будь-якому комп'ютері, що під'єднаний до мережі Internet. Для цього необхідно ввести у браузері цього комп'ютера URL, де розміщена серверна програма, наприклад: conser.cdu.edu.ua.

На активній Internet-сторінці, що завантажиться, необхідно натиснути “Сервер контролю”, і тоді завантажиться сам web-інтерфейс програмного комплексу (рис. 3).

Мережений навчально-контролюючий комплекс створено для підтримки курсу “Основи інформатики” і може з успіхом використовуватися при вивченні математичних основ інформатики як в середніх, так і вищих навчальних закладах. У перспективі планується удосконалити програму „Сервер контролю”, доповнивши її додатковими можливостями інтерактивного спілкування викладача і студента під час дистанційного контролю.

Література:

1. Баранов В.Ю. Методи оцінювання семантично різномірної відповіді у системі комп'ютерного тестування знань // Комп'ютерно-орієнтовані системи навчання: Зб. наук. праць. – К.: НПУ ім. М.П. Драгоманова. – Випуск 5. – 2002. – С.97-110.
2. Булах І.Є. Комп'ютерна діагностика навчальної успішності. – К.: ЦМК МОЗ України, УДМУ, 1995.– 221 с.
3. Леоненко Л.Л., Поддубный Г.В. Теория подобия конечных последовательностей и ее приложение к распознаванию образов // Автоматика и телемеханика. – М., 1996. – №8. – С.119-131.
4. Мисник Л.Д., Катаєва Є.Ю. Система ТЕСТ – базовий засіб для побудови сучасної методики автоматизованого навчання та контролю знань у ВНЗ // Комп'ютерно-орієнтовані системи навчання: Зб. наук. праць. – К.: НПУ ім. М.П. Драгоманова. – Випуск 6. – 2002. – С.40-48.
5. Сілкова О.В. Застосування автоматизованих атестаційних систем та класифікація тестових завдань при використанні комп'ютерних технологій // Комп'ютерно-орієнтовані систе-

- ми навчання: Зб. наук. праць. – К.: НПУ ім. М.П. Драгоманова. – Випуск 3. – 2001. – С.173-178.
6. Триус Ю.В, Любченко К.М., Латанюк І.В., Дяченко А.Ю. Комплекс інструментально-контролюючих програм для вивчення теоретичних основ інформатики // Комп'ютер в школі та сім'ї. – 2000. – №1. – С. 23-28.
 7. Ухань П.С. Вимоги до автоматизованої системи контролю знань // Комп'ютерно-орієнтовані системи навчання: Зб.наук. пр. – К.: НПУ ім. М.П. Драгоманова, 2001. – Вип. 3. – С. 146-155.

ОСОБЛИВОСТІ МЕТОДИКИ ВИКЛАДАННЯ МЕТОДІВ НАБЛИЖЕНИХ ОБЧИСЛЕНЬ ДЛЯ СТУДЕНТІВ ТЕХНІЧНИХ СПЕЦІАЛЬНОСТЕЙ

О.В. Тумашова, О.М. Коссак
м. Львів, Національний університет “Львівська політехніка”
kora@LITech.net

В зв'язку з комп'ютеризацією сучасної вищої освіти розширилися можливості дослідження об'єктів навколишнього середовища. Для того, щоб дослідити будь-яке явище або об'єкт навколишнього середовища, необхідно дати йому математичний опис, тобто описати його математичну модель у вигляді рівнянь, формул або систем рівнянь, які б якомога точніше описували цей об'єкт. Технологією побудови та аналізу за допомогою сучасних персональних комп'ютерів математичних моделей є обчислювальний експеримент. В основі його лежить розроблення чисельних методів і алгоритмів розв'язування даної математичної моделі, програмування на ПК обчислювального алгоритму, проведення обчислень та аналіз результатів. При розв'язуванні задачі на ПК ми завжди одержуємо не точний, а деякий наближений розв'язок вихідної задачі. Одній і тій же математичній задачі можна поставити у відповідність множину різних дискретних моделей.

Бурхливий розвиток персональних комп'ютерів дав можливість розв'язувати складні інженерно-технічні задачі. Але без практичних навичок застосування наближеного та чисельного аналізів ці задачі розв'язати практично неможливо. Цим пояснюється надзвичайно велике зацікавлення методами наближених обчислень, які складають зміст великого розділу сучасної математики – обчислювальної математики.

Головною метою підготовки фахових спеціалістів є вміння вибрати оптимальний метод наближеного розв'язування для сформованої математичної моделі. Це досягається за рахунок:

- використання ілюстрованих прикладів, пов'язаних з напрямком фахової підготовки;
- вибору оптимального методу, який найточніше наближає математичну модель, що досягається порівнянням його з іншими

методами;

– реалізації чисельних методів на ПК за допомогою алгоритмічної мови Turbo Pascal.

Авторами розроблений комплекс програм на основі алгоритмічної мови Turbo Pascal [1], якій дозволяє реалізувати вибраний чисельний метод для наближеного розв'язання фахової задачі.

Мета викладання курсу “Методи наближених обчислень” є освоєння студентами чисельних методів та застосування їх до розв'язання інженерно-технічних задач за допомогою сучасних ПК. Завдання вивчення курсу полягає у формуванні у студентів твердих фундаментальних знань і практичних навичок використання чисельних методів. Особлива увага приділяється набуттю студентами теоретичних та практичних знань у побудові розв'язків сучасних інженерно-технічних задач.

Література

1. Коссак О., Тумашова О. Методи наближених обчислень: Навч. посібник – Львів: БаК, 2003. – 168 с.

ВИКОРИСТАННЯ МЕТОДУ МНОЖНИКІВ ПРИ ВИВЧЕННІ ЦИКЛІВ

О.Г. Тюрін

м. Мелітополь, Мелітопольський державний педагогічний
університет
qknife@mail.ru

Кожна складна, а тому і цікава програма завжди містить оператори циклу. Циклічні оператори виконують у програмі подвійну роль: по-перше, скорочують обсяг програми і тому значно полегшують її читабельність [1, 2], по-друге, виконують дії, кількість яких невизначена константою. Записати такі дії без використання циклів, а лише за допомогою повного обсягу операторів присвоювання або виклику підпрограм зовсім неможливо (наприклад, видати усі числа від 1 до N). Тому вивчення циклів є дуже актуальною та цікавою, але у той же час складною та суперечливою темою [3, 4]. Школярів та студентів не важко навчити, коли потрібно використовувати цикл, важко навчити, як скласти циклічну конструкцію.

В якості альтернативного методу вивчення циклів у даній статті пропонується метод множників, використання якого розглядається на прикладах використання циклу з параметром мови програмування Паскаль – FOR..TO..DO. Але даний метод може використовуватися у будь-якій мові, бути елементом багатьох методик навчання програмуванню. Наприклад, він є складовою частиною методу алгоритмічних схем, який значно полегшує вивчення програмування [5–7].

Для того, щоб визначити, чи є можливим вказати дії у циклі, потрібно вміти розрізняти однакові, подібні та різні дії, знати, що таке циклічна змінна, ітерація, крок, початкове та кінцеве значення циклічної змінної.

Однакові дії – це дії, які складаються із однакових операторів та однакових значень к ним.

Наприклад, Write(1);Write(1);Write(1);

Подібні дії – це дії, які складаються із однакових операторів та різних значень к ним. Але при цьому дані мають між собою закономірність, яку можна виразити формулою через невідому.

У наступних прикладах формулу будемо виражати через змінну X , яка буде позначати номер даного, тобто послідовно змінюватись як 1,2,3 і т.п.

`Write(1);Write(2);Write(3);` Формула: X .

`Write(2);Write(3);Write(4);` Формула: $X+1$.

`Write(2);Write(4);Write(6);` Формула: $X*2$.

`Write(5);Write(4);Write(3);` Формула: $6-X$.

Найчастіше, подібні дані мають лінійну закономірність, при якій значення змінюються з однаковим кроком.

При визначенні однакових та подібних даних необхідно пам'ятати, що однаковість або подібність дій визначається тільки через числа, тобто подібні дані повинні відрізнятися лише числовими значеннями. Якщо дії мають хоч одну відмінну літеру, це вже різні дані, які ніколи не можливо скоротити для використання у циклі.

Саме тому, коли, наприклад, необхідно запросити від користувача 10 чисел, їх потрібно позначити або однією змінною (тоді це будуть однакові дії), або елементами масиву, які закономірно відрізняються лише номером (це вже будуть подібні дії). Позначення цих даних через $A, B, C \dots$ є розповсюдженою помилкою початківців, тому що у цьому випадку існування циклу не можливе.

Різні дії – це дії, яка складаються із різних операторів або дії з даними без закономірності.

`Write(1);Readln(A);X:=5;`

`Write(1);Write(12);Write(-3);`

`Write(A);Write(B);Write(C);`

Циклічна змінна (ЦЗ) – це змінна, яка керує роботою циклу

У прикладах подібних дій циклічною змінною виступав би X .

Ітерація – це кожне чергове виконання циклу. Скільки раз виконається цикл, стільки у ньому ітерацій.

Якби приклади подібних дій виконувались у циклі, то такий цикл містив би 3 ітерації.

Крок – це величина змінювання циклічної змінної після виконання кожної ітерації.

У прикладах подібних дій, крок для X дорівнює $+1$, так як після кожної ітерації, X збільшується на 1.

Початкове значення (ПЗ) – це значення, з якого розпочинається змінювання циклічної змінної.

Кінцеве значення (КЗ) – це значення, на якому закінчується змінювання циклічної змінної.

У наших прикладах початкове значення = 1, кінцеве = 3.

Ці значення впливають на формулу закономірності, крок, але кількість ітерацій обчислюється однаково по формулі: $|КЗ-ПЗ|+1$

5,4,3: ПЗ=1, КЗ=3, Крок=+1, Формула:(6-X),

Кількість ітерацій:|3-1|+1=3

5,4,3: ПЗ=5, КЗ=3, Крок=-1, Формула:(X),

Кількість ітерацій:|5-3|+1=3

Для створення циклу необхідно спочатку бачити усі повні дії, що містять лише оператори умови, присвоювання або виклику підпрограм. Якщо серед дій буде виділятися частина, що повторюється, з однаковими або подібними значеннями, то можлива присутність циклу. Частина, що повторюється, найчастіше складається з одного оператора. Якщо частина складається з декількох операторів, цикл повинен буде містити складовий оператор. Далі виконують такі кроки:

1. Спочатку вказують шапку циклу:

for <Ц.З.>:=0 to <кількість частин-1> do

Тобто початкове значення по нашому методу завжди повинне бути 0, а крок циклічної змінної +1.

2. Після шапки циклу вказують одну частину, що повторюється, причому:

а) однакові значення залишити без змін;

б) подібні значення вбрати, а замість них вказати формулу:

ПЗ послідовності + Крок послідовності * Ц.З.

Ц.З. у формулі вказується таж сама, що і у шапці циклу. При необхідності, формулу потрібно взяти у дужки.

Розглянемо деякі приклади використання методу. У прикладах, числові послідовності частин, що повторюються, будемо позначати жирним шрифтом.

ВИДАТИ ЧИСЛА 1 2 3 4 5:

Повні дії: Write(1);Write(2);... Write(5);

Скорочення: тут бачимо одну подібну послідовність (1,2 ... 5), яка починається зі значення 1 і має крок +1, тому використовуємо формулу. Кількість частин = 5:

*for x:=0 to 4 do Write(1+x); {1+x=1+1*x}*

ЗНАЙТИ КІЛЬКІСТЬ ЛІТЕРИ 'А' У РЯДКУ.

Повні дії: *Readln(Str);K:=0;*

if Str[1]='a' then K:=K+1;

if Str[2]='a' then K:=K+1;

...

if Str[Length(Str)]='a' then K:=K+1;

Скорочення: перші два оператори не потрапляють до циклу (вони є неподібними до основної частини). Блок подібних дії містить одну однакову послідовність з одиниць, і одну подібну (1,2 ... Length(Str)). Кількість частин = Length(Str):

Readln(Str);K:=0;

for x:=0 to Length(Str)-1 do if Str[1+x]='a' then K:=K+1;

ВИДАТИ ЧИСЛА ВІД 1 ДО N ПО ДІАГОНАЛІ.

Повні дії: *Readln(n);*

GotoXY(1,1);Write(1);

GotoXY(2,2);Write(2);

...

GotoXY(n,n);Write(n);

Скорочення: введення з клавіатури не потрапить до циклу. Подібні дії має три подібні послідовності, які починаються з одного і того ж значення (1) і мають однаковий крок (+1). Кількість частин = n. Наявність декількох операторів у частині, що повторюється, обумовлює необхідність складового оператора:

for x:=0 to n-1 do begin GotoXY(1+x,1+x); Write(1+x); End;

Вкладені цикли виникають тоді, коли після першого скорочення знов виділяються частини, які можна скоротити.

ВИДАТИ ЗАДАНІ ЧИСЛА ТАБЛИЦЕЮ:

Повні дії:

1 2 3 4		<i>Write(1); Write(2);... Write(4); Writeln;</i>
2 3 4 5		<i>Write(2); Write(3);... Write(5); Writeln;</i>
3 4 5 6		<i>Write(3); Write(4);... Write(6); Writeln;</i>

Як бачимо, дії не можна скоротити до одного циклу, поперше, із-за WriteLn і Write(*) (так як дії, які відрізняються хоч однією літерою є різними), а, по-друге, тому, що числа, записані в одну послідовність усе рівно не будуть мати закономірність. Але при цьому ми можемо скоротити дії по рядках:

1-ше скорочення:

```
for x:=0 to 3 do Write(1+x); WriteLn;
for x:=0 to 3 do Write(2+x); WriteLn;
for x:=0 to 3 do Write(3+x); WriteLn;
```

Але тепер ми бачимо, що знову є частина, що повторюється. Вона містить дві однакові і одну подібну послідовності. Для другого скорочення заведемо нову змінну і тепер у формулах будемо використовувати її, а не циклічну змінну першого скорочення.

2-ге скорочення:

```
for y:=0 to 2 do begin {так як у цьому циклі два оператори}
for x:=0 to 3 do Write(1+y+x); WriteLn; End;
```

ВИДАТИ ЗАДАНІ ЧИСЛА ТАБЛИЦЕЮ:

Повні дії:

```
1 2 3 | Write(1);... Write(3); WriteLn;
1 2   | Write(1);... Write(2); WriteLn;
1     | Write(1);... Write(1); WriteLn;
```

1-ше скорочення:

```
for x:=0 to 2 do Write(1+x); WriteLn;
for x:=0 to 1 do Write(1+x); WriteLn;
for x:=0 to 0 do Write(1+x); WriteLn;
```

2-ге скорочення:

```
for y:=0 to 2 do begin
for x:=0 to 2-y do Write(1+x); WriteLn; {2-y=2+-1*y}End;
```

Застосовуючи метод множників, можна заперечити, що не дуже зручно використовувати початкове значення 0. У даних прикладах легше було б взяти за такий 1. Але у цьому випадку метод стає не універсальним. Початкове значення 0 обрано тому, що будь-яка закономірність скорочуються по однаковій формулі лише при ПЗ=0. Наприклад, для виведення таких чисел таблицею, нелегко підібрати формулу при ПЗ=1 і відносно просто при ПЗ=0:

1 8 15	1-ше скорочення:
5 13 21	<i>for x:=0 to 2 do Write(1+7*x);Writeln;</i>
9 18 27	<i>for x:=0 to 2 do Write(5+8*x);Writeln;</i>
	<i>for x:=0 to 2 do Write(9+9*x);Writeln;</i>
	2-ге скорочення:
	<i>for y:=0 to 2 do begin</i>
	<i>for x:=0 to 2 do Write(1+4*y+(7+y)*x);Writeln;</i>
	<i>End;</i>

Література

1. Кергаль М. Методы программирования на Бейсике (с упражнениями): Пер. с англ. – М.:Мир, 1991. – 288 с.
2. Довгаль С.И., Сбитнев А.И. Персональные ЭВМ: ТурбоПаскаль V6.0, Объектное программирование, локальные сети (учебное пособие). – К.: Информсистема сервис, 1993. – 440 с.
3. Грузман М.З. Обучение учащихся средней школы программированию на основе структурного подхода: Дис. к.п.н. / НИИ педагогики УССР. – К., 1986. – 176 с.
4. Кушниренко А.Г., Лебедев Г.В. 12 Лекций о том, для чего нужен школьный курс информатики и как его преподавать. Методическое пособие. – М.: Лаборатория Базовых Знаний, 2000. – 464 с.
5. Тюрін О.Г. Методика вирішення задач з програмування на основі методу алгоритмічних схем // Науковий вісник Південноукраїнського державного педагогічного університету ім. К.Д.Ушинського: Зб. наук. праць. Випуск 10. – Одеса: ПДПУ ім. К.Д.Ушинського, 2002. – Частина 1. – С.108-111.
6. Тюрін О.Г. Використання методу алгоритмічних схем при вивченні програмування // Наукові записки. – Випуск 46. – Серія: Педагогічні науки. – Кіровоград: РВЦ КДПУ ім. В. Винниченка. – 2002. – С.122-124.
7. Тюрін О.Г. Створення циклічних конструкцій за методом алгоритмічних схем // Наукові записки. – Випуск 51. – Серія: Педагогічні науки. – Кіровоград: РВЦ КДПУ ім. В. Винниченка. – 2003. Частина 1. – С.191-194.

ЕЛЕКТРОННІ СИСТЕМИ КОНТРОЛЮ ЗНАТЬ ТА ЗАСОБИ ДЛЯ ЇХ СТВОРЕННЯ

І.Є. Фільо

м. Рівне, Український державний університет водного
господарства та природокористування

Обов'язковою умовою підвищення якості навчання студентів є систематичний контроль засвоєння знань. Поряд із використанням традиційних форм і засобів організації контролю доцільним є проведення тестування. Тести допомагають не тільки виявити готовність студентів до освоєння нового навчального курсу, виявити конкретні пробіли в знаннях студентів, спланувати необхідну цілеспрямовану роботу щодо корекції процесу навчання, але й прогнозувати подальший процес навчання і його результати. Розрізняють дві технології тестування: *бланкова* і *комп'ютерна*. У порівнянні з традиційними формами контролю комп'ютерне тестування має ряд переваг:

- швидке одержання результатів тестування;
- звільнення викладача від трудомісткої роботи з обробки результатів тестування;
- однозначність фіксування відповідей, конфіденційність при анонімному тестуванні;
- комп'ютерне тестування створює позитивну мотивацію у студентів – тестування на комп'ютері цікавіше в порівнянні з іншими формами опитування.

Тому представляється актуальною практика розробки різноманітних програмних засобів для підготовки і організації тестування з використанням комп'ютера.

Комп'ютерний тестовий контроль може бути організований у двох напрямках. Перший – це створення окремих, ізольованих систем контролю знань і другий – тестові комплекси як компоненти електронних підручників, інтелектуальних навчальних систем.

Комп'ютерне тестування може бути *контрольним* або *навчальним*. Контрольне тестування призначене для підсумкової перевірки знань студента. Після складання тесту студент отримує відомість про свою успішність: кількість вірних та невірних

відповідей, час витрачений на складання тесту та отриману оцінку. Навчальне тестування використовується, як правило, для самоконтролю. Після проходження тесту студентові надається інформація про результати тестування і посилання на матеріали з питань, на які він відповів невірно. Ознака навчального або контрольного тесту показує, чи містять питання в тесті посилання на навчальну інформацію.

На сучасному етапі розвитку програмного та технічного забезпечення можна виділити такі системи комп'ютерного тестового контролю:

Тестуюча програма-оболонка, що може працювати в одному з трьох режимів: створення нової бази тестів; редагування існуючої бази тестів; тестування.

Програма дозволяє змінювати кількість питань, що задаються при тестуванні і час на один тест. По закінченні тестування видається протокол з результатами. Розроблені програми дозволяють вирішити задачу автоматизації створення тестів і процедури тестування на основі закритих тестів із множинним вибором варіанта відповіді і можуть бути використані для контролю в процесі освоєння студентами матеріалу різних навчальних дисциплін.

Існує велика кількість систем, призначених для створення електронних тестів та проведення тестування: TestEdit 2.1 ТОО "ИНСИКОМ", м. Новосибірськ, "Тести 6.01" Олександра Мельникова, AnyTest невідомого автора, "Конструктор Тестів II 2.01" TsoftGroup, Test Commander 2.02 Great Hedgehog's Software Research Labs, SunRav TestOffice і багато інших.

Дуже часто для створення комп'ютерних тестуючих систем використовують сучасні мови об'єктно-орієнтованого програмування, такі як Visual Basic, Delphi.

Інтерактивні документи для проведення комп'ютерного тестування. Відмінною рисою таких документів є швидкість і простота створення збірників тестів для динамічної перевірки знань і самоперевірки. Інтерактивний документ може складатися з декількох окремих файлів-сторінок, кожний з яких може бути як текстовим, так і документом будь-якого іншого виду. В залежності від дій користувача змінюється порядок перегляду, автоматично відкриваються інші документи чи відтворюються мультимедійні елементи.

тимедійні файли. Інтерактивні електронні документи використовують такі засоби, як *гіперпосилання* і *макроси*, а також застосовують *об'єкти*, вставлені в ці документи.

Як засоби підготовки інтерактивних документів для проведення комп'ютерного тестування, а також їхнього перегляду, можна використовувати такі програми пакета MS Office, як Word, Excel, Access, PowerPoint.

Тестуючий комплекс – один з найважливіших компонентів електронного підручника, призначений для придбання навичок рішення задач, проведення контрольних робіт, оцінки засвоєння учнем розділів курсу. Після такої оцінки повинен бути обраний той варіант подання навчального матеріалу, що найкраще відповідає індивідуальним психофізіологічним особливостям того, хто навчається. Існуючі можливості методики і техніки дозволяють це зробити.

On-line тестування. На сьогодні розроблено багато методів on-line тестування, які базуються на використанні CGI та інтерактивних HTML-форм. Наслідуючи основні принципи роботи з on-line документами, вони реалізують вільний принцип пересування по документу та, відповідно, вільний порядок відповідей на запитання. Кожна спроба відповісти фіксується та враховується під час формування тестової оцінки. Звичайно кожному студентові дозволяють пройти тестування кілька разів. Тестова система як програма – це сукупність Web-сторінок, Java- та Perl-скриптів. Web-сторінка запускається на виконання за допомогою будь-якого браузера. Розроблені програми керують і контролюють появою на екрані тої чи іншої сторінки після відповідної дії користувача. Такі новітні технології як гіпертекстова мова з Java-скриптом та шлюзовий інтерфейс WWW-серверів дозволяє проводити тестування як в локальній так і в глобальній мережах. Кожен варіант відповіді посилається на певну функцію Java-скрипта, який, в свою чергу, переводить варіант відповіді в спрощений вигляд і відсилає його на сервер для перевірки програмою CGI (звичайний шлюзовий інтерфейс). Сервер перевіряє правильність відповіді, зберігає результати виконання тесту в базі даних для кожного користувача, і після завершення тестування повертає користувачу його результати. CGI дає можливість учителю адмініструвати тест. Зокрема, для більшої захи-

шеності тесту використовується аутентифікація, тобто перевірка прав користувача на управління тестом через введення пароля.

Тестові завдання такого типу: користувачеві потрібно обрати одну відповідь з декількох або користувачеві потрібно обрати декілька вірних відповідей, найзручніше реалізовувати за допомогою інтерактивних документів комп'ютерного тестування. Досить потужним засобом для створення збірок тестів є програма MS PowerPoint, за допомогою якої можна швидко і просто розробити тестуючу інтерактивну презентацію з певної теми, яка б реалізувала саме такий тип тестів.

Застосування комп'ютерних слайдових презентацій у навчальному процесі, зокрема для проведення комп'ютерного тестування, має такі переваги:

1) допомагає викладачу забезпечити зручне і наочне подання матеріалу, створити уніфіковану структуру слайду з питаннями;

2) забезпечують відновлення початкового стану файлу після завершення тестування;

3) дозволяють інтенсифікувати і персоніфікувати процес повторення матеріалу студентом, прискорити його адаптацію до віртуального середовища у випадку наступного комп'ютерного тестування;

4) забезпечують публікацію в Internet та перегляд за допомогою Web-браузера;

5) мають властивість захоплювати користувача (студента) і підтримувати в ньому зацікавленість матеріалом;

6) тестуючі інтерактивні презентації створюють позитивну мотивацію у студентів – контроль стає цікавішим в порівнянні з іншими формами опитування.

Інтерактивну презентацію умовно можна поділити на три частини: титул, тест і результати тестування (рис. 1).

Слайд-титул містить інформацію про тему тестування та студента який проходить тест, також проводиться перевірка наявності на диску текстового файлу послідовного доступу, який був створений при попередньому тестуванні. В цьому файлі мають реєструватися всі результати тестування для кожного тест-слайда. Тест-слайди відображають питання з варіантами відповідей, причому найчастіше вони мають лінійну схему навігації з

можливістю або ні повернення до попереднього слайду. Слайд-результат служить для виведення результатів тестування так, щоб студент міг бачити власну оцінку, кількість вірних і неправильних відповідей та завершення процесу тестування.



Рис. 1. Лінійна навігаційна схема тестової слайдової презентації

Таким чином, можна стверджувати, що сучасні системи комп'ютерного тестового контролю допоможуть підготувати та реалізувати контроль, оптимізувати процес створення тестів, скоротити час на перевірку.

Література

1. Лозниця В.С. Психологія і педагогіка: основні положення. Навчальний посібник для самостійного вивчення дисципліни. – К.: ЕксОб, 2000. – 304 с.
2. Мануйлов В.Г. Введение в технологию разработки педагогических мультимедийных мастер-шаблонов. // Информатика и образование. – 2001. – №2. – С. 70-77.

НОВІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ У ВИЩІЙ ШКОЛІ: ПРОБЛЕМИ, АКТУАЛЬНІСТЬ

Н.А. Хміль, С.В. Дяченко

м. Луганськ, Луганський національний педагогічний університет
імені Тараса Шевченка
dvd_lug@online.lg.ua

У складному комплексі першочергових проблем комп'ютеризації й інформатизації вищої школи та освіти в цілому однією з ключових задач виступає підготовка педагога до його професійної діяльності в умовах впровадження нових інформаційних технологій (НІТ) в освіту.

Ключовою фігурою інформатизації освіти є вчитель, педагог-предметник, а не вчителі інформатики, які вважалися головними провідниками ідей інформатизації протягом тривалого часу [1]. На сьогоднішній день склалася така ситуація, що більшість вчителів-предметників не мають ніяких знань в області інформатики і навичок роботи на персональному комп'ютері. Ті ж, котрі мають деякі навички роботи на комп'ютері, як правило, не орієнтуються в питаннях педагогічної інформатики (зокрема, не володіють знаннями щодо вимог до педагогічних програмних засобів, знанням методик організації навчальної діяльності на предметних уроках з використанням персонального комп'ютера тощо). Вчителів-предметників, здатних самостійно і грамотно створити найпростіший педагогічний програмний засіб (ППЗ) необхідного змісту зі свого предмету, в країні на даний момент – одиниці.

Таким чином, на сучасному етапі найважливішою задачею навчання студентів педагогічних вузів є підготовка майбутніх вчителів до успішного застосування нових інформаційних технологій у процесі навчання учнів загальноосвітніх шкіл. У зв'язку з цим одним з важливих критеріїв професіоналізму сучасного вчителя є його підготовка в області інформатики та інформаційних технологій.

Підготовка фахівців педагогічних вузів різних рівнів (бакалавр, учитель, магістр) в області інформатики виступає в декількох аспектах: загальноосвітній аспект містить загальний вступ до

інформатики і практикум з інформаційного моделювання, педагогічний аспект розглядає педагогічну інформатику, технологічний аспект реалізується у вивченні предметів НІТ та методики викладання предмета [2].

Загальним введенням до інформатики передбачається знайомство майбутніх вчителів з різними видами операційних систем, уміння працювати з персональним комп'ютером тощо.

Педагогічний аспект акцентує свою увагу на тому, що істотною особливістю прищеплювання інформаційної культури студентам є спрямованість на майбутню професійну діяльність. На цьому принципі повинна ґрунтуватися вузівська інформатика – вчителю-предметнику потрібна особлива, предметна інформатика. Мова йде про вивчення й використання спеціальних засобів НІТ, застосування яких найбільш характерне для даної предметної області. Студент, майбутній вчитель-предметник, повинен знати про можливості використання у своїй повсякденній і майбутній професійній діяльності нових інформаційних технологій, вміти аналізувати інформацію, знати особливості інформаційних потоків у своїй сфері діяльності.

Технологічний аспект реалізується у вивченні предметів НІТ і методики викладання предмета. Протягом ряду років у нашому університеті викладається дисципліна «Нові інформаційні технології», яка навчала основним принципам роботи на персональному комп'ютері та знайомила з новим програмним забезпеченням, але в сучасних умовах цього недостатньо. Швидка зміна поколінь комп'ютерної техніки, безперервне відновлення технічних характеристик і параметрів ОТ (ЕОМ) ускладнюють створення стабільних навчально-методичних засобів професійного навчання щодо комп'ютерної техніки, вимагають періодичного відновлення комп'ютерної техніки, перегляду і корегування психолого-педагогічних і дидактичних умов її ефективного використання в навчально-виховних цілях.

Аналіз викладання НІТ показав необхідність організації занять дещо іншого призначення, а саме спрямованих на розробку електронних навчально-методичних посібників і форми їхнього ефективного застосування в процесі навчання. На наш погляд, одним з найважливіших компонентів підготовки майбутніх вчителів повинне стати формування знань, умінь і навичок в області

розробки педагогічних програмних засобів. У зв'язку з цим нами ведуться дослідження в області змісту дисципліни «Нові інформаційні технології» для педагогічних вузів. Нами була розроблена програма, яка включає наступні розділи:

- текстовий редактор для вивчення додаткових можливостей роботи з текстовою інформацією;
- програма підготовки презентацій для створення мультимедійних презентацій й можливості їх використання в навчальному процесі;
- методика підготовки навчально-дидактичного матеріалу та структуризація його при створенні електронних навчальних посібників;
- електронна пошта й Інтернет-ресурси для використання додаткових можливостей при організації занять з використанням інформаційних технологій і пошуку додаткової інформації.

Однією з головних задач дисципліни НІТ є задача навчити студентів – майбутніх вчителів-предметників – створювати такі педагогічні програмні засоби, які відповідали б загальноприйнятим психолого-педагогічним нормам навчального процесу. Тобто, програмний продукт повинен відбивати як мінімум 3 функції навчального процесу:

- 1) розвиток логічного мислення учнів (ППЗ допомагають учневі самому добувати необхідні знання, тобто працювати творчо);
- 2) поповнення необхідних знань;
- 3) оцінювання процесу навчання, в якому ініціатором виступає сам учень. При цьому зростає мотиваційний фактор такого навчального процесу.

Вчитель, який не має систематизованих знань змісту і можливостей нових програмних засобів, не має навичок і умінь їхнього практичного застосування в навчально-виховному процесі, зазнає значних труднощів під час використання нових інформаційно-комунікаційних технологій у навчанні. Навчити жити і працювати в умовах, які швидко змінюються – найважливіша задача вищої школи, але це неможливо без формування інформаційної культури.

Усе вищесказане доводить актуальність обраного нами дослідження з формування інформаційної культури майбутніх вчи-

телів-предметників і робить розробки щодо внесення коректив у зміст освіти педагогічних вузів необхідними на сучасному етапі.

Література

1. Румянцев И.А., Степанов С.А. Концепция среднего общеобразовательного учреждения постиндустриального общества // Информатика и образование. – 2001. – № 8.
2. Марков С.А. Информатика как базовая наука образования // Информатика и образование. – 1998. – № 6.

РОЗРОБКА ПРОГРАМНОГО КОМПЛЕКСУ ТЕМАТИЧНОГО ОЦІНЮВАННЯ ЗНАТЬ УЧНІВ

С.В. Шаров

м. Мелітополь, Мелітопольський державний педагогічний
університет
seg@mpu.melitopol.net

Дванадцятибальна шкала оцінювання знань учнів передбачає обов'язковий тематичний контроль знань, який забезпечує об'єктивне оцінювання досягнень учнів і відображає реальні досягнення у вивченні програмного матеріалу.

Сьогодні поряд із традиційними формами контролю знань дуже широко використовується контроль та аналіз при використанні комп'ютерної техніки. Вагомими перевагами такого контролю є об'єктивність оцінювання знань, заощадження урочного часу, безпосередня фіксація результатів, забезпечення оперативного зворотного зв'язку, можливість використання як індивідуально, так і для цілих груп. Ці переваги дають вчителю більшу можливість для індивідуальної роботи як із встигаючими, так із менш успішними учнями.

Разом з тим, використання комп'ютерів в навчальному процесі повинно передбачати розробку якісних комп'ютерних навчальних, контролюючих та інших програм, які ефективно і швидко можуть подати певний матеріал у зрозумілій формі, визначити рівень знань учнів з даного предмету, провести аналіз та корекцію. Розробка таких програм – це комплексне завдання, яке вирішується спільними зусиллями педагогів, методистів і програмістів. Для успішного використання в школі програмно-педагогічний засіб повинен відповідати певним вимогам, зокрема стійкості, простоті, очевидності, слухняності.

Всі ці фактори та вимоги були використані нами при розробці “Програмного комплексу тематичного оцінювання знань учнів” (автор Бельчев П.В., за редакцією д.пед.н., проф. Павленко А.І.), який призначений для організації контролю знань учнів у загальноосвітніх навчальних закладах. Даний комплекс складається з трьох блоків: блоку контролю, блоку адміністрування та аналізу даних, блоку допомоги користувачу (довідкова інформа-

ція). Зупинимося на блоці контролю.

В його основу було покладено поняття про адаптивний тест, відповідно якому кожне наступне завдання залежить від вірності або помилковості відповіді на попереднє. Це дозволяє враховувати рівень підготовленості учня, дає можливість отримувати завдання відповідно його знанням.

Структура та зміст завдань тематичного тестування забезпечується розподілом завдань за трьома рівнями складності. До завдань першого рівня відносять основні, достатньо прості запитання. Ці завдання передбачають вибір однієї правильної відповіді із декількох варіантів. До завдань другого рівня належать більш складні завдання, які передбачають вибір однієї чи декількох вірних відповідей із декількох варіантів. Завдання третього рівня передбачають вибір декількох вірних відповідей із декількох варіантів, але вже в певному порядку.

Така форма контролю природно повинна починатися з виконання завдання другого рівня складності, яке орієнтовано на середнього учня. В залежності від того, вірно чи невірно було виконане завдання, наступне завдання буде відповідно вищого або нижчого рівня складності.

Всього учню пропонується виконати десять завдань, за вірне виконання яких він може отримати максимум десять балів. Таким чином, програмний комплекс поєднує переваги 12-бальної системи оцінювання навчальних досягнень учнів з комп'ютерною формою контролю.

Схильність учня до роботи з програмним засобом значною мірою визначається наявністю зручного інтерфейсу. Дотримання цієї вимоги відбувається за допомогою наступних особливостей:

- 1) всі дії по введенню інформації були зведені до вибору одного варіанту із декількох запропонованих;
- 2) блок контролю налічує мінімальну кількість елементів управління і вікон, кожне з яких виконує специфічну функцію;
- 3) кожний рівень складності при контролі знань має свій колір та пояснювальний текст, що дозволяє досить легко орієнтуватися в рівнях;
- 4) вся текстова інформація виконана шрифтом без позначок, достатньо великого розміру, різним кольором

для поліпшення сприйняття інформації;

5) вибір потрібної інформації для реєстрації модуля здійснюється за допомогою переліків, що дає певні переваги:

- значне заощадження часу при виборі даних;
- економія міста на екрані монітору;
- виключення помилок при введенні даних;
- виключення помилок при доступі до бази даних.

Програмний комплекс відповідає критерію зрозумілості, відповідно до якого програма не повинна ускладнювати роботу користувача необхідністю вибору з декількох елементів керування. Це вирішується за допомогою активізації або деактивізації елементів на певних етапах роботи програми. Найбільш повно цей критерій реалізовано у вікні вибору даних про модуль та учня, який буде проходити тестування. При такому виборі даних кожний наступний перелік активізується тільки в разі активізації попереднього. Під час завантаження програми учень може скористуватися тільки першим переліком. Так послідовно проходить вибір інформації в залежності від попереднього вибору, тобто дані водночас фільтруються. Використання цього способу введення даних дозволяє виключити можливість необачного або випадкового натискання елементів керування, а також значно зменшити об'єм інформації, яку потрібно опрацювати при виборі необхідної.

Після завершення тестування інформація про результати тестування подається у вікно, яке містить дані про кількість вірно виконаних завдань по кожному рівню складності, а також суму балів в цілому. Більш повну інформацію можна отримати в блоці адміністрування та аналізу даних, який дозволяє отримати:

- відомості про учня за модуль;
- відомості про учня за модуль (помилки);
- відомості про клас за модуль;
- відомості про клас за предмет.

Ще одна вимога при розробці контролюючих програм – виключення або зведення до мінімуму кількість вгаданих вірних відповідей. Для вирішення цієї проблеми ви використали наступні засоби:

1. Перший включає в себе специфіку самого тестування,

який складається з трьох рівнів складності. Перелік завдань визначається в першу чергу підготовленістю учня, який проходить тестування та залежить від вірності або невірності відповіді на поточне запитання.

2. Кількісний засіб, відповідно якого в процесі складання тестових завдань на кожний рівень потрібно скласти двадцять запитань. Таким чином, сумарна кількість завдань по трьом рівням складає шістдесят.

3. Завдання, які пропонуються виконати учню, отримуються випадковим чином. Після того, як учень вибрав всі необхідні дані і почав тестування, програма на початку роботи створює три масиви з десяти елементів кожний (максимальна кількість завдань тестування), які відповідають трьом рівням складності і значення яких вибираються із певного діапазону. В залежності від того, яким за номером було задане запитання і від того, вірна або невірна була попередня відповідь, буде визначатися номер завдання в загальній базі завдань. Номер поточного завдання є номером елементу масиву завдань відповідного рівня складності.

Всі дані про результати тестування, а також інформація про навчальний заклад, викладачів та їх навантаження, учнів, завдання до модулів тощо зберігається в базі даних Interbase. Дана база належить до архітектури клієнт-сервер, характерною особливістю якої є перенесення обчислювального навантаження на сервер бази даних (SQL-сервер). Також до переваг даної архітектури можна віднести:

- SQL-сервер Interbase призначений для зберігання та обробки великих об'ємів інформації в умовах одночасної роботи с базою даних множини клієнтських додатків;

- більшість обчислювальних процесів проходить на сервері, що знижує вимоги до потужності комп'ютера клієнту;

- така архітектура значно підвищує безпеку даних, тому що програма клієнта не має можливості працювати безпосередньо з даними, а робить це через - SQL-сервер;

- програмний комплекс працює в мережі Internet, тобто може бути використаний для дистанційного контролю знань.

Нами було розроблено та апробовано тестові завдання з таких загальноосвітніх предметів, як інформатика (за експериментальним підручником Жалдака М.І та Морзе Н.В.), фізики для 7-8

класів. Опис комплексу взагалі та методичні рекомендації до складання тестових завдань буде видано окремим посібником.

Програмно-педагогічний засіб пройшов апробацію в більш ніж десяти школах Запорізької області та отримав гриф “Рекомендовано Міністерством освіти та науки України”.

Література

1. Анастаси А. Психологическое тестирование: Книга 1. Пер. с англ. / Под ред. К.М. Гуревича – М.: Педагогика, 1982. – 320 с.
2. Фаронов В.В., Шумаков П.В. Delphi 5. Руководство разработчика баз данных – М.: Нолидж, 2000. – 640 с.
3. Методичні рекомендації для студентів 4 курсу з виконання та оформлення результатів практики з розробки програмних педагогічних засобів. Навчальний посібник. – Мелітополь: Маримакс, 2001. – 48 с.

ПРОБЛЕМИ ВИКЛАДАННЯ ІНФОРМАТИКИ У ВИЩИХ НАВЧАЛЬНИХ ЗАКЛАДАХ

Ю.В. Шмиголь

м. Полтава, Полтавська державна аграрна академія

JuliaS79@ukr.net

Незважаючи на бурхливе впровадження інформатики в наше життя, досі відсутній єдиний підхід до означення і структуризації цієї науки.

Формально в нашій країні інформаційні технології очолюють перелік пріоритетних напрямків науково-технічного розвитку країни. Інформатика як обов'язкова дисципліна вивчається в середній і вищій школі незалежно від напрямку підготовки спеціалістів. У той же час інформатиці дуже важко завойовувати рівноправне місце серед провідних фундаментальних наукових напрямків.

Інформатизація повинна стимулювати подальше вдосконалення самої технології та організації освітнього процесу. Сьогодні уже реальністю стали такі нові освітні технології, як електронні підручники, електронні факультети, дистанційне навчання тощо.

Швидке удосконалення комп'ютерів і програмних засобів, інформаційних технологій приводить до нових шляхів розвитку дисципліни "Інформатика" і ставить перед системою освіти нові задачі:

- перегляд цілей і змісту курсу "Інформатика";
- розробка диференційованих критеріїв оцінювання [4];
- аналіз якості підготовки випускників вузів до життя в інформаційному суспільстві.

Турбує спеціалістів і проблема ролі і місця інформатики як наукової і навчальної дисципліни. Зараз відбувається процес неухильного росту світоглядної ролі інформатики, росту її інтегруючої ролі по відношенню до багатьох наукових та науково-технічних дисциплін і напрямків.

Підготовка сучасних спеціалістів формується на фоні активної інформатизації суспільства та освіти всіх рівнів. В умовах масового поширення сучасних інформаційних та телекомуніка-

ційних технологій стає актуальним питання формування інформаційної культури широкого кола користувачів.

Інформатика як навчальний предмет включена в навчальні плани усіх сучасних спеціальностей. Основна мета цього предмету – навчити молодого спеціаліста основам використання персонального комп'ютера у своїй професійній діяльності для того, щоб полегшити виконання завдань та зменшити затрати часу і коштів. Саме комп'ютер дозволяє розвинути у студента самостійність роботи і мислення. Він надає широкі можливості для розвитку творчих здібностей, для підготовки професійного трудового потенціалу [5].

Основні напрямки застосування інформатики у вищій школі:

- 1) моделювання різноманітних процесів та явищ;
- 2) автоматизація проектування пристроїв і систем;
- 3) автоматизовані навчальні курси;
- 4) інформаційно-довідкова служба, інформаційна підтримка освітнього процесу;
- 5) розрахунок, обробка результатів вимірювань і експериментальних досліджень;
- 6) дистанційна освіта;
- 7) педагогічна інформатика;
- 8) електронні підручники і, як наслідок, електронні бібліотеки;
- 9) телекомунікаційні технології [3].

Найголовніша проблема зараз при викладанні інформатики у вищих навчальних закладах – це різний початковий рівень студентів-першокурсників. Практика показує, що, крім цього, перехід більшості учорашніх школярів від класно-урочної системи до переважно самостійних занять часто є досить складним і супроводжується великими труднощами. Студенти-першокурсники часто не вміють перебудувати навчання у відповідності до нових умов. А це сприяє виникненню негативного ставлення до освіти в цілому [1].

Враховуючи те, що зараз в школах викладаються як машинні, так і без машинні курси, то студенти мають і різні практичні навички та теоретичні знання. У школах України зустрічається дуже різна техніка – від Корветів, БК до найостанніших моделей комп'ютерів класу Pentium IV. Проблема загострилася і у зв'язку

з тим, що з'явилося багато шкіл нового типу, які вивчають поглиблено один або кілька предметів одного напрямку. Тому нерідко серед першокурсників зустрічаються такі, які володіють знаннями з окремих розділів програми вузу [1].

Саме тому викладачу вищого навчального закладу дуже важко проводити як лекції, так і лабораторно-практичні заняття. У цьому випадку просто необхідно застосовувати диференційований підхід до викладання даної дисципліни. Це один із найважливіших моментів, на які повинні звернути свою увагу викладачі інформатики. Це стосується як теоретичного матеріалу, так і індивідуальних завдань на лабораторно-практичні заняття, для самостійної роботи. Для того, щоб максимально диференціювати рівень знань, у перелік завдань необхідно включити питання різного рівня складності, які б дозволяли виявити у студента максимум його здібностей та нахилів. Основою диференціації при викладанні дисципліни “Інформатика та комп'ютерна техніка” і супутніх їй предметів, є поняття індивідуальності – неповторності людини. Індивідуалізація навчання – це організація навчального процесу, в якому вибір способів, прийомів, темпу навчання; враховує індивідуальні відмінності студентів, рівень знань, розвиток їх здібностей. Диференціація в нашому розумінні – це врахування індивідуальної своєрідності студентів і їх групування на основі особистісних якостей для окремого навчання (наприклад, за різними навчальними планами і програмами). Саме такий напрямок викладання дозволить покращити рівень знань, умінь та навичок студентів.

Необхідно застосовувати і принцип адитивності, за яким при пристосуванні персонального комп'ютера до індивідуальних особливостей студента є можливість вести діалог зі студентом; корегувати процес навчання; створювати стан психологічного комфорту, що веде до впевненості студента в собі, своїх знаннях [2].

Для того, щоб студент після закінчення вищого навчального закладу мав визначений рівень інформаційної культури, потрібна неперервна комп'ютерна освіта, оскільки науково-технічний прогрес не стоїть на місці. Цю функцію виконують поряд з базовим курсом інформатики і курси на вибір (студента чи вузу) та спеціальні дисципліни.

Література

1. Баловсяк Н.В. Диференційований підхід до розробки планів вивчення предмета студентами з різним рівнем шкільної підготовки на першому курсі вузу (на прикладі курсу „Інформатика та комп’ютерна техніка”). // Регіональні Перспективи. – 2001. – № 1. – С. 41-44.
2. Стороженко В.О., Джурка Г.Ф., Шиян Н.І. Педагогічні можливості використання комп’ютерних технологій у вищій школі. // Інформаційні технології навчання у вищих закладах освіти: Збірник матеріалів. / Кол. авт. – Суми: Вид-во СумДУ, 2001. – Част. 2. – С. 138-141.
3. Ткачук В. Інформаційні технології в системі вищої освіти: етапи впровадження. // Вища освіта України. – 2002. – №4. – С. 43-47.
4. Коваленко О.О. Методика викладання дисципліни “Економічна інформатика” // Регіональні Перспективи. – 2001. – № 1. – С. 39-41.
5. Лігум Ю.С. Роль економіки та інформатики в підготовці сучасного спеціаліста. // Нові технології навчання: Наук.-метод. зб. / Ред. кол.: Б.І. Холод (головний редактор), О.Я. Савченко, О.І. Ляшенко, А.М. Федяєва та ін. – К.: НМЦВО, 2000. – Вип. 27. – С. 27-32.

Зміст

<i>О.В. Бабиц.</i> Вивчення основ SQL в курсі «Бази даних та інформаційні системи».....	3
<i>В.Ю. Баранов.</i> Комп'ютерне тестування з інформатики: підсумки педагогічного експерименту в Одеській національній академії зв'язку.....	6
<i>Е.В. Белова, А.С. Мустафа, Д.А. Фадеєв.</i> Использование информационных технологий в обучении информатике.....	13
<i>М.А. Бондаренко, О.С. Иванченко.</i> Електронний підручник “Програмування на Object Pascal у системі об'єктно-орієнтованого програмування Delphi 6”.....	18
<i>Л.Л. Вакуленко.</i> Роль інформаційної системи в удосконаленні регулювання імпорتنих операцій в Україні.....	22
<i>К.Ю. Васильєв, А.В. Кривил'єв.</i> Изучение языка Simnet II в рамках деятельностного подхода.....	26
<i>И.Н. Вдовиченко.</i> Общие тенденции развития экспертных систем в исследовании учебного процесса.....	29
<i>В.В. Войтенко, А.В. Морозов.</i> Сучасний системний підхід до викладання основ програмування.....	34
<i>Т.В. Волкова.</i> Підготовка інженерів-педагогів в університеті.....	37
<i>Л.П. Воронеж, В.А. Сергієнко.</i> Зміст та завдання курсу “Інформатика та комп'ютерна техніка” для студентів аграрних закладів освіти III-IV рівнів акредитації.....	46
<i>Ю.О. Дорошенко, Н.В. Семенюк.</i> Програмне забезпечення навчального призначення з екології людини.....	52
<i>О.М. Дресєв, З.Ю. Філер.</i> Спектральний аналіз майже періодичних сигналів.....	64
<i>В.Н. Евтєєв.</i> Блеск и нищета вербального тренинга.....	71
<i>О.В. Ємець, К.М. Скиба, В.М. Шепеленко, С.В. Марченко, Т.О. Рудик.</i> Методика розпізнання Past Indefinit і Participle II для систем машинного перекладу.....	76
<i>Н.С. Завізєна.</i> Інформаційні системи при вивченні дисциплін економічного профілю.....	81
<i>Н.С. Завізєна, Е.Г. Гуржий.</i> Внедрение тестовых программ в учебный процесс.....	85

<i>О.Ф. Клименко, Ю.М. Красюк, Л.М. Петренко, В.В. Соловйова.</i> З досвіду викладання курсу інформатики для студентів економічних спеціальностей.....	90
<i>О.В. Клочко.</i> Лабораторні роботи як метод активізації пізнавальної діяльності студентів з дисциплін комп'ютерного циклу у вищих аграрних навчальних закладах.....	92
<i>Е.А. Кожаев, Г.Г. Маклакова, Г.Ю. Маклаков.</i> Специализированный аппаратно-программный комплекс для чтения электронных книг.....	99
<i>И.А. Колпаков, В.Г. Рябцев.</i> Метод тестирования цифровых систем со встроенной памятью.....	104
<i>Ю.М. Красюк.</i> Використання рейтингової системи контролю знань, умінь та навичок студентів при навчанні інформатики.....	109
<i>І.В. Крашеніннік.</i> Теоретичні та практичні засади впровадження педагогічної технології “створення ситуації успіху” в процес викладання дисципліни “Інформатика та комп'ютерна техніка” у вищому навчальному закладі.....	118
<i>И.А. Кривель, А.Н. Моргун.</i> Некоторые аспекты преподавания основ алгоритмизации и программирования.....	125
<i>С.В. Кукліна, Н.В. Моїсеєнко.</i> Електронний посібник з курсу “Основи захисту інформації”.....	130
<i>Г.И. Кулик.</i> О повышении уровня компьютерного образования студентов технических специальностей.....	135
<i>Л.Л. Леоненко.</i> Оцінювання відповідей “відкритого” типу за умов комп'ютерного тестування знань.....	138
<i>Н.А. Леонова, О.А. Хараджян.</i> Курс “Цифрові ЕОМ” в педагогічному ВНЗ.....	144
<i>Р.В. Лихачов.</i> Декілька питань про методи викладання ознайомчих курсів інформатики у ВНЗ (частина друга).....	149
<i>С.С. Лихман.</i> Особенности преподавания дисциплины «Микропроцессорные устройства».....	153
<i>В.Г. Логвиненко.</i> Метод априорной оценки информативности факторов, формирующих познавательную самостоятельность студентов при изучении дисциплин компьютерного цикла.....	155

<i>Ю.Г. Лотюк, О.Є. Щодро.</i> Формування дослідницьких вмінь і навичок у студентів на основі вивчення моделей механічних систем у середовищі Simulink	161
<i>І.М. Лукаш.</i> Застосування дослідницького методу у процесі навчання інформатики.....	166
<i>Г.Ю. Маклаков.</i> Методы оценки влияния современных компьютерных технологий на человека.....	171
<i>Г.Ю. Маклаков, Е.А. Кожаяев, Г.Г. Маклакова.</i> Перспективы использования современных микроконтроллеров для создания портативных устройств мультимедийного отображения учебной информации.....	176
<i>Ю.С. Матвієнко.</i> Реалізація алгоритмів математичних обчислень в системі MathCAD	180
<i>О.А. Медведева.</i> Работа с финансовыми функциями Excel	185
<i>А.Ю. Мельников.</i> Методика преподавания компьютерных дисциплин студентам специальности «Экономическая кибернетика».....	190
<i>О.І. Миронова.</i> Самостійна робота студентів в умовах системної інтеграції інформаційних технологій	193
<i>А.А. Мясищев, В.М. Полозова.</i> Анализ целесообразности использования Squid для доступа в Интернет	200
<i>А.А. Мясищев, В.М. Полозова.</i> Ознакомление с командами POP3-сервера при изучении дисциплины «Компьютерные сети» для получения e-mail.....	210
<i>Я.Л. Напреев.</i> Створення та опублікування навчальних програмних продуктів	214
<i>С.М. Овчаров.</i> Формування педагогічної спрямованості навичок програмування майбутніх учителів інформатики	218
<i>О.Г. Онишко.</i> Підручник як один із основних дидактичних засобів розвитку творчого потенціалу студентів під час навчання інформатики	222
<i>М.Е. Отставнов.</i> Введение в открытые операционные системы.....	228
<i>Е.А. Панина, Л.В. Гусева.</i> Создание и статистическая обработка психологических тестов средствами ТП Excel.....	236
<i>О.В. Пахомова.</i> Деякі аспекти навчання комп'ютерної лексики на кібернетичних спеціальностях (на матеріалі англійської мови)	238

<i>Ю.М. Пилипенко, Л.П. Голубєв.</i> Знайомство з ОС Windows та MS-Word у ході вивчення інформатики	241
<i>А.П. Полищук, С.А. Семериков.</i> О реализации практикума по программированию лексических и синтаксических анализаторов при создании языковых интерпретаторов	250
<i>Н.І. Правороська.</i> Мотиваційна функція системи задач у навчанні інформатики	260
<i>П.А. Ротаєнко, Н.І. Самойленко, Л.П. Семко.</i> Про основні питання комп'ютерного контролю знань, умінь і навичок ...	269
<i>О.Ю. Рудик, К.М. Скиба, Т.О. Рудик, Н.В. Райко.</i> Методологія застосування інформаційних технологій у студентській науковій роботі з іноземної мови.....	276
<i>Г.И. Сажко, А.Н. Майданович.</i> Формирование эргономических знаний у студентов инженерно-педагогических специальностей.....	280
<i>В.Д. Сквирский, Н.А. Семенов.</i> Особенности изучения аппаратных средств персональных компьютеров	285
<i>Е.И. Скворцова.</i> Решение трансцендентных уравнений методом Ньютона при изучении табличного процессора Excel	290
<i>А.Ю. Соколов, Г.Н. Жолткевич, И.Т. Зарецкая, О.В. Яровая.</i> Информатика для инженерных специальностей технических университетов	297
<i>А.Ю. Соколов, И.А. Трофимова.</i> Тестирующий программный комплекс.....	302
<i>А.М. Стрюк, М.С. Стрюк.</i> Використання віртуальних лабораторій при вивченні курсу “Теорія операційних систем”	305
<i>Ю.В. Триус.</i> Мережевий навчально-контролюючий комплекс з математичних основ інформатики.....	310
<i>О.В. Тумашова, О.М. Коссак.</i> Особливості методики викладання методів наближених обчислень для студентів технічних спеціальностей.....	320
<i>О.Г. Тюрін.</i> Використання методу множників при вивченні циклів.....	322
<i>І.Є. Фільо.</i> Електронні системи контролю знань та засоби для їх створення	328
<i>Н.А. Хміль, С.В. Дяченко.</i> Нові інформаційні технології у вищій школі: проблеми, актуальність	333

<i>С.В. Шаров.</i> Розробка програмного комплексу тематичного оцінювання знань учнів.....	337
<i>Ю.В. Шмиголь.</i> Проблеми викладання інформатики у вищих навчальних закладах	342

Наукове видання

**Теорія та методика навчання
математики, фізики, інформатики**

Випуск 4

В 3-х томах

Том 3

Підп. до друку 03.03.2004
Папір офсетний №1
Ум. друк. арк. 18,5

Формат 80×84 1/16
Зам. №1-0303
Наклад 300 прим.

Жовтнева друкарня
50014, м. Кривий Ріг-14, вул. Електрична, 5
Тел. (0564) 664381

E-mail: cc@kpi.dp.ua