

95.2  
И 741

ISSN 0234—0453

# ИНФОРМАТИКА И ОБРАЗОВАНИЕ

6 1990

ТЭКС





*Что такое компьютер в сознании ребенка?  
Открывает ли общение с ним новые возможности  
для развития детей?  
Как меняются роль учителя и его поведение  
во время компьютерных уроков?*

*На эти и многие другие вопросы должен дать ответ  
педагогический эксперимент, который проводится  
в московской школе № 1158  
(статья «На подступах к компьютерной педагогике»)  
На фото: ученики экспериментальной школы № 1158*



# ИНФОРМАТИКА И ОБРАЗОВАНИЕ

OldPC.su

7003

музей компьютеров

## Содержание

### Методика обучения

Гольцман М., Дуванов А., Зайдельман Я., Первин Ю. Арифметические исполнители	3
Ватник Н., Куприянов С., Лобанова Л., Пахомов Ю. Опыт разработки и использования УАОС	15
Деятельность младших школьников в компьютерной игровой учебной среде	19
Каймин В. Курс информатики: состояние, методика и перспективы	26

### Кабинет ВТ

Гуткин М., Иванов А., Новосельцев С., Христочевский С. Учебные персональные ЭВМ	34
Григорьев С. Графические средства системы Пролог-Д	47
Тищенко В. Настройка простейшего компьютера	50
Кузнецов А. Графический планшет	53
Артамонов В. Музыка КУВТ «Корвет»	54
Еремин Е. Улучшение Турбо-Паскаля для «Ямахи»	58
Пустовойтов С. Об одной особенности КУВТ-86	58

### Клуб БК

Зальцман Ю. Архитектура и ассемблер БК-0010	61
Квазиподпрограммы	70
Спасение программ при зависании	71
Подключение БК-0010 к телевизорам 4УСЦТ	72
Новый подход к построению баз данных на БК-0010	72
Полку периферии прибыло	73

### Педагогический опыт

Кузьмин Ю., Рибуде И., Кондратенко С. Использование ЭВМ при обучении русскому языку	74
Гварамия Г., Маргвелашвили И., Мосиашвили Л. Опыт разработки компьютерных учебных пособий по физике	79
На подступах к компьютерной педагогике	82

## Внеклассная работа

- Переход И., Касаткин В. Комбинаторные задачи 90  
Белоножко О., Юдилевич А. Психологическая диагностика детей и компьютер 91

## НИТ в дошкольном образовании

- Зворыгина Е. Педагогические подходы к компьютерным играм для дошкольников 94  
Глушкова Е., Леонова Л., Сазанюк З., Степанова М. Гигиенические требования к занятиям для дошкольников 102

## Нам пишут

- О программах для «Корвета» 105  
Чтобы не бегать по классу 106  
Ищем спонсоров 107  
Письмо и комментарий 107

## Информация

- Ассоциация учителей информатики проводит конкурсы 111  
Международный семинар «Образование и информатика» 113  
Государственный реестр программ 115  
Первый раз в «пилотный» класс 116  
«ИНФО» в новом году 117

## Библиография

- По страницам тематических планов издательств на 1991 г. 118  
Новые журналы 121  
Напечатано в 1990 г. 125

Главный редактор  
академик

**В. А. МЕЛЬНИКОВ**

Редакционная  
коллегия

**И. Н. АНТИПОВ**

**В. Н. АФАНАСЬЕВ**

**И. М. БОБКО**

**Г. В. ГОДЖЕЛЛО**

**С. А. ЖДАНОВ**

**Б. В. ЛОМОВ**

**Ю. В. ЛУИЗО**

(зам. главного  
редактора)

**Н. Г. МЕЛЬДИАНОВ**

**И. С. ОРЕШКОВ**

**О. К. ПАВЛОВА**

**А. Ю. УВАРОВ**

**А. И. ФУРСЕНКО**

**В. О. ХОРОШИЛОВ**

**К. В. ШЕХОВЦЕВ**

(редактор отдела)

В оформлении номера принимали участие: Э. Бажилин, Ю. Са-  
рафанов, В. Соломонов.

Редактор отдела *А. Кравцова*

Научный редактор *Н. Копытина*

Зав. редакцией *Н. Игнатова*

Художественный редактор *Л. Коновалова*

Корректор *Л. Яковлева*

Сдано в набор 21.09.90. Подписано в печать 6.11.90.

Формат 70×100 1/16 Бумага офсетная № 1. Печать офсетная.

Усл. печ. л. 10,40. Усл. кр.-отт. 42,88. Уч.-изд. л. 12,90.

Тираж 72 570 экз. Заказ 1923. Цена 60 коп.

Издательство «Педагогика» Академии педагогических наук СССР  
и Государственного комитета СССР по печати.

*Адрес для переписки:* 119034, Москва, Смоленский бульвар, д. 4.

Адрес редакции: Лефортовский пер., д. 8.

Телефон редакции: 261-11-29.

Ордена Трудового Красного Знамени Чеховский полиграфический комбинат  
Государственного комитета СССР по печати.  
142300, г. Чехов, Московской обл.

© «Педагогика», «Информатика и образование», 1990.

М. ГОЛЬЦМАН, А. ДУВАНОВ, Я. ЗАЙДЕЛЬМАН, Ю. ПЕРВИН

## Арифметические исполнители

Одна из доступных детям и формализованных предметных сред — арифметические выражения. Поэтому фундаментальное понятие исполнителя важно показать именно в этой среде. Арифметические исполнители дают возможность наглядно и эффективно ввести еще одно существенное понятие — параметр команды.

Арифметические исполнители — это еще один методический узел, в котором перекрещиваются педагогические направления Роботландии: алгоритмическое и компьютерное.

Урок 11.6. Тема: «Исполнитель Автомат»

План урока:

Проверка домашнего задания.

Повторение: анализ известных программ как исполнителей.

Общие свойства исполнителей, реакции на ошибки.

Исполнители как ситуации-модели.

Числовой конструктор.

Исполнитель Автомат.

Лабораторная работа: Автомат.

Домашнее задание: § 4.9; составить алгоритм работы исполнителя Автомат для получения заданного числа.

Детям предлагается простой исполнитель Автомат, подробно описанный в § 4.9.

### § 4.9. Исполнитель Автомат

#### СРЕДА

Среда исполнителя Автомат состоит из экрана, на котором высвечиваются трехзначные числа, и пульта управления с тремя кнопками. Исполнитель работает с числовой информацией (см. рис. 1).



#### КОМАНДЫ

СКИ Автомата  
СБРОСЬ  
ПРИБАВЬ  
УМНОЖЬ

#### КАК ОТДАЮТСЯ

Чтобы отдать команду Автомату, нужно нажать соответствующую кнопку на пульте управления.

#### КАК ВЫПОЛНЯЮТСЯ

**СБРОСЬ** — на экране Автомата устанавливается число 0.

**ПРИБАВЬ** — к числу на экране прибавляется один.

**УМНОЖЬ** — число на экране умножается на два.

#### КОГДА НЕ МОГУ

Исполнитель попадает в ситуацию НЕ МОГУ при попытке получить на экране число, большее 999.

**Пример.** Получить на экране Автомата число 5 (см. табл. 1).

Таблица 1

Команда		На экране	
		до выполнения	после выполнения
СБРОСЬ	все, что угодно	000	
ПРИБАВЬ	000	001	
ПРИБАВЬ	001	002	
УМНОЖЬ	002	004	
ПРИБАВЬ	004	005	

**Вопросы и упражнения.**

1. Используя запись решения примера в качестве образца, получите на экране Автомата числа 11, 21, 99.

4 Рассказ об Автомате лучше вести, вызвав на экран заставку исполнителя. Тогда школьникам легко понять, почему этот исполнитель не выдает сообщения НЕ ПОНИМАЮ: обсуждая схему знакомства с Автоматом дети видят, что способ задания команд — это нажатие кнопок на панели Автомата. Этот способ не позволяет задать Автомату команду, не входящую в его СКИ. Следовательно, сообщение НЕ ПОНИМАЮ здесь никогда не появится.

А вот сообщение НЕ МОГУ появиться может. Хорошо, если эту возможность дети обнаружат сами. Сообщение НЕ МОГУ появляется при переполнении (именно это слово надо обязательно произнести!). Поскольку дети со слов учителя знают, что на экране можно записать только трехзначные числа, они отвечают на вопрос о самом большом допустимом числе (999) и самом малом числе, не помещающемся на экране. И тогда уже практически готов ответ на вопрос о том, как возникает сообщение НЕ МОГУ.

Рассказ учителя об Автомате сопровождается простым примером на демонстрационном экране. Например, получение числа 13:

- СБРОСЬ ( 0)
- ПРИБАВЬ ( 1)
- УМНОЖЬ ( 2)
- ПРИБАВЬ ( 3)
- УМНОЖЬ ( 6)
- УМНОЖЬ (12)
- ПРИБАВЬ (13)

Алгоритмы, выполняемые Автоматом, очень просты: любой из них порождает возрастающую последовательность чисел. Учитель может проверить, насколько понятен детям Автомат, если попросит их ответить на вопросы:

«Какой командой заканчивается выполняемый Автоматом алгоритм, если требуется получить нечетное число?» (правильный ответ однозначен — «только командой ПРИБАВЬ»);

«В каком случае действие двух команд ПРИБАВЬ дает такой же результат, что и выполнение одной команды УМНОЖЬ?» (здесь ответ однозначен, но более труден: когда на экране число 2).

**Урок 11.7. Тема:** «Решение задач на исполнителе Автомат»

**План урока:**

Игра в отгадывание алгоритма.

Проверка домашнего задания: исполнитель Автомат; лабораторная работа.

Обсуждение оптимального алгоритма работы Автомата.

Лабораторная работа: решение задач на Автомате по оптимальному алгоритму.

Домашнее задание: § 4.10 с упражнениями.

Примеры домашнего задания (из § 4.9) проверяются на компьютере. Дети выполняют работу индивидуально, перенося на экран команды, записанные в домашней тетради. Так же как и на прошлом уроке, можно пока не требовать от учащихся оптимальности алгоритма.

Отметив разброс количества команд в алгоритмах, получающих одно и то же число, учитель ставит новую задачу: получить на экране Автомата некоторое число, например 57, за наименьшее число шагов.

Метод решения подробно разобран в § 4.10. На уроке таблицу оптимального решения учитель строит вместе с учениками. При этом построении таблица растет естественным образом вниз, но затем нумерация команд осуществляется в обратном порядке снизу вверх. Нумерация и определяет порядок, в котором детям предстоит набирать команды на компьютере. Следует заметить: что было на экране перед командой СБРОСЬ, не важно, ведь ее результатом всегда будет нуль.

**§ 4.10. Как правильно работать с Автоматом**

— Ты сделала задачу про Автомат? — спросил Петя Кук Катю Пушкинову перед уроком информатики.

— Да, но мое решение длинное. Число 21 у меня получается за 11 команд. За 3 команды — СБРОСЬ, ПРИБАВЬ, ПРИБАВЬ — я получила число 2. Потом решила, что, так как команда УМНОЖЬ «быстрее» команды ПРИБАВЬ, лучше использовать ее. У меня получились еще три команды (см. табл. 2).

Таблица 2

Команда	На экране	
	до выполнения	после выполнения
УМНОЖЬ	002	004
УМНОЖЬ	004	008
УМНОЖЬ	008	016

Если еще раз выдать команду УМНОЖЬ, на экране получится уже 32, а это больше 21. Значит, теперь придется применять команду ПРИБАВЬ пять раз, чтобы получить 21 из 16.

— Да, твое решение длинное, мое состоит всего из 9 команд,— сказал Петя и показал свое решение (см. табл. 3).

Таблица 3

Команда	На экране	
	до выполнения	после выполнения
1. СБРОСЬ		000
2. ПРИБАВЬ	000	001
3. ПРИБАВЬ	001	002
4. УМНОЖЬ	002	004
5. УМНОЖЬ	004	008
6. ПРИБАВЬ	008	009
7. ПРИБАВЬ	009	010
8. УМНОЖЬ	010	020
9. ПРИБАВЬ	020	021

— Чтобы построить самое короткое решение,— сказал на уроке учитель,— надо начать с конца.

Допустим, что на экране Автомата уже получено число 21. Какой командой это число могло быть получено? Только командой ПРИБАВЬ (см. табл. 4).

Таблица 4

Что на экране есть	Какой командой получено	Что на экране было
021	ПРИБАВЬ	020

Теперь на экране число 20. Как оно было получено? Так как мы стремимся сократить число команд, то самое разумное — получить его «быстрой»

командой УМНОЖЬ из числа 10 (см. табл. 5).

Таблица 5

Что на экране есть	Какой командой получено	Что на экране было
020	УМНОЖЬ	010

Число 10 можно за одну команду УМНОЖЬ получить из числа 5. Так рассуждая, мы наконец добираемся до числа 0. Теперь остается только занумеровать команды в том порядке, в котором они должны отдаваться Автомату, и решение готово (см. табл. 6).

Таблица 6

Что на экране есть	Какой командой получено	Что на экране было	Номер команды
021	ПРИБАВЬ	020	8
020	УМНОЖЬ	010	7
010	УМНОЖЬ	005	6
005	ПРИБАВЬ	004	5
004	УМНОЖЬ	002	4
002	УМНОЖЬ	001	3
001	ПРИБАВЬ	000	2
000	СБРОСЬ		1

Как видите, самое короткое решение содержит 8 команд.

Урок 11.8. Тема: «Исполнители Вычислители»

План урока:

Обсуждение домашнего задания.

Простейший вычислитель.

Вычислитель с памятью.

Игры с черными ящиками.

Домашнее задание: составить алгоритмы для Вычислителя-2; придумать алгоритмы для игры с черными ящиками.

Рассмотрение задач, приводящих к понятию стека, делается в три этапа.

На первом этапе учитель предлагает детям придумать исполнителя, который будет выполнять арифметические операции над любыми двумя числами (а не только простейшие сложения и умножения, как в Автомате). СКИ такого исполнителя формируется довольно быстро. Учителю, возможно, надо будет лишь немного корректировать предложения детей, чтобы прийти к таким названиям команд:

СЛОЖИ  
ВЫЧТИ  
УМНОЖЬ  
ДЕЛИ

Очень важно усвоить, что любая из перечисленных команд станет понятной исполнителю только после того, как вслед за названием команды будут названы два числа, над которыми надо выполнить соответствующую арифметическую операцию. Например, команда сложения может иметь вид: СЛОЖИ 5 7 или СЛОЖИ 12 13.

Числа, указанные вслед за командой, называются параметрами этой команды. Таким образом, каждая из команд нашего исполнителя — это команда с двумя параметрами. Несмотря на то что количество всевозможных пар чисел, которые могли бы фигурировать в командах, бесконечно, понятие параметра дает возможность сохранить одно из важнейших свойств каждого исполнителя — ограниченность СКИ.

6 Понятие параметра, вообще говоря, заставляет пересмотреть взгляд на системы команд некоторых ранее изученных исполнителей. Например, вместо 6 команд в системе команд Ханойской башни можно рассматривать СКИ из единственной команды с параметрами.

Теперь можно полностью рассмотреть исполнителя Вычислителя по схеме знакомства. СКИ уже построена:

СКИ ВЫЧИСЛИТЕЛЯ	
СЛОЖИ $x$ $y$	(означает $x + y$ )
ВЫЧТИ $x$ $y$	(означает $x - y$ )
УМНОЖЬ $x$ $y$	(означает $x \cdot y$ )
ДЕЛИ $x$ $y$	(означает $x : y$ )

Создавая среду Вычислителя, важно отметить, что он имеет счетное устройство (процессор) с тремя окошками. Одно служит для ввода первого параметра команды, другое — для второго, а в третьем Вычислитель показывает результат операции.

Как обычно, предлагается поиграть в новый исполнитель. Для игры потребуются два артиста. Один из них — это исполнитель Вычислитель. Другой — Процессор, он представляет собой элемент среды. Вычислитель, получив от класса команду, может ответить НЕ ПОНИМАЮ (если команды нет в СКИ) или отдать Процессору параметры и указание для выполнения операции. Ученик, изображающий Процессор, получает листочек с написанным на нем первым параметром в левую руку, со вторым — в правую, а о результате сообщает Вычислителю. Учитель может заранее подготовить карточки с числами от 1 до 20 (для простейшей игры этого диапазона будет достаточно). Процессор может отрицательно покачать головой. Вычислитель понимает это покачивание как сообщение о невозможности выполнить операцию и произносит НЕ МОГУ.

Второй этап приближения к стеку начинается с небольшого усложнения задачи: надо вычислить  $2 \cdot 3 + 4$ .

Очень просто выполняется первое действие — умножение  $2 \cdot 3$ . Но чтобы выполнить второе действие — сложение, результат первого действия надо помнить. Конечно, если роль исполнителя играет ученик, он может запомнить число 6. Впрочем, если придется запоминать, например, результат произведения  $245 \cdot 117$ , задача становится труднее.



Вывод: если приходится делать не одно, а два действия подряд, надо заставить исполнителя запоминать промежуточный результат. Появляется исполнитель с памятью. Назовем его Вычислитель-2. Существенно, что речь идет о простейшем запоминающем устройстве, т. е. памяти, хранящей всего одно число. Вместе с появлением памяти должна появиться и команда, загружающая число в память.

СКИ Вычислителя-2	
ЗАПОМНИ $x$	
СЛОЖИ $x$	
ВЫЧТИ $x$	
УМНОЖЬ $x$	
ДЕЛИ $x$	

По команде СЛОЖИ 5 (заметьте, с одним параметром!) Вычислитель-2 берет число из памяти (результат предыдущей операции), складывает его с указанным в команде параметром (число 5) и отправляет результат в память. В памяти теперь будет храниться новый результат, вытеснивший записанное там ранее число.

Точно так же выполняются команды УМНОЖЬ 3 (утроение хранимого в памяти числа), ДЕЛИ 4 (уменьшение числа, хранившегося в памяти, в четыре раза).

Имея такую СКИ, можно записать задание исполнителю для решения примера, с ко-



торого началось обсуждение второго этапа:

**ЗАПОМНИ 2**  
**УМНОЖЬ 3**  
**СЛОЖИ 4**

У этого исполнителя то же самое вычислительное устройство. Однако в его среде дополнительно появляется память.

В Вычислитель-2 тоже надо сыграть, по крайней мере, один-два раза.

Прежде чем перейти к черным ящикам, учитель формулирует домашнее задание — составить алгоритмы для решения следующих примеров с помощью Вычислителя-2:

$$(23+49) \cdot 15;$$
$$17 \cdot 12 - 18;$$
$$2 \cdot 3 + 4 \cdot 5.$$

В действительности на Вычислителе-2 успешно решаются только первые два примера. Для третьего потребуется сделать запоминание промежуточного результата. С этого трудного момента начнется следующий урок.

**Урок 11.9. Тема: «Стек, действия на стеке»**

**План урока:**

Игра с черными ящиками.

Проверка заданий для Вычислителя-2.

Стек, примеры применений.

Глубина стека, верхушка стека.

Особенности стековых арифметических операций.

Лабораторная работа: исполнитель Плюрик.

Домашнее задание: § 4.11; нарисовать стек, изобразить в рисунках выполнение операции вычитания.

Игра в Вычислитель-2 по первым двум примерам домашнего задания проходит по сценарию прошлого урока. В третьем примере выясняется, что Вычислителю-2 работать трудно. Вычисляя выражение  $2 \cdot 3 + 4 \cdot 5$ , надо запоминать не одно число, а два. А ведь бывают примеры и гораздо труднее! Значит, нужен исполнитель, который будет готов запоминать два числа и более. Вот тут-то и появляется стек.

Учитель может использовать материал учебника для изложения принципов работы стека. Следует продемонстрировать достаточно наглядную модель стека. Например, стеклянная мензурка, в которую закладываются мячики для настольного тенниса (рис. 2). Для этой демонстрации мячики важно индивидуализировать: выкрасить в разные цвета или начертить на них разные цифры.

Исполнитель Вычислитель-3, выполняющий арифметические операции и имеющий память, которая организована в виде стека, имеет такую СКИ:

СКИ Вычислителя-3
ЗАПОМНИ x
СЛОЖИ
ВЫЧТИ
УМНОЖЬ
ДЕЛИ

Только одна команда в составе этой СКИ **7** имеет параметр. Это команда загрузки числа в стек. Все команды арифметики не имеют параметров. Параметры оказываются ненужными, поскольку исполнитель устроен так, что оба числа, участвующие в операции, он должен взять из стека и результат этой операции снова запомнить в стеке.

На уроках информатики было уже несколько встреч со стеком. Вспомните Машиниста. Железнодорожный тупик, куда Машинист загонял вагоны, — самый настоящий стек. Из стека-тупика можно взять самым первым только тот вагон, который в него был поставлен последним.

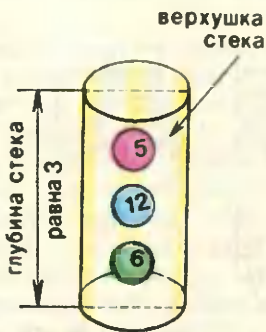
А еще раньше на экранах появились Ханойские башни. Каждая башня — это тоже стек. Когда требуется снять кольцо со стержня, для этого можно воспользоваться только последним из надетых на стержень колечек.

Стеки — это не только игрушки. Для решения многих задач память компьютеров часто организуют в виде стека.

Термин «стек» назван, но, по существу, еще не раскрыт. С этой целью определяется терминология, относящаяся к стеку: элемент стека, верхушка стека, — доступный элемент стека, глубина стека, количество элементов в стеке (обобщающий термин «элемент» можно пока не употреблять в беседе с детьми, поскольку сегодня и на ближайших уроках в качестве элементов стека будут фигурировать исключительно числа). Следует отметить, что стек представляет собой динамическую информационную структуру (детям об этом можно сказать словами: «Состояние стека постоянно изменяется в ходе обработки информации»).

Важно различать текущую и максималь-

2



ную глубину стека. Текущая глубина — это количество чисел в стеке в рассматриваемый момент времени, а максимальная глубина — это техническая характеристика исполнителя. Она определяет, сколько чисел способен исполнитель хранить в своем стеке.

У каждого конкретного исполнителя со стеком без труда определяется максимальная глубина. Так, максимальная глубина стека-мензурки — это число размещающихся в ней мячиков; у Машиниста — длина пути, измеряемая в количестве вагонов.

8 Рассказывается об арифметических операциях, выполняемых на стеке. Естественно первые примеры построить на операции сложения — тогда можно пока опустить обсуждение размещения слагаемых, упрощая тем самым изложение сложного механизма. Потом надо объяснить принципиальную существенность порядка операндов в стеке для операции вычитания (важный тезис: порядок операндов в вычитании — причина, порядок размещения их в стеке — следствие), попросить школьников объяснить, почему это ограничение несущественно для операции сложения (в силу коммутативного закона сложения). Аналогичные рассуждения об операциях умножения и деления дети могут провести самостоятельно (под управлением учителя).

Еще одна особенность арифметических операций, выполняемых на стеке, выражается в отсутствии параметров у арифметических команд исполнителя. Коль скоро число операндов всегда строго фиксированно, можно условиться, откуда исполнитель будет их извлекать, и потому не указывать, задавая операцию. Такое условленное место для операндов-параметров — верхушка стека. Сначала берется первый операнд (порядок, как только что объяснено, существенен). В освободившуюся верхушку поступает следующий элемент-число. Это число — второй (и последний!) операнд арифметической операции. Выполнив операцию, исполнитель посылает результат снова в верхушку стека. Таким образом, после каждой операции текущая глубина стека уменьшается на единицу: взяли из стека два элемента, а вернули один.

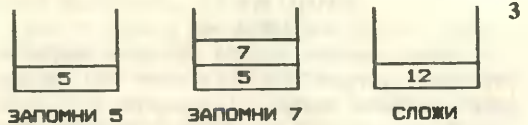
Объясняя операцию сложения, учитель рисует на кодоскопе состояние стека на каждом из этапов операции. Например, для сложения  $5+7$  задание исполнителю будет иметь вид:

**ЗАПОМНИ 5**  
**ЗАПОМНИ 7**  
**СЛОЖИ**

а соответствующий рисунок (рис. 3):

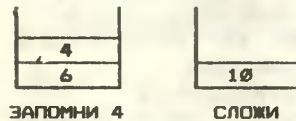
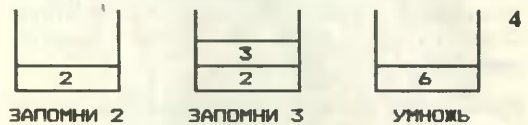
В домашнем задании предлагается по-

строить такую же серию рисунков для операции вычитания двух чисел.



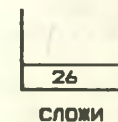
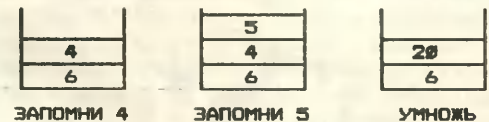
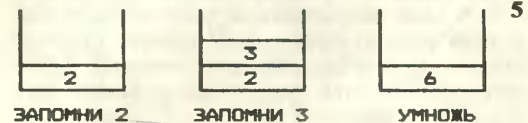
Приведенный здесь пример сложения, как нетрудно вспомнить, будет доступен и первому Вычислителю. Более того, там этот пример даже выглядел проще. Теперь вспомним пример, который был сделан Вычислителем-2:  $2 \cdot 3 + 4$ . На Вычислителе-3 решение записывается так (рис. 4):

**ЗАПОМНИ 2**  
**ЗАПОМНИ 3**  
**УМНОЖЬ**  
**ЗАПОМНИ 4**  
**СЛОЖИ**



А теперь вернемся к примеру, который оказался недоступным для Вычислителя-2:  $2 \cdot 3 + 4 \cdot 5$ . Вот его решение (рис. 5):

**ЗАПОМНИ 2**  
**ЗАПОМНИ 3**  
**УМНОЖЬ**  
**ЗАПОМНИ 4**  
**ЗАПОМНИ 5**  
**УМНОЖЬ**  
**СЛОЖИ**



Вычислитель-3 имеет такой же процессор, как и первые два Вычислителя. Тем не менее он может решать более сложные задачи. Это объясняется тем, что у него появилась память из многих ячеек, организованных в виде стека.

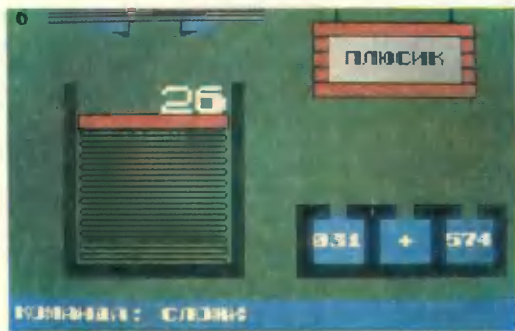
Заключительная часть урока — демонстрация компьютерной модели Вычислителя-3. Это программный исполнитель Плюсик. Дети, получив Плюсик на экраны, сразу узнают в его СКИ команды Вычислителя-3.

Впрочем, есть одна новая команда — **ОЧИСТЬ**. Она очищает стек от всех элементов. В Вычислителе-2 запись всегда ведется в единственную ячейку памяти, старое содержимое ячейки при этом теряется, поэтому нет необходимости в специальной очистке.

Учитель показывает, как работает команда **СЛОЖИ**. Затем он просит детей повторить действия, очистить экран и выполнить ту же самую операцию для тех же чисел, запоминаемых в другом порядке. Хорошо, если они смогут объяснить (вспомнить из предшествующих фрагментов урока) совпадение двух результатов: «От перемены мест слагаемых сумма не меняется».

Другие команды можно на этом уроке не показывать (если только не остался резерв времени). Дети прочитают о них в заданном на дом параграфе и подробно опробуют их на следующем уроке.

## § 4.12. Исполнитель Плюсик



### СРЕДА

Плюсик работает со стеком, в котором может храниться от одного до трех чисел, и счетным устройством для выполнения арифметических операций.

Счетное устройство работает так. Если, например, от 7 отнимается 5, то число 7 вводится в *первый вход* устройства, а число 5 — во *второй вход* устройства. Устройство сообщает, какую операцию выполнить, оно работает и выдает результат на *выходе* Р.

## КОМАНДЫ

СКИ Плюсика  
**ОЧИСТЬ**  
**ЗАПОМНИ** число  
**СЛОЖИ**  
**ВЫЧТИ**  
**УМНОЖЬ**  
**ДЕЛИ**

### КАК ОТДАЮТСЯ

Команды исполнителю набираются на клавиатуре компьютера заглавными русскими буквами точно так, как они обозначены в СКИ. В команде **ЗАПОМНИ** через пробел записывается число, которое нужно запомнить в стеке. Полное имя команды можно сокращать до первой буквы: О, З, С, В, У, Д.

### КАК ВЫПОЛНЯЮТСЯ

**ОЧИСТЬ** — стек очищается от всех чисел. 9

**ЗАПОМНИ** число — число запоминается в стеке.

**СЛОЖИ** — первое число из стека поступает на вход 2 счетного устройства, второе число — на вход 1, выполняется операция сложения; результат с выхода Р запоминается в стеке.

**ВЫЧТИ** — первое число из стека поступает на вход 2 счетного устройства, второе число — на вход 1, выполняется операция вычитания; результат с выхода Р запоминается в стеке.

**УМНОЖЬ** — первое число из стека поступает на вход 2 счетного устройства, второе число — на вход 1, выполняется операция умножения; результат с выхода Р запоминается в стеке.

**ДЕЛИ** — первое число из стека поступает на вход 2 счетного устройства, второе число — на вход 1, выполняется операция деления; результат с выхода Р запоминается в стеке.

### КОГДА НЕ МОГУ

**ЗАПОМНИ ЧИСЛО** — число больше 999 или в стеке нет места.

**СЛОЖИ** — в стеке нет двух чисел или сумма больше 999.

**ВЫЧТИ** — в стеке нет двух чисел или число в верхушке стека больше числа, расположенного под ним.

**УМНОЖЬ** — в стеке нет двух чисел или результат больше 999.

**ДЕЛИ** — в стеке нет двух чисел, или деление на нуль, или нацело не делится.

**Пример.** Написать для Плюсика программу следующих вычислений:  $240 - 17 \cdot 11 + 18$

**Решение**

ЗАПОМНИ 240  
 ЗАПОМНИ 17  
 ЗАПОМНИ 11  
 УМНОЖЬ  
 ВЫЧТИ  
 ЗАПОМНИ 18  
 СЛОЖИ

**Задачи и упражнения.**

1. Конюх и Кукарача пошли на базар. Конюх купил 9 упаковок овса по 5 кг в каждой, а Кукарача — 7 пачек ванильных сухариков по 3 кг. В Роботландии килограмм овса стоит 15 коп., а килограмм сухариков — 13 коп. Сколько денег заплатили друзья за покупку?

2. Проверить арифметическое равенство:

$$(13 \cdot 7 + 8 \cdot 5 + 4) : (17 \cdot 7 - 8 \cdot 11 - 16) = 9.$$

3. На рисунке показан стек Плюсика. Определите, какая команда выполнялась последней.

а)	<table style="border-collapse: collapse; width: 100px; height: 100px;"> <tr><td style="padding: 5px;">3 2 5</td></tr> <tr><td style="padding: 5px;">7 8</td></tr> <tr><td style="padding: 5px;">1 9</td></tr> </table>	3 2 5	7 8	1 9	б)	<table style="border-collapse: collapse; width: 100px; height: 100px;"> <tr><td style="padding: 5px;"> </td></tr> <tr><td style="padding: 5px;"> </td></tr> <tr><td style="padding: 5px;"> </td></tr> </table>			
3 2 5									
7 8									
1 9									

**Урок 11.10. Тема: «Исполнитель Плюсик»**

**План урока:**

Игра в стек.

Беседа о Плюсике.

Лабораторная работа: Плюсик.

Анализ исполнителя.

Домашнее задание: § 4.12; умножение с помощью сложения.

Урок начинается с игры. Учитель предлагает выполнить арифметические операции на модели стека, создаваемой из спичечных коробков. Предварительно на них надо наклеить белые бумажки и написать цифры. Чтобы не брать гору коробков и не усложнять информационную задачу вычислительными трудностями, в этой игре предлагается ограничиться лишь однозначными операндами и в минимальной степени использовать умножение, приводящее к большим числам.

Учитель объявляет, что предстоит игра в Плюсик. Как и в других играх с Вычислителями, здесь участвуют два артиста: Плюсик и Процессор. Плюсик получает команду, извлекает из стека данные, передает их

Процессору, получает от него результат и заносит его в стек. Напомним, что левая рука Процессора предназначена для приема первого операнда, правая — для второго. Артисты должны стоять на некотором расстоянии друг от друга, чтобы можно было фиксировать процесс передачи информации.

Команды выписаны на кодоскопе. Это необходимо, чтобы ученик, играющий роль Плюсика (да и весь класс, наблюдающий за игрой), мог в нужный момент ответить сообщением НЕ ПОНИМАЮ. Аварийные ситуации до сих пор не обсуждены в классе. О них предстоит разговор на сегодняшнем уроке. В игре учитель покажет несколько ситуаций, порождающих ошибки.

В каждой такой ситуации учитель останавливает игру и дает пояснения. Например, если стек вначале пуст, то после команд:

**ЗАПОМНИ 5  
 СЛОЖИ**

ученик-исполнитель, соорудив стек из единственного коробка, должен ответить естественным НЕ МОГУ. Следует комментарий учителя:

— Для выполнения любой арифметической операции нужны два числа. Имея только одно число или пустой стек (ни одного числа), исполнитель не может выполнить операцию.

Рассмотрим пример:

**ЗАПОМНИ 2  
 ЗАПОМНИ 3  
 СЛОЖИ**

По такой серии команд ученик-Плюсик сначала кладет в стек-коробок число 2, затем — число 3. По команде СЛОЖИ он передает Процессору число с верхушки стека (3) в левую руку, потом очередную верхушку (2) — в правую и говорит Процессору: «Сложи». Процессор возвращает Плюсику коробок с результатом (число 5). Плюсик укладывает его в верхушку стека.

Далее можно продолжать:

**ЗАПОМНИ 3  
 ВЫЧТИ**

Сверху на коробок с числом 5 укладывает-ся коробок с числом 3, а затем оба коробка заменяются числом 2. Дадим теперь команды:

**ЗАПОМНИ 5  
 ВЫЧТИ**

Как и в игре с Вычислителем-1, Процессор, получив данные, покажет головой (не разрешается вычитать из меньшего числа большее). Тогда ученик-Плюсик должен сказать НЕ МОГУ и вернуть данные в стек.

Важно продемонстрировать вычисление сложных выражений. Учитель может показать на кодоскопе выражение  $4 + 3 + 2 + 1$  и попросить одного из школьников дать не-

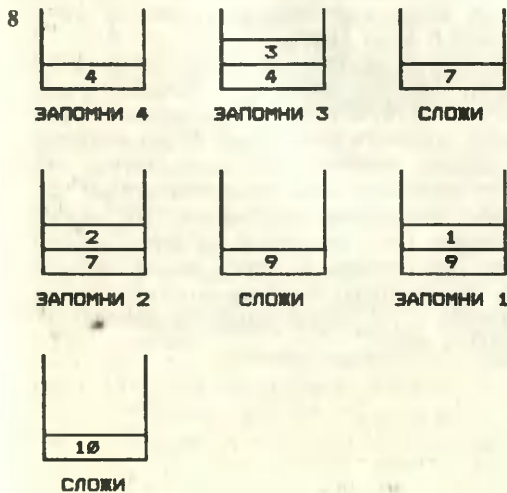
обходимые команды исполнителю. Здесь возможны варианты (можно попросить детей сделать несколько вариантов). Вариант с длинным стеком (рис. 7):

ЗАПОМНИ 4  
ЗАПОМНИ 3  
ЗАПОМНИ 2  
ЗАПОМНИ 1  
СЛОЖИ  
СЛОЖИ  
СЛОЖИ



Вариант с коротким стеком (рис. 8):

ЗАПОМНИ 4  
ЗАПОМНИ 3  
СЛОЖИ  
ЗАПОМНИ 2  
СЛОЖИ  
ЗАПОМНИ 1  
СЛОЖИ



Подобных примеров, решаемых разными способами, может быть создано достаточно много.

Рассказывая о Плюсике (§ 4.12), учитель активно использует демонстрационный экран. Демонстрируя каждую команду, он просит детей повторить ее на ученических компьютерах.

Надо обратить особое внимание на причины возникновения сообщения НЕ МОГУ. Например, можно попросить детей сделать четыре команды ЗАПОМНИ подряд:

ЗАПОМНИ 4  
ЗАПОМНИ 3  
ЗАПОМНИ 2  
ЗАПОМНИ 1

На четвертой команде Плюсик выдаст сообщение НЕ МОГУ. Такая ситуация не могла возникнуть в игре с коробками. Это должно быть объяснено: в игре не существовало ограничения на глубину стека, а в стеке Плюсика максимальная глубина точно определена — она равна 3. Это означает, что в таком стеке нельзя запомнить более трех чисел. 11

Еще одна причина появления сообщения НЕ МОГУ — использование более чем трехзначных чисел. Попробуйте умножить 500 на 3. Плюсик снова скажет НЕ МОГУ.

После лабораторной работы следует коллективно провести анализ исполнителя Плюсик. Для этого учитель снова использует известную схему знакомства. Классу предстоит ответить на вопросы:

в какой форме представлена обрабатываемая исполнителем информация (Плюсик работает исключительно с числовой информацией);

в какой среде работает исполнитель (ограничения стека и счетного устройства);  
какова система команд исполнителя;  
как выполняется каждая из команд;  
как подается каждая команда (краткие и полные названия);

в каких случаях исполнитель Плюсик выдает сообщения НЕ МОГУ (переполнение стека, арифметические операции над стеком, содержащим менее двух чисел, вычитание меньшего из большего, деление не нацело, слишком большое число).

Урок 11.11. Тема: «Алгоритмы работы со стеком»

План урока:

Разминка — черные ящики.

Проверка домашнего задания.

Решение типовых задач на исполнителе Плюсик.

Конструирование арифметических задач.  
Домашнее задание: арифметические зада-

чи из школьного учебника математики.

Возможностей Плюсика достаточно для реализации вычислительной части большинства школьных арифметических задач. Такие задачи традиционно решаются с помощью плана из вопросов-действий: отвечая на каждый вопрос плана, школьник определяет необходимое действие и производит соответствующее вычисление.

Предусматривая решение задачи на Плюсике, необходимо, конечно, выделить вопросы-действия, однако выполнение вычислений откладывается. В результате из пунктов плана получается единая вычислительная формула, которая программируется для Плюсика.

Такая методика оказывается полезной не только для информатики, но и служит пропедевтикой решения задач в общем виде.

Для этой части урока рекомендуются три задачи.

**Задача № 1.** С первого поля крестьяне собрали 38 т картофеля, со второго — 19, с третьего — 47, а с четвертого — 26. Сколько всего картофеля собрали крестьяне?

**Задача № 2.** Петя Кук по просьбе мамы купил 3 булки по 26 коп. за штуку, 7 коробков спичек по 2 коп. за коробок и билеты в кино для себя и брата (цена одного билета 25 коп.). Сколько денег потратил Петя?

**Задача № 3.** На каникулах дети поехали на экскурсию в Крым. Они ехали 13 ч на поезде со скоростью 67 км в ч. После этого они поехали на троллейбусе. Общая длина маршрута — 947 км. Сколько километров ребята проехали на троллейбусе?

В первой задаче итоговая формула очевидна и не требует постановки отдельных вопросов:  $38 + 19 + 47 + 26$ . Число слагаемых превосходит глубину стека Плюсика, поэтому команды запоминания и сложения неизбежно будут чередоваться. Возможно несколько правильных вариантов.

Затем решается вторая задача. И в этом случае формула выписывается легко, это сумма произведений:

$$3 \cdot 26 + 7 \cdot 2 + 2 \cdot 25.$$

Возможная программа для Плюсика:

ЗАПОМНИ 3  
ЗАПОМНИ 26  
УМНОЖЬ  
ЗАПОМНИ 7  
ЗАПОМНИ 2  
УМНОЖЬ  
СЛОЖИ

ЗАПОМНИ 2  
ЗАПОМНИ 25  
УМНОЖЬ  
СЛОЖИ

Решение первых двух задач убедило детей, что для программирования на Плюсике необходимо сначала получить общую формулу. В третьей задаче составить формулу труднее. На примере этой задачи можно рассмотреть методику систематического построения формулы исходя из привычных вопросов-пунктов.

**Вопрос 1.** Сколько километров проехали дети на поезде?  
13-67.

Вычисление не производится: в общую формулу войдет не результат, а само произведение.

**Вопрос 2.** Сколько километров проехали дети на троллейбусе?  
947—13-67.

При программировании этой итоговой формулы для Плюсика нужно вспомнить, что уменьшаемое должно попасть в стек раньше вычитаемого. Возможное решение:

ЗАПОМНИ 947  
ЗАПОМНИ 13  
ЗАПОМНИ 67  
УМНОЖЬ  
ВЫЧТИ

На следующем этапе урока детям предлагается задание-импровизация: они должны сами сочинить условие задачи для решения ее на компьютере с помощью Плюсика. Несколько человек по очереди предлагают классу свои формулировки задач.

Учитель фиксирует две-три постановки задачи, выбирая наиболее удобные для решения на Плюсике, а затем класс коллективно выбирает из них одну. Эта задача становится объектом следующего задания для всего класса. Вновь включаются экраны, на которых детей ждет Плюстик.

Домашнее задание (две арифметические задачи) полезно задать по школьному учебнику математики. Такая форма задания будет неявно подтверждать слова, произнесенные на первых занятиях по информатике, об использовании компьютеров на уроках по другим школьным предметам. Поскольку учебников по математике у детей может сейчас не оказаться, адрес задач должен быть указан точно. Учитель должен заранее заглянуть в учебник и выписать номера задаваемых задач.

## ПРЕДЛАГАЕМ «МИКРОН»

В «ИНФО» № 4 за 1990 г. помещена статья В. Каптелинина и Т. Дроздовой «Опыт обучения работе с текстовым редактором». В ней приведен анализ нескольких профессиональных текстовых редакторов с точки зрения специфического пользователя — учителя информатики общеобразовательной школы. По существу заметка формулирует технические требования к адаптируемому текстовому редактору учебного (школьного) назначения.

Коллектив разработчиков программно-методической системы «Роботландия» (ИПС АН СССР) сообщает, что сформулированным требованиям отвечает текстовый редактор «М и к р о н», входящий в состав системы, но допускающий и автономное использование.

«Микрон» успешно прошел экспериментальное опробование в школьном учебном процессе. Методика применения «М и к р о н а» на уроках информатики у младших школьников будет описана в одном из ближайших номеров журнала. В настоящее время «М и к р о н» реализован на ПЭВМ «Ямаха» MSX-1 и MSX-2, а также на компьютерах IBM PC.

Условия поставки можно выяснить на предприятии «Роботландия», в числе учредителей которого — ИПС АН СССР.

Адрес: 152140, г. Переславль-Залесский, ул. Полевая, д. 4, кв. 35. Первин Ю. А.

Телефон служебный (085-35) 2-08-65.

---

Предприятие «Роботландия», специализирующееся на разработке и внедрении программно-методических систем обучения информатике, готовит к выпуску продукт, который включает: системы учебных и профессиональных программных средств, составляющих содержание и инструментарий курса информатики для взрослых, в первую очередь для руководителей школ и учителей предметников;

серию сценариев видеофильмов, которые служат основным дидактическим средством для проведения самостоятельных занятий пользователя у компьютера;

методическое пособие, содержащее поурочные комментарии к курсу.

Предприятие, заинтересованное в создании интерфейса, максимально удобного для широкого круга пользователей, просит сообщить пожелания, которые могут быть учтены в ходе разработки: объем курса (количество часов, ПЭВМ, лексическая база системы); предложения по содержанию курса (основные тематические разделы, типовые и желаемые программные средства); другие пожелания.

«Роботландия» ожидает писем, пожеланий и рекомендаций по адресу: 151140, г. Переславль-Залесский, ул. Полевая, д. 4, кв. 35. Первину Юрию Абрамовичу. Служебный телефон: (08535) 2-08-65.

# Институт программных систем АН СССР объявляет конкурс

учебных программ по курсам, которые могут быть проведены в Международной детской компьютерной летней школе 1991 г.

Школа будет проводиться на берегу Плещеева озера под г. Переславлем-Залесским в несколько смен, каждая продолжительностью 24 дня. В течение смены школьники должны ознакомиться с языками программирования, прикладными программными системами, новыми информационными технологиями (экспертные системы, базы знаний, обработка изображений, сети, функциональное и логическое программирование) и их применениями в гуманитарных областях деятельности человека (культура, история, экология и т. д.).

Выбрав для себя тему (курс), школьники в течение смены выполняют индивидуальный или коллективный проект и по результатам работы делают сообщение (доклад) на научной конференции в конце смены. Предусматривается участие школьников из разных стран мира.

Техника, которой будет располагать школа, — персональные РС-совместимые машины (XT и AT) и «Ямахи» MSX-1 и MSX-2, графические планшеты, сканеры.

С преподавателями, победившими на конкурсе учебных программ, будут заключены контракты на период работы в школе.

**Документация, присылаемая на конкурс, должна включать:**

1. Сведения о конкурсантах: фамилия, имя, отчество, год рождения, ученая степень, ученое звание, место работы, должность, специальность, стаж, список трудов (или количество трудов и список важнейших), адрес, телефоны, другие средства коммуникации.

2. Объяснительная записка по предлагаемому курсу: обоснование актуальности; ориентация, в том числе возрастная; предполагаемые предварительные знания и умения; требуемая для реализации курса вычислительная техника и периферия; объем курса и форма его проведения; желательные условия организации учебного процесса; потребность в ассистентах, их квалификация.

3. Учебная программа с разбивкой по дням и часам (не более 19 дней; в день не более 3—4 ч.).

4. Предполагаемый список тем для индивидуальных и коллективных проектов учащихся.

Общее описание оригинальных проектов.

5. Дополнительные сведения и пожелания.

**Документы следует направлять по адресу:**

152140, г. Переславль-Залесский, Институт программных систем АН СССР, директору профессору Альфреду Карловичу Айламазяну — с обязательной пометкой на конверте «Конкурс МДКЛШ». Документы должны быть получены не позднее 1 февраля 1991 г.

Последующая переписка и конкурс будут завершены к 1 мая 1991 г. В интервале между 1 февраля и 1 мая 1991 г. Институт программных систем АН СССР проведет семинар с обсуждением конкурсных программ.

Для справок.

Адрес: 152140, г. Переславль-Залесский, а/я 46.

Телефоны: (08535) 20865,  
(08535) 98128.



## Опыт разработки и использования УАОС

В последнее время рядом авторов [1, 2] активно разрабатывается концепция универсальных автоматизированных обучающих систем (УАОС). В данной статье анализируется опыт разработки и использования УАОС, накопленный в Коломенском педагогическом институте.

Общая структура УАОС на концептуальном уровне имеет вид (рис. 1):

ния учебного материала и оценки качества знания.

Подсистема редактирования обеспечивает возможность оперативно заносить во внешнюю память учебную информацию с последующим редактированием; редактировать справочник информационно-поисковой подсистемы; редактировать таблицы связей между разделами и



Информационно-поисковая подсистема выполняет функции поиска в базах данных конкретной предметной области (информатика, математика, история, география), курса, темы, раздела, типа обучающего курса и режима, которые задаются пользователем — преподавателем, студентом, прикладным программистом.

Подсистема обучения обеспечивает выдачу порции учебной информации (справочной, информационной, обучающей, контролирующей, программируемой и т. д.) в виде произвольной последовательности вопросов-эталонов, ответов и моделей — аналитических, графических и эвристических.

Модели могут использоваться для иллюстрации процесса обучения, лингвистической обработки ответа, закрепле-

эталонами, разделами и темами, темами и курсами и т. д., а также редактировать таблицы связей между эталонами и моделями, поддерживающими процесс обучения.

При работе с УАОС используется терминология:

эталон — вопрос или ответ, принятые преподавателем за стандарт;

таблица связей — матрица, элементами которой являются либо номера соответствующих предметных областей, курсов, тем, разделов, эталонов, либо номера двоичных признаков, указывающих на наличие связки между двумя объектами  $\theta_{ij}$  области  $U$ :

$$P_{ij} = \begin{cases} 1, & \text{если } \theta_{ij} \in U \\ 0, & \text{если } \theta_{ij} \notin U \end{cases}$$

Объект  $\theta_{ij}$  области  $U$  — подмножество

учебной информации из иерархически вложенного множества, причем уровень вложенности  $n \leq 6$ ;

размерность обучающего курса — число  $m$  вопросов или ответов, содержащихся в буфере подсистемы обучения ( $m \leq 10$ );

размерность эталона — количество символов эталона, поддерживаемое буфером символьной строки интерпретатора Бейсик ЭВМ «Искра-226» ( $l \leq 240$ ) [1];

оценка — целое число  $k = f(m, t)$ , где  $m$  — размерность курса,  $t$  — время, затрачиваемое пользователем на работу с курсом. Оценка выражается в баллах, причем вид функциональной зависимости задается преподавателем — специалистом в конкретной предметной области знаний;

форма ответа — символьное или цифровое обозначение утверждения пользователя (число, арифметическое выражение, уравнение, историческая дата, слово, предложение, текст и т. д.).

Программный инструментарий в УАОС построен на модульном принципе, позволяющем произвольно задавать конфигурацию системы путем вызова подпрограммы из внешней памяти. Такая организация программного инструментария позволяет поддерживать многовариантность форм вопросов, ответов, моделей и гибкость в использовании как различных режимов работы, так и различных типов обучающих курсов.

Рассмотрим методику применения УАОС на ряде факультетов педагогического института (физико-математическом, общетехническом, историческом).

УАОС использовалась при изучении основ информатики и вычислительной техники студентами III курса математического отделения ФМФ по численным методам математического анализа и линейной алгебры по разделам: «Решение алгебраических уравнений с одним неизвестным», «Решение систем и линейных алгебраических уравнений с неизвестными методом исключения Гаусса»; «Интерполяционные полиномы Лагранжа и Ньютона», «Численное интегрирование» и др.; на II курсе физического отделения ФМФ — по программированию на ЭВМ и численным методам; слушателями фа-

культета повышения квалификации преподавателей педагогических училищ — на общетехническом факультете.

Подбор содержания для проверки знаний при помощи УАОС предполагал систематический контроль минимальной подготовленности обучаемого и ориентирования его в учебном материале. Количество заданий, их последовательность, краткость формулировки были направлены на решение вопроса о допуске к учебному занятию и не ставили целью заменить устные аудиторские формы работы. УАОС применялась и для консультирования студентов, отсутствовавших при коллективном использовании системы. В таком случае преподаватель, не отвлекаясь от работы со всем потоком, получал возможность индивидуально помочь обучаемому.

В начале цикла обучения УАОС использовалась для приобретения и закрепления у будущих пользователей важнейших навыков работы с ЭВМ «Искра-226»: работы с клавиатурой, использования директив для управления вычислительным процессом, владения простейшими приемами редактирования текста.

Методические материалы, наполняющие систему, могут быть применены и для других УАОС, а также для варианта настоящей УАОС, адаптированной на другие персональные компьютеры, например «Ямаху», БК-0010, БК-0010Ш. Эти же методические материалы можно использовать для контроля первого и второго уровней. К настоящему времени выполнен и статистически обработан контроль первого уровня. На наш взгляд, внедрение машинных методов обучения в курсе ОИВТ особенно оправдано тем, что средство обучения выступает в роли предмета обучения.

Свои особенности имеет применение УАОС на историческом факультете. Система была первоначально использована при преподавании курса «Хронология» (учебный цикл «Вспомогательные исторические дисциплины»). Выбор именно этого курса диктовался рядом соображений. Во-первых, одной из целей изучения исторической хронологии является обучение студентов методам редукции дат, выраженных в различных календар-

ных системах. А так как редукция связана с математическими вычислениями (применяются формулы, коэффициенты, календарные поправки), это открывает возможность моделирования заданий. Во-вторых, введение в содержание заданий исторических дат способствует их усвоению и запоминанию, и в-третьих, помогает студенту-гуманитарию преодолеть внутреннюю неуверенность перед компьютером, возобновить и закрепить полученные в средней школе навыки общения с машиной.

Подготовленная программа охватывает три блока вопросов, соответствующие трем разделам практической хронологии: взаиморедукция дат мусульманского и григорианского календарей, перевод юлианских дат; редукция дат, выраженных в византийской эре «от сотворения мира». Программа носит контролирующий характер с элементами обучения. Студенту задается режим работы под управлением системы.

Блоки включают 10 вопросов-заданий, часть из них носит информационный, «теоретический», а часть — «практический» характер. Для иллюстрации приведем формулировки вопросов обоих типов.

«За сколько лет в юлианском календаре накапливается ошибка в одни сутки? На какой день приходится начало года в мусульманском календаре? Что считается началом суток в лунном мусульманском календаре?»

«Перевести на эру хиджра дату Куликовской битвы. Перевести на новую эру дату смерти Тимура (807 г. х.). В какой день отмечалось 280-летие Полтавской битвы? Перевести на юлианский календарь дату Андрусовского перемирия с Польшей — 30 января 1667 г.»

При выполнении задания студент вводит в машину окончательный результат вычислений в символьном или цифровом виде (в зависимости от вопроса). Система информирует студента о правильности или ошибочности ответа. В последнем случае предусмотрена возможность (только для заданий по переводу дат, выраженных в византийской эре) нового вычисления и ввода в машину второго результата, но, разумеется, со снижением оценки. Оценки выставляются за

ответы по каждому блоку заданий, что позволяет определить пробелы в знаниях испытуемого.

Опыт использования компьютеров продемонстрировал целесообразность их включения в процесс изучения хронологии и позволил ввести ЭВМ в преподавание новой истории.

Курс «Новая история, ч. I (1640—1870)» представляет серьезную трудность для студентов-историков из-за обилия учебного материала: фактов, терминов, дат. Так как хронологические задания наиболее легко формализуются, УАОС использовалась для проверки усвоения исторических дат. Это тем более желательно, поскольку одновременно студенты изучают курс ЭВТ.

Контролирующая программа для ЭВМ «Искра-226» была составлена по весьма сложному разделу курса — «Английская буржуазная революция XVII в.» — именно с целью проверки знания хронологии [3, 4].

Студенты сдавали зачет на ЭВМ. Содержание зачета составили 8 заданий, причем невыполнение одного из них снижало оценку на балл. Невыполнение трех и более заданий оценивалось «неудовлетворительно», что влекло за собой пересдачу зачета по иной программе.

В программе использовались задания нескольких типов. Тип 1 — выбор правильной даты предлагаемого события из нескольких дат. Тип 2 — противоположное задание — выбор одной из предлагаемых характеристик даты. Тип 3 — ответ на вопрос, требующий расчета: «Сколько лет прошло от события А до события Б?»

При подготовке заданий с целью их усложнения варианты ответа подбирались по принципу сходства, например: «Протекторат Кромвеля продолжался: с 06.12.1653 по 01.09.1658 г.; с 16.12.1653 по 03.09.1658 г.; с 10.12.1652 по 16.12.1659 г.» (второй ответ правильный).

По мнению студентов, подобная форма обучения чрезвычайно перспективна и полезна.

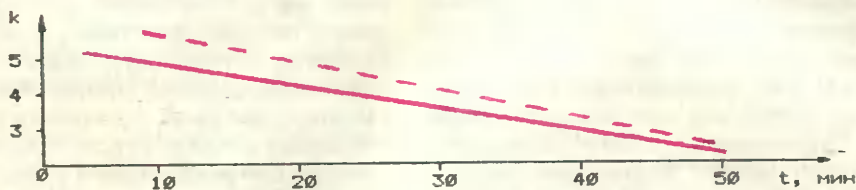
По результатам педагогического эксперимента на физико-математическом и историческом факультетах, используя

статистические методы обработки данных, можно сформулировать следующие выводы.

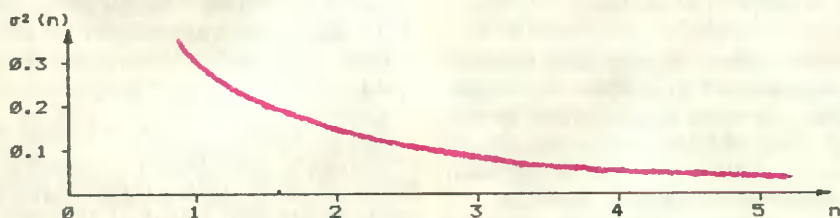
1. Надежность, гибкость и комфортность работы пользователя с УАОС поддерживается модульным принципом разработки программного инструментария системы. (Число модулей системы к настоящему моменту достигает 60, при этом модули разделяются по выполняемым ими функциям на информационно-поисковые, контролирующие, обучающие, моделирующие, коммуникационные и редактирующие.)

нескольким разделам одной и той же темы. Это позволяет говорить об объективности оценивания знаний конкретного пользователя в процессе приобретения им навыков работы с системой. На рис. 2 показана линейная регрессионная зависимость  $k = f(m, t)$  для двух различных потоков студентов физико-математического и исторического профилей. На рис. 3 показана зависимость дисперсии оценки знаний среднестатистического обучаемого от числа пройденных им разделов курса. Данные были получены при внедрении в течение ряда лет

2



Условные обозначения:  
— зависимость для физико-математического профиля  
- - - зависимость для исторического профиля



3

2. Методическая значимость использования системы определяется сценарием предъявления учебной информации, разработка которого целиком является функцией пользователя-специалиста в предметной области.

3. На основе изучения корреляционной зависимости вида  $k = f(m, t)$ , где  $m$  — размерность курса,  $t$  — время, затрачиваемое пользователем на работу с курсом,  $k$  — оценка знаний в баллах, нами сделан вывод о линейности связи оценки знаний от времени работы с курсом независимо от размерности курса.

Важной особенностью работы с УАОС является устойчивость характера данной связи для одной и той же группы пользователей при контроле их знаний по

экспериментальной методике работы с системой в учебных группах всех курсов физико-математического и исторического факультетов Коломенского пединститута.

4. Результаты описанного выше статистического анализа не позволяют, однако, в настоящее время провести методически обоснованную разработку таймера.

5. Практический опыт внедрения системы подтвердил правильность поэтапной разработки системы: разработка технического задания, консультации с потенциальными пользователями, разработка экспериментальной версии системы и непрерывная адаптация исходной версии к изменяющимся требованиям методистов.

1. *Куприянов С. В.* Опыт разработки и внедрения в учебный процесс автоматизированных обучающих систем // Применение ЭВМ в учебном процессе. Махачкала, 1989.
2. *Савельев А. Я., Новиков В. А., Лобанов Ю. И.* Подготовка информации для

автоматизированных обучающих систем / Под ред. А. Я. Савельева. М.: Высшая школа, 1986.

3. Новая история. Ч. I / Под ред. А. В. Азо. М.: Высшая школа, 1988.

4. *Пронштейн А. П., Кияшко В. Я.* Хронология. М.: Высшая школа, 1981.

## Деятельность младших школьников в компьютерной игровой учебной среде

Разработка компьютерных игровых сред, ориентированных на детей дошкольного и младшего школьного возраста, — одна из наиболее актуальных проблем компьютеризации обучения. Игра как вид человеческой деятельности обладает большим мотивирующим и эмоциональным потенциалом, который можно и нужно использовать в учебных целях. Деятельность в игровой компьютерной среде позволяет ставить учебные цели, связанные с овладением элементами компьютерной грамотности. Естественно, что уже при разработке этой среды должны быть учтены особенности не только учебных задач, которые предполагается в ней решать, но и процесса их решения. Следовательно, разработка компьютерной игровой учебной среды (именуемая далее КИУС) требует предварительного анализа содержательного и операционального аспектов учебной деятельности, выработки средств описания и преобразования объектов, которые моделируются в учебных задачах, включая соответствующее программное обеспечение.

Психолого-педагогические особенности деятельности учащихся в КИУС не только представляют значительный интерес как в теоретическом (например, при сравнительном анализе традиционных и компьютерных средств решения учебных задач и процессов их решения), так и в практическом плане только сами по себе, но и являются основанием для следующих этапов проектирования КИУС. При этом проектирование и разработка являются итераци-

онным процессом, реализующимся в логике восхождения от абстрактного к конкретному.

Конструирование учебной проблемы должно предполагать:

- определение перечня объектов и/или множеств объектов, которые можно объединить в наборы учебных задач;
- выделение всевозможных операций и функций, применяемых к выделенным объектам и/или множествам;

- формирование компьютеризированных игровых сред (как динамических, так и статических) на основе выделенных объектов, множеств, операций и функций;

- разработку психолого-педагогического обеспечения, позволяющего строить эффективную учебную деятельность детей в игровой среде с использованием теоретико-множественного аппарата.

Объектно-ориентированная компьютеризованная среда обладает большими возможностями для эффективной организации учебной деятельности детей от 3 до 10 лет. Закрепление сформированных знаний и навыков возможно в определенной языковой среде, реализующей достаточно простое описание моделей объектов.

Любая КИУС должна обеспечивать развитие творческих компонентов мышления у обучаемых. Для этого необходимо следующее:

- соответствие программно-информационных средств компьютерной системы структуре процесса решения задач;
- адекватность моделей объектов, включаемых в деятельность учащихся при

решении конкретной задачи.

В технологию обучения, реализуемую в КИУС, заложены следующие принципы.

1. Учащемуся должна быть предоставлена возможность свободно манипулировать объектами.

2. Процедуры (операции), применяемые к объектам, должны быть хорошо описаны и понятны учащемуся, обладать «психологической естественностью» (соответствовать основным компонентам процесса решения задач в соответствующей предметной области).

3. Желательно свести к минимуму использование дополнительных языковых средств для описания детьми в среде компьютерной системы объектов и применяемых к ним процедур (наиболее эффективно использование функциональных клавиш и/или программируемой клавиатуры).

4. Необходимо обеспечить ребенку возможность конструирования различных моделей мира в среде компьютерной системы. Под построением моделей мира будем понимать реализацию компьютерной модели, соответствующей восприятию ребенком определенных событий и ситуаций в окружающей среде. К примеру, формирование изображения группы предметов, взаимодействие с объектами компьютерной игры, манипулирование набором исполнителей и т. п.

Предпосылками успешной реализации учебной деятельности являются сформированные у ребенка действия по классификации и сравнению. Ее ориентиром служит схема выбора объектов, выработанная учащимся (или предложенная ему). Классификация выполняется по групповым (множественным) и функциональным признакам. В этом случае естественной будет реализация множественных операций над объектами создаваемой модели мира. Динамику создаваемого мира можно задать при помощи группы аффинных преобразований, дополнив ее рядом специальных функций. Указанный набор функций выполняет роль посредника между учащимися и программно-информационными средствами КИУС. Посредническая роль заключается в достаточно строгом описании исходного и результирующего

состояний объектов строящейся модели мира.

С другой стороны, формирование действия моделирования требует построения композиций из отобранных множеств объектов, которое должно входить в структуру строящейся модели. Построение композиций осуществляется в процессе последовательного и целенаправленного применения к этому множеству соответствующих преобразований (если они существуют и могут быть выполнены). Преимуществом КИУС является то, что в ее среде возможна реализация широкого спектра преобразований над множествами.

В мышлении детей дошкольного и младшего школьного возраста наглядно-образное и наглядно-действенное преобладают над абстрактным, отвлеченным. Поэтому достаточно абстрактные понятия — множества, функции, преобразования, операции и т. п. — необходимо интерпретировать на доступном для этого возраста уровне.

В данной статье приведены основные результаты психолого-педагогического исследования, проведенного на материале системы ЭЛЬКОН (электронный конструктор), разработанной сотрудниками отдела диалоговых обучающих систем Института кибернетики им. В. М. Глушкова АН УССР и ориентированной на деятельность детей дошкольного и младшего школьного возраста. КИУС, реализуемая в среде системы, предполагает выполнение следующих условий, необходимых для решения учебных задач.

1. Возможность свободного манипулирования учащимся с объектами среды.

2. Соблюдение требования «психологической естественности» процедур.

3. Минимизация языковых средств описания и манипулирования объектами в среде.

4. Обеспечение возможности конструирования (композирования) ребенком различных моделей из предложенных ему (или сформированных самостоятельно) графических базовых элементов (примитивов).

Система ЭЛЬКОН состоит из семейства композиционных таблиц (КОМТАБ) — специального набора ма-

триц размерностью  $32 \times 32$ ; ленточной памяти, объединяющей графические базовые элементы (ГБЭ) и библиотеки функций, реализующих теоретико-множественные и аффинные операции и преобразования.

Библиотека ГБЭ представляет собой ленту, разбитую на конечное число ячеек (кадров). Каждая ячейка (кадр) определяет графический базовый элемент. В некотором смысле систему можно определить как модель машины Тьюринга с конечным числом состояний и динамически задаваемым алфавитом. Выделенным элементом алфавита может быть пустой ГБЭ.

Лента располагается на экране вертикально. Просмотр кадров ленты осуществляется последовательно в направлении вверх или вниз. Кадр, содержащий необходимый в данный момент ГБЭ, должен быть установлен между специальными стрелками-указателями. Каждый кадр, как уже было отмечено, содержит выпуклое замкнутое подмножество точек плоскости, которое определяет ГБЭ. В процессе целенаправленного выделения ГБЭ (или, более строго, подмножеств точек плоскости (кадров)) ребенок, используя описанные преобразования и функции, формирует множество конечного числа подмножеств точек плоскости.

В режимах работы с библиотекой ГБЭ учащийся может сформировать и/или изучить графическую модель каждого элемента из множества ГБЭ, используемых при построении графического изображения (модели мира). При необходимости учащийся может включить в библиотеку новую графическую модель на уровне описания ее в виде ГБЭ.

Основные правила построения графических композиций представляются при помощи коммутативной группы преобразований: ПЕРЕНОС; МАСШТАБ; ПОВОРОТ; СИММЕТРИЯ. С их помощью реализуются функции: ОТПЕЧАТКА ГБЭ, ПОВОРОТЫ, СИММЕТРИЯ (горизонтальная и вертикальная), ПОСТАВИТЬ И СТЕРЕТЬ ТОЧКУ, ЗАПИСЬ РАБОЧЕГО ГБЭ, ПРОСМОТР ВНИЗ, ПРОСМОТР ВВЕРХ, ПЕРЕНОС. Для получения композиции используются

множественные операции: ОБЪЕДИНЕНИЕ, ПЕРЕСЕЧЕНИЕ, ИНВЕРСИЯ, логические функции: «XOR, OR, AND» и функции определения и замены цвета изображения ГБЭ и КОМТАБ.

Каждому режиму работы с библиотекой ГБЭ соответствует свой набор экранных функций. В библиотеку экранных функций ЭЛЬКОН включены функции, реализующие операции над множествами, обеспечивающие связь режимов между собой и ряд специальных преобразований ГБЭ и КОМТАБ как множеств. Обращение к каждой функции осуществляется при помощи функциональных клавиш, клавиш управления курсором и программируемой клавиатуры.

Дополнительно реализованы функции: поставить точку текущего размера, удалить точку текущего размера, изменить текущий размер точки, сменить рисунок,

21



быстрое перемещение, создание ГБЭ, работа с памятью, наложение ГБЭ на поле, сжатие части поля в ГБЭ, печать полей и библиотеки, ведение файловой системы, увеличение размера сжимаемого участка, перемещение части рисунка, уменьшение размера сжимаемого участка, печать КОМТАБ и библиотеки.

Последовательность применения и/или использования экранных функций определяет целенаправленность деятельности ребенка в среде системы, хотя, вообще говоря, их использование (применительно к выбранным ГБЭ) может иметь произвольный характер. Однако более целесообразно, с точки зрения достижения учебного эффекта, связать

применение этих функций со структурой конструируемой модели объекта или процесса.

Следует отметить, что учащийся фактически осуществляет конструирование модели в строго заданной функциональной среде. Сформированное множество конечного числа подмножеств точек плоскости можно определить как область значения суперпозиции использованных учащимся функций системы.

Описание динамических процессов в среде построенных моделей графических объектов можно задавать при помощи группы аффинных преобразований, к которым относятся функции: ПЕРЕНОС, ПОВОРОТ, ВЕРТИКАЛЬНАЯ И ГОРИЗОНТАЛЬНАЯ СИММЕТРИЯ, СЖАТИЕ, УВЕЛИЧЕНИЕ (режим ЛИНЗА). Для этого в среде системы реализован интерпретатор, ориентированный на группу аффинных преобразований.

В экспериментальном исследовании изучались возможности разработанной КИУС, связанные в первую очередь с задачами естественного воспитания (выполнение рисунков, изобразительной деятельности) и формирования геометрических представлений.

Первая серия экспериментов (испытываемые — дети 6—8 лет СШ № 132 г. Киева) была посвящена усвоению детьми компонентов программы экспериментального обучения изобразительному искусству в начальной школе.

Цели эксперимента: адаптация и дальнейшая разработка КИУС для решения задач:

а) обеспечения свободного композирования в ходе создания рисунка на компьютере;

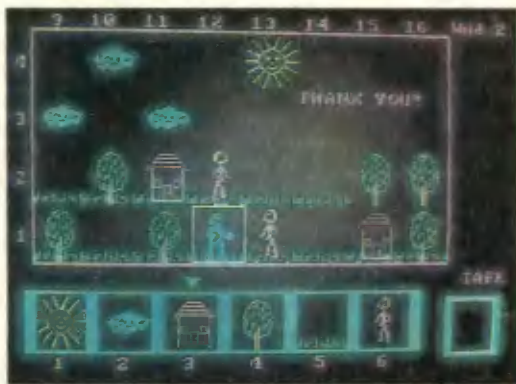
б) эстетического воспитания и развития общей культуры учащихся (формирование композиционного равновесия, центрирования, гармония цвета и т. п.);

в) формирования представлений об аффинных преобразованиях на плоскости (параллельный перенос, симметрия, поворот, подобие и пр.);

г) формирования элементов компьютерной грамотности (навыки работы с ЭВМ).

Серия экспериментов включала 8 занятий (кружковая работа). Ознаком-

ление с операциями переноса и печати (1, 2), с поворотами, ознакомление с функциональными клавишами, обеспечивающими задание цвета (3, 4), работа, направленная на создание своей библиотеки ГБЭ (режим «Линза», 5, 6, 7).



Анализ рисунков, выполненных в ходе свободного композирования на 1-м и 2-м занятиях (библиотека примитивов из 6 элементов: дом, дерево, тучка, человек, травка и солнышко; цвет — черный), показывает, что дети стремились использовать все 6 примитивов (91 %). 54 % испытуемых пытались заполнить примитивами верхнюю и нижнюю полосы (сверху — солнышко и тучки; остальные примитивы — внизу). Отдельные детали располагались на 2-й и 5-й полосах; 3-я и 4-я, а также центр рисунка оставались свободными. 18 % испытуемых заполнили примитивами все 36 имеющихся квадратов. 28 % разместили элементы рисунка по всему полю, оставляя свободными 1—2 полосы (преимущественно 3-ю и 2-ю).

При анализе рисунков с использованием поворотов, вертикальной и горизонтальной симметрии, операции объединения и клавиши цвета выявились следующие тенденции:

а) по окраске рисунков наиболее предпочтительным выступает синий цвет (более 60 % детей делали его основным фоном рисунка). Наименее предпочтительный — черный (например, на 4-м занятии он выступал основным на 15 % рисунках, а на 3-м занятии в 66 % рисунках вообще не использовался); более того, дети пытались замесить цвет черного примитива на любой



другой (статистически значимых тенденций здесь не обнаружено), пользуясь клавишей цвета;

б) композиционно размещение деталей рисунка на экране выглядит так: примитивами заполнены все клеточки (центр экрана полностью занят) — 15 %; в остальных рисунках заполнены до 2/3 имеющихся клеточек поля, центр рисунка свободен, верхние полосы использовались для дождика, тучек и солнышка, нижние — для деревьев, домиков, человечков, травы и т. д. (около 85 %).

На этих двух занятиях рекомендовалось применять повороты и использовать вертикальную и горизонтальную симметрию. Поэтому из набора шести предложенных примитивов использовалось четыре-пять. К ним добавлялся самостоятельно созданный примитив «дождик» из примитива «травка» либо с помощью двух поворотов к исходному примитиву, либо применялась операция «горизонтальная симметрия». Дети говорили, что хотели изобразить на рисунке «дождливый день». Во всех случаях применялись предложенные преобразования (от 1 до 9 на одном рисунке). При создании нового примитива «дождик» дети убедились в эквивалентности последовательного применения двух поворотов и операции «горизонтальная симметрия».

В отношении сюжета дети разделились следующим образом.

60 % (в основном девочки) стремились применять изученные преобразования в небольшом количестве — так, чтобы тот же рисунок остался «правдоподобным»; операция «поворот на  $90^\circ$  и  $180^\circ$  практически не применялась.

40 % детей (в основном мальчики) интересовала не «правдоподобность» рисунка, а возможность применить к элементам картинке все изученные виды поворотов и симметрий, т. е. стремление изменить стандартное расположение примитива и желание реализовать полученные знания преобладали над реальностью рисунка. В картинках такой направленности встречались различные преобразования (от 6 до 9); из них наиболее часто применялись повороты на  $90^\circ$  («домик», «деревцо», «челове-

чек» на боку) или повороты на  $180^\circ$  (те же примитивы «вверх ногами»).

Группа занятий была посвящена созданию или редактированию графических элементов и осуществлялась последовательно на 5-м, 6-м, 7-м занятиях. На 5-м, 6-м заданием ставилось следующим образом: создать новый примитив посредством движения точки по специальной таблице «ЛИНЗА».

Выяснилось, что тематика создаваемых рисунков резко отличалась у мальчиков и у девочек. Мальчики создавали роботов, телевизоры, машины, самолеты, спортивные снаряды, цифры, а девочки предпочитали узоры для салфеток, клумбы для цветов, стадионы и т. п. Некоторые дети заинтересовались перемещением точки по диагонали, вследствие чего ознакомились с работой двух соответствующих клавиш одновременно. Причем мальчики создавали свой рисунок только посредством движения точки в режиме ЛИНЗА, не прибегая к помощи библиотеки. Часть девочек охотно составляли узоры, пользуясь имеющимися на ленточке отрезками прямых, звеньями ломаных, кривыми линиями, не прибегая на первом этапе работы в режиме ЛИНЗА. На первых двух занятиях при создании своих ГБЭ дети рисовали медленно, с частыми исправлениями, изменениями отдельных деталей по ходу создания картинке. Рисунки мальчиков носили техническую направленность, а процесс деятельности характеризовался стремлением к самостоятельности, чего нельзя было сказать о девочках, чаще нуждавшихся в оценке и одобрении со стороны учителя.

На 7-м занятии предлагалось задание: а) самостоятельное создание примитива, выполнение рисунка с использованием этого примитива; б) сжатие получившейся картинке определенное количество раз, запись нового примитива на ленточку; в) выполнение еще одного рисунка, состоящего в изменении предыдущего с помощью операции объединения, аффинных преобразований и применения операции инверсии (негатива).

Анализ последнего рисунка позволяет сделать выводы.

1. Тематика созданных примитивов свидетельствует о явно выраженной

половой дифференциации, что отмечалось и на занятиях по двум предыдущим рисункам.

2. Промежуточные и итоговые картинки (около 70 %) показали овладение учащимися способом действий по достижению композиционного равновесия рисунка, в частности его симметрии и центрирования. Это соответствует третьему типу композиции (установленной для некомпьютерного варианта Ю. А. Полуяновым) по анализу продуктов деятельности детей, рисующих регулярно, работы которых экспонировались на всесоюзных и международных выставках. При третьем типе «изображения упорядочены между собой и в формате листа по принципам симметрии». При использовании всех структурных частей «рабочего поля» (особенно центра и края) последовательно закрепляются связи по симметрии, обуславливающие композиционное равновесие рисунка; замысел имеет сюжетную форму.

Следует отметить, что рисунки, выполненные на начальных и средних этапах работы в кружке, соответствовали второму типу композиции, т. е. «изображения были упорядочены между собой, но выполнены не в формате листа, использовался верх и низ листа, возникали связи по ритму, реже — по симметрии, функциональные связи передавались соседством изображений; замысел складывался из перечисления предметов, связанных по своим реальным функциям и соседством изображения на рисунке. Это и объясняется тем, что на первых занятиях ученики приобретали навыки работы с клавиатурой компьютера, а к третьему типу композиции дети пришли после построения рисунков, где применялись различные виды аффинных преобразований (прежде всего симметрий). При этом акцент с исполнительной части способа действий постепенно смещался на ориентировочную часть при осуществлении различных преобразований (операциональный аспект моделирования сводился к нажатию управляющих клавиш). К появлению композиции третьего типа приводила и серия «Незнайкиных задач». Так как учебная цель компьютерного кружка состояла в формировании представле-

ний об аффинных преобразованиях в ходе свободного композирования, ученикам I класса был предложен цикл «Незнайкиных задач», в которых в игровой форме (игровой персонаж Незнайка — один из применявшихся приемов драматизации учебной задачи) моделировались параллельный перенос, различные виды симметрии, повороты, подобие и соответствующие особенности композиционного равновесия рисунка. В разных сериях Незнайка, путешествуя по стране «Геометландии» (парафраз «Матланда», построенного С. Пейпертом в системе ЛОГО), устраняет разрушения, причиненные «разбойницей-резинкой», помогает жителям сказочной страны, выступает в роли спортсмена и т. п. Как отметил В. В. Давыдов, способы художественной деятельности не могут быть выражены в понятийной форме, в виде формул, схем и т. д. Усвоение новых способов выразительного изображения решается путем постановки перед учащимися особых учебных задач, основанных на применении детьми учебного действия моделирования, на примере овладения ими способом действий по достижению композиционного равновесия рисунка, в частности его симметрии. В серии задач на параллельный перенос (2-е занятие), выполняемой операционально с помощью клавиш перемещения, Незнайка продолжает гирлянду, заменяет лопнувшее колесо, сажает аллею деревьев. В серии задач на центральную симметрию (4-е занятие) он дорисовывает различные части геометрических фигур, причем недостающие части симметричны оставшимся относительно центра симметрии. Выполнение задач этого вида легко осуществляется благодаря усвоению операций поворота, горизонтальной и вертикальной симметрии. При решении дети использовали два способа: достраивали недостающие части фигур, имеющих центр симметрии, либо с помощью композиции двух поворотов по  $90^\circ$  (1-й способ), либо используя вертикальную (3-я задача) или горизонтальную (1-я, 2-я задача) симметрию (2-й способ).

В 3-й серии задач моделировались ситуации на осевую симметрию (7-е занятие). Незнайка изобретает способ

упрощенного написания букв или цифр, используя их осевую симметрию, он борется с разрушениями резинки, дорисовывая половинки вытертых фигур, видит отражения различных объектов в воде, убеждаясь в том, что они симметричны относительно некоторой оси (например, относительно мачты той лодочки, с борта которой смотрит в воду Незнайка). При решении этой серии задач учащиеся выделяли различные виды осевой симметрии (горизонтальную и вертикальную), учились строить фигуры, симметричные данной относительно некоторой оси (причем эта ось могла быть как заданной, так и самостоятельно найденной учащимися), знакомились со свойствами осевой симметрии (равенство фигур, переориентация, равноудаленность между одноименными точками фигур, имеющими ось симметрии, и т. д.) и ее признаками.

В 4-й серии учебных задач (8-е занятие) моделировались более сложные композиции объектов с использованием как осевой, так и центральной симметрии, причем учащимся предлагалось самостоятельно найти центр или ось симметрии (свойства симметрии «обыгрывались» в соответствующей драматизации). Кроме описанных выше серий «Незнайкины задачи» использовались при моделировании других видов аффинных преобразований — в игровых задачах рассматривались также подобие, композиция нескольких поворотов. Очевидно, что решение серии учебных задач способствовало выработке тенденции к центрированию — она была выражена в компьютерном варианте ранее, чем при использовании традиционных средств рисования, и проявилась на завершающем этапе работы в среде системы ЭЛЬКОН.

#### Выводы.

1. Реализация компьютеризованных учебных объектно-ориентированных сред обеспечивает процесс целенаправ-

ленного планирования и организации учебной деятельности младших школьников.

2. Включение в компьютеризованную учебную среду теоретико-множественных операций и аффинных преобразований позволяет:

реализовать достаточно полный набор функций для свободного манипулирования учебными моделями объектов; реализовать композиционную структуру учебного диалога;

обеспечить высокий уровень интерпретации каждой строящейся учащимся модели объекта (объектов) с использованием понятий ФУНКЦИЯ и МНОЖЕСТВО.

3. Объектно-ориентированная учебная деятельность учащихся носит игровой характер и обеспечивает элементы творчества.

4. Прагматика применения компьютеризованных учебных объектно-ориентированных сред заключается в подборе и формировании широкого класса задач по определенным предметным областям с выделением структурированных множеств объектов и определением над ними соответствующего класса функций.

#### Литература

1. Давыдов В. В. Проблемы развивающего обучения. М. Просвещение, 1986.
2. Полуянов Ю. А. Методы изучения изобразительного творчества детей // Новые исследования в психологии. 1980. № 2. С. 68—73.
3. Papert S. Mindstorms. Children, Computers, and Powerful Ideas. Brighton: Harvester Press Limited, 1980.
4. Житомирский В. Г., Шеврин Л. Н. Геометрия для малышей. М.: Педагогика, 1975.

М. ДВОРИНА, А. ДОВГЯЛЛО,  
Э. ИВАХНЕНКО, Е. КОБЕРНИК,  
Е. МАРГУЛИС, А. СТРИЖАК,  
А. ЧИГОРЕВ

## Курс информатики: состояние, методика и перспективы

В данной статье приводятся и комментируются наиболее распространенные мнения учителей, высказываемые ими во время встреч, проводимых автором в различных городах страны, и во время учебы в МГИУУ. (Учителя работают по учебнику автора в машинном варианте.)

### Каково содержание курса информатики?

Большинство учителей считают, что в курс информатики должны входить следующие темы:

элементы программирования (с использованием языков Бейсик, Фокал, Рапира или Е-практикум);

элементы алгоритмизации (с использованием для описания блок-схем или алгоритмического языка из учебника А. П. Ершова);

методика решения задач на ЭВМ (с составлением соответствующих алгоритмов, программ и получения результатов решения на ЭВМ с их помощью);

работа на ЭВМ (с редакторами текстов, графическими редакторами, электронными таблицами и т. п.);

элементы информационных технологий (базы данных, информационно-поисковые системы и т. д.);

элементы основ вычислительной техники и ее применений (логические основы, системы счислений, архитектура и типология ЭВМ и др.).

### Что такое компьютерная грамотность?

Обсуждение этого вопроса практически всегда заканчивается победой точки зрения, что компьютерная грамотность — это практические умения подготовки и редактирования текстов, чертежей, рисунков, выполнения простейших расчетов и поиска информации на ЭВМ, а не навыки программирования на ЭВМ.

Кроме того, обсуждение этого вопроса, как правило, переходит на обсуждение основных целей курса информатики

в средней школе и дискуссию об информационной культуре, алгоритмическом, логическом и алогическом мышлении. В этой связи хотелось бы остановиться на этих вопросах подробнее.

Под информационной культурой прежде всего понимаются умения получать, накапливать, искать, собирать и передавать информацию с помощью ЭВМ, используя базы данных и различные информационные системы. Эти технические умения будут все более важны для людей по мере развития информатизации общества.

В информационную культуру в широком ее понимании кроме чисто технических навыков должны входить и умения выражать свои мысли и идеи в литературной, графической и художественной форме с использованием ЭВМ. Более того, в эту культуру должны входить и умения общаться и сотрудничать с другими людьми.

Однако сомнительно, что для развития у людей такой культуры необходимо формирование у них алгоритмического мышления, которое в первом и пока единственном методическом пособии для учителей информатики раскрывается как умение читать, понимать и исполнять алгоритмы (предписания для исполнителей).

Другой крайностью, на наш взгляд, является реальная опасность формирования у детей алогического мышления, выражающегося в неумении формулировать и высказывать свои мысли, неумении планировать свою деятельность, неумении находить контакт с людьми и т. д. Лучшими средствами для закрепле-

ния таких качеств является ничем не ограничиваемое использование в курсе информатики языков программирования типа Бейсик или Фокал с инородной лексикой.

### Какие темы в каких классах можно изучать?

Тема «Работа на ЭВМ» признается всеми учителями как бесспорно необходимая. Однако мнения о начале ее изучения расходятся: некоторые считают, что эту тему нужно изучать в XI или X классе, большинство — что это нужно делать в VIII или VI, а радикалы — в младших классах или даже в детских садах.

Тема «Элементы информационных технологий» признается как полезная для всех учеников. Ориентировочное время освоения новых технологий — IX—X классы. Однако ее изучение упирается в общий вопрос: где приобрести необходимое программное обеспечение для имеющихся в школе компьютеров?

Тема «Элементы программирования» — крайне важная. В то же время большинство учителей считают, что изучение программирования в полном объеме нужно проводить в спецшколах и на факультативах. Однако все соглашались с тем, что изучение начал программирования нужно для всех учеников.

Тема «Элементы алгоритмизации» многими учителями преподается с использованием блок-схем, так, как этому до сих пор учат студентов в технических вузах и многих университетах. Большинство учителей не знают, что такое структурное программирование, структурный подход к разработке программ и зачем профессиональные программисты для этих целей используют псевдокод.

Тема «Решение задач» поддерживается учителями, имевшими опыт преподавания математики и физики, и вызывает некоторое сопротивление учителей с инженерным образованием. Очень малое число учителей имеет практический опыт решения научных или инженерных задач на ЭВМ.

Тема «Элементы основ вычислительной техники» вызывает дискуссии, поскольку разные учителя имеют различные точки зрения на содержание этой темы и жалуются на отсутствие или нехватку у них необходимого программного обеспечения.

Базой для обучения основам информатики, вычислительной техники и программирования при подготовке специалистов как за рубежом, так и у нас в стране всегда было, есть и будет более глубокое изучение математики и развитие конструктивного логического мышления.

Логическое мышление — это прежде всего умения рассуждать, доказывать, подбирать факты, аргументы и обосновывать предлагаемые решения. По нашему глубокому убеждению, без этих умений невозможна никакая результативная деятельность и никакое сотрудничество — человеку остается только бездумное исполнительство.

Традиционно развитие логического мышления считалось прерогативой школьного курса математики. Однако в методических пособиях для учителей математики последних лет красной линией проходит установка на формирование у всех учащихся не логического, а алгоритмического мышления (?!).

При таком подходе эффективное освоение новой техники, новых технологий, новых форм производства и тем более выявление и развитие у детей творческих способностей, на наш взгляд, невозможны.

### Нужно ли и как учить программированию?

На этот вопрос практически все учителя информатики говорят: нужно изучать определенные элементы программирования. Основной вопрос сводится к тому: в каком объеме и как? Большинство учителей используют для этого язык Бейсик, имеющийся в различных диалектах на всех школьных ЭВМ. Главной методической проблемой для них является подбор тем и задач.

Однако до сих пор для большинства учителей остается открытым главный методический вопрос — как за короткий

сеанс работы на ЭВМ во время урока обеспечить отладку всеми учениками всех их программ? Настоящие трудности возникают тогда, когда ученики работают на ЭВМ со своими собственными программами, которые составлены самостоятельно и которые в силу этого содержат много ошибок.

Такого рода занятия являются серьезным испытанием для учителей — они должны оперативно найти ошибки в ученических программах и предложить им правильные исправления. В среде профессиональных программистов решение данной проблемы найдено в 70-х гг., на фирме IBM.

В этих целях организуются бригады программистов и каждый из них обязывается кроме текста программы представлять еще два описания — спецификацию (сценарий) программы и описание логики ее работы в форме структурированных алгоритмов, записанных на псевдокоде, т. е. на обычном разговорном языке со строгим соблюдением правил структурного программирования.

Все алгоритмические и синтаксические ошибки при такой организации работ бригадой программистов выявляются за столом без выхода на ЭВМ. Отладка программ при этом сводится к их техническим испытаниям и проверке правильности результатов решения поставленных задач.

Данная технология успешно переносится в ученическую среду, допуская перекрестную проверку программ, коллективную работу над программами, работу по самостоятельным заданиям и даже организацию конкурсов на лучшую программу.

Сложность заданий для учеников очень легко варьировать. Простейшими, но в то же время весьма увлекательными являются графические и диалоговые программы. Их составление не требует глубоких математических знаний, а в основном изобретательности и фантазии.

В этих заданиях необходимо самостоятельно придумать занимательные картинки, вежливые формы диалога, смешные ситуации и т. п. Эти задания доступны для всех ребят и не только в X—XI, но и в VIII—IX классах. В

любом случае все 100 % учеников, выполнив серию заданий, полностью осваивают навыки самостоятельного составления сценариев, алгоритмов и отладки программ на ЭВМ.

## Что такое — информатика?

На этот очень существенный вопрос многие учителя отвечают, что информатика — это новая индустрия, индустрия накопления, передачи и представления информации с использованием сетей ЭВМ, персональных компьютеров и всевозможной информационной техники.

На более серьезный вопрос — является ли информатика наукой? — отрицательный ответ дают не только и не столько ведущие методисты, сколько ведущие ученые-кибернетики. Ясно, что самостоятельное развитие новой индустрии, новых технологий невозможно без самостоятельных теоретических исследований.

Создание ядерной и космической промышленности у нас в стране в 40—50-х гг. было бы невозможно без развития у нас же в 20—30-х гг. теоретической физики. Успехами наших физиков-теоретиков и физиков-экспериментаторов мы обязаны не только сохранению мира на Земле, но и началу кардинальных преобразований в нашей стране.

Успехи в развитии собственной отечественной индустрии информатики и вычислительной техники невозможны без соответствующей подготовки своих собственных специалистов и ученых в этой области. Для этого, естественно, необходимы надлежащие изменения в организации отечественного образования и науки.

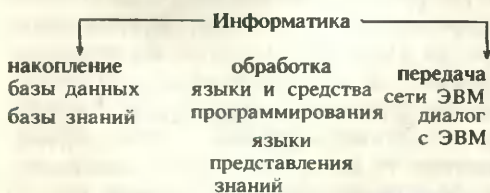
Соответственно необходимо уточнение целей и содержания информатики в средних школах. Математика, физика, биология, химия, история и другие школьные дисциплины входят в учебные планы средних школ постольку, поскольку существуют соответствующие фундаментальные науки и элементарные знания и представления из этих наук необходимы всем.

Положение информатики в средних школах останется двусмысленным до тех

пор, пока не будет сформулировано и не станет общепризнанным содержание информатики как фундаментальной научной дисциплины.

Родоначальниками информатики в нашей стране были академик В. М. Глушков, написавший в конце 70-х гг. книгу о безбумажной информатике, и академик А. П. Ершов, под руководством и при непосредственном участии которого был написан первый школьный учебник по информатике.

Согласно определению А. П. Ершова, информатика — это наука, изучающая законы и методы накопления, обработки и передачи информации с помощью ЭВМ. Рассматривая основные средства накопления, обработки и передачи информации с использованием ЭВМ, можно полностью уяснить содержание такого технического или, точнее, технологического подхода к информатике как научной дисциплине (см. схему).



Кроме компьютерных средств мы еще долгое время для накопления информации будем использовать обычные архивы, библиотеки и бумажные носители.

В качестве основных средства обработки информации в ЭВМ еще долгое время будут использоваться различные языки программирования и обработки данных (Паскаль, Бейсик, Ада, Си и т. п.). В перспективе к ним присоединятся языки представления знаний, с помощью которых будут создаваться на ЭВМ различного рода экспертные системы.

Основными средствами передачи информации станут сети ЭВМ, которые наравне с радио, телевидением, телефоном и телеграфом станут средствами массовых информационных коммуникаций. При этом новейшие компьютеры будут иметь развитые средства ведения диалога с людьми в звуковой, графической

и иных сенсорных формах.

Комплекс научных вопросов создания и развития различных информационных средств и образует предмет информатики как научной дисциплины.

В то же время организация использования этих технологий не может затронуть вопросы протекания информационных процессов в общении и работе людей, в жизни общества, работе технических устройств и функционировании живых организмов. Включение в информатику этого круга вопросов превратит ее в фундаментальную естественнонаучную дисциплину.

### Каковы перспективы информатики в школе?

Главный вопрос учителей — сохранится ли в школе курс информатики? В тех школах, где уже имеются кабинеты вычислительной техники, учителя информатики считают безоговорочно: нужно сохранять. Но остаются вопросы: где взять учебники, достать программное обеспечение? Как получить методическую литературу? Где можно пройти переподготовку по машинному варианту обучения?

В школах, не имеющих компьютеров, варианты решений различны — от отказа от ведения курса и перевода его в кружки и факультативы до продолжения безмашинного варианта или организации и посещения межшкольных центров для проведения практикумов на ЭВМ.

В школах, где кабинеты установлены уже давно, учителя проводят самые различные эксперименты по раннему изучению информатики, используя для этого кружки, факультативы или часы за счет трудового обучения. Естественно, что практически у всех учителей имеются положительные результаты, так как дети с большой охотой работают в кружках, участвуют в экспериментах и тем более с компьютерами.

Во многих из этих школ проходят эксперименты по раннему освоению компьютерной грамотности и начал программирования. При этом используются различного рода исполнители и пакеты деловых программ — редакторы текстов, графические редакторы.

В старших классах начинаются эксперименты по использованию электронных таблиц и языков программирования для решения различного рода математических задач — вычислительных, геометрических, исследований графиков, функций и т. п.

В наиболее продвинутых школах приступили к решению задач информатизации школы, используя ребят и компьютеры для подготовки информации и ведения различного рода архивов на ЭВМ. Широко используются различные упрощенные базы данных, информационные подсистемы.

В то же время большинство учителей сходятся во мнении, что освоение информатики с ЭВМ целесообразно начинать не с X, а с VIII класса. При этом некоторые учителя считают целесообразным начинать с исполнителей и ориентируются на новый учебник.

Другие учителя считают, что начинать следует с изучения работы на ЭВМ с редакторами, электронными таблицами, а затем уже с исполнителями и языками программирования.

Все учителя признают, что более глубокое изучение программирования необходимо проводить в старших классах, но при этом дифференцированно — для учеников с математическими, техническими и гуманитарными наклонностями: углубить для математических классов, сохранить в определенной мере для классов с технической ориентацией и упростить для классов с гуманитарной.

### Где приобрести программное обеспечение?

Для созданного нами пробного учебника по информатике необходимы следующие программы:

- 1) клавиатурный тренажер — для освоения клавиатуры ЭВМ;
- 2) редактор текстов — для редактирования текстов на ЭВМ;
- 3) графический редактор — для выполнения рисунков и чертежей с помощью компьютеров;
- 4) электронные таблицы — для выполнения простейших расчетов на персональных компьютерах;

5) табличная база данных — для накопления и поиска информации на ЭВМ в простейшей табличной форме;

6) диалоговый интерпретатор Бейсика — для составления и отладки на ЭВМ простейших диалоговых программ;

7) диалоговый интерпретатор Пролога — для составления и апробации простейших баз знаний и простейших моделей «искусственного интеллекта».

Все перечисленные программы в настоящее время есть практически для всех основных типов школьных ЭВМ: БК-0010, «Корвет», УКНЦ, «Агат», «Правец» и «Ямаха». Тиражирование этих программ вместе с методической литературой предполагается проводить через Всесоюзную ассоциацию учителей информатики. Параллельно ведутся переговоры о тиражировании необходимого программно-методического комплекса через республиканские министерства и комитеты по науке и образованию.

С методическими материалами можно познакомиться по статьям, опубликованным в 1989—1990-е гг. в журнале «ИНФО». В них изложена методика, связанная с первой половиной учебника, — методика обучения компьютерной грамотности и началам программирования. Методика обучения технике решения задач и основам информатики издана Московским городским институтом усовершенствования учителей. Пока ее вместе с задачиком и программным обеспечением могут получить только те учителя, которые приезжают в МГИУУ на курсы освоения методики обучения информатике по машинному методу.

### Зачем нужен язык Пролог?

Язык Пролог в нашем учебнике первоначально появился как тренажер для освоения элементов логики. Дело заключается в том, что изучение логических основ ЭВМ, основ алгоритмизации и логических основ поиска информации дидактически невозможно без предварительного изучения таких понятий, как логические связки и логические операции «и», «или» и «не». Это «кирпичики», из которых строятся логические схемы ЭВМ, алгоритмы, программы и запросы к базам данных. В

30



противном случае возникает масса неясностей и недоразумений.

Более серьезным является вопрос о роли логики в курсе информатики и в образовании вообще. Недостатком, на мой взгляд, А. П. Ершова была потеря в учебнике информатики разделов, связанных с изучением элементов и пропедевтики математической логики. К сожалению, многие созданные позднее учебники воспроизвели ту же ошибку.

Роль логики в образовании и науке трудно переоценить. Не случайно самый первый учебник по логике в России был написан не кем иным, как Михаилом Васильевичем Ломоносовым — создателем Московского университета и одним из основоположников отечественной науки.

Отсутствие логики в школьной информатике и в школьной математике — серьезный тормоз в изучении и современной вычислительной техники, и современной науки, и современной математики. И уж тем более значительным тормозом является отсутствие в вузах и университетах хороших курсов по содержанию и математической логике.

Выход из сложившегося положения мы видим во внедрении в средние школы курса информатики, в котором в удачной дидактической форме с упражнениями на ЭВМ наши дети изучат в переплетении элементы содержательной, математической и компьютерной логики. Соответствующим, и притом удачным, средством для упражнений на ЭВМ является язык Пролог, позволяющий решать огромное число содержательных, математических и информационно-логических задач.

Более того, возможности демонстрировать на ЭВМ с помощью языка Пролог в простейших формах методы ведения диалога, накопления и поиска информации, обработки данных и решения самых разнообразных задач превращают этот язык в самое удобное средство для изучения идей, методов и средств информатики. И уж тем более на простейших примерах с помощью языка Пролог можно показать ученикам все наиболее важные вычислительные и интеллектуальные возможности

современных и перспективных компьютеров.

### Удовлетворены ли мы своим учебником?

Подготовленный нами и выпущенный издательством «Просвещение» в 1989 г. пробный учебник по информатике полностью соответствует идеям и концепциям рукописи, представлявшейся на конкурс учебников в 1987 г. Эта рукопись, победившая в конкурсе, была написана в жестком соответствии с конкурсной программой.

Первоначально учебник был ориентирован на математические школы и классы с углубленным изучением математики. При этом материалы учебника многократно были проверены в физико-математической школе МИЭМа, в которой в течение многих лет работали авторы. Школа была создана в 1966 г. профессором Л. В. Тарасовым, автором одной из новейших концепций интегративного подхода к образованию в средних школах.

Массовые эксперименты по нашему учебнику и нашей методике потребовали серьезной доработки рукописи учебника. Но главное состоит в том, что методика, положенная в основу учебника, оказалась удачной и не пре-

терпела каких-то существенных изменений. Суть же ее состояла и будет состоять в обязательном взаимоувязывании на уроках информатики элементов теории с практическими занятиями на ЭВМ.

В то же время сохранение старой программы с размытым определением предмета информатики и с разделением теоретических, практических и общеобразовательных сведений являлось для нас главным тормозящим фактором. Новая редакция учебника, над которой уже завершена работа, написана полностью в соответствии с методикой обучения, отработанной в общеобразовательных средних школах, и в полном соответствии с новым уточненным пониманием информатики как учебного предмета и научной дисциплины.

**Школьные компьютерные классы  
нового поколения  
на основе персонального компьютера**

**«Сириус»**

| на смену компьютерам БК-0010 и «Агат»

**Компьютер «Сириус» — обладатель диплома  
ВДНХ СССР 1990 г.**

ПК «Сириус» наиболее полно отвечает современным требованиям, предъявляемым к школьному компьютеру: *безотказность в работе, простота в обращении, цветная графика высокого разрешения, хороший дизайн и богатое программное обеспечение.*

Высокая надежность нового класса достигнута благодаря применению импортной комплектации, использованию ударопрочного корпуса и специальной высокоскоростной помехозащищенной локальной сети.

Программная совместимость с известным английским школьным компьютером «ZX Spectrum» позволяет использовать обширную библиотеку программ — более 5000 наименований.

Таковыми возможностями не обладает ни один отечественный школьный компьютер.

**Технические характеристики ПК «Сириус»:**

тип микропроцессора — Z80A (США);

быстродействие, оп/с — 800 тыс.;

адресуемая память, Кбайт — 64;

разрядность шины данных — 8;

разрядность шины адреса — 16;

емкость ОЗУ, Кбайт — 48;

емкость ПЗУ, Кбайт — 16;

отображение информации на экране цветного монитора — 256×192 точки в 16 цветах (при двух градациях яркости), в символьном режиме — 24 строки по 32 символа;

программное обеспечение — Бейсик (ПЗУ), Си, Лого, Форт, Паскаль, Пролог, ассемблер, текстовые и графические редакторы, база данных, около 600 системных, 40 школьных обучающих и 1500 игровых программ, которые тоже можно купить у нас.

Класс состоит из рабочего места учителя и рабочих мест учеников, связанных локальной вычислительной сетью.

Поставка классов осуществляется по системе «под ключ».

**Заявки направлять по адресу: 249031, Калужская обл., пос. Прогва, а/я № 113, НТК «Системотехник».**

**Телефон для справок: (08432) 22-182.**

# ЧТО МОЖЕТ ЭВМ?

## Пора передохнуть

Конечно, подсчитать пульс — не велика проблема, и возлагать ее на компьютер вроде и неловко, но одна из японских фирм решила все-таки наладить выпуск автоматических цифровых приборов для измерения частоты пульса и, похоже, не прогадала. Симпатичный по дизайну и удобный в обращении пульсомер пользуется у покупателей спросом.

Чтобы воспользоваться прибором, необходимо просунуть палец в широкое кольцо пульсомера, отрегулировать прижим датчика и включить питание. Через несколько секунд на экране жидкокристаллического дисплея покажутся цифры, указывающие частоту сердечных сокращений.

## Миллион секретов

Видеть такую надпись на упаковках с дверными замками мы уже привыкли. Скоро подобные слова можно будет отнести и к новым высокоточным модульным разъемам, которые, как ожидается, станут международным стандартом для телекоммуникационных и локальных линий связи. Эти разъемы выбраны фирмой Du Pont Electronics для шины Futurebus+, которая станет предшественником следующего после VMEbus и Multibus II поколения стандартных шин.

Новый разъем получил название «Metral» и, имея модульную структуру, способен расширяться. Несмотря на своеобразное конструктивное исполнение, разъем полностью совместим с разъемами AT&T Microelectronics (США) и Ericsson Telecom (Европа). Каждый модуль разъема в плане имеет вид матрицы на 24 (4×6) контакта, расположенных с шагом 2 мм по обеим координатам, и включает различ-

ные функциональные компоненты, позволяющие стыковать между собой две платы и плату с кабелем для сопряжения с внешними устройствами. Модули можно собирать в блоки на 24, 48, 96 или 192 контакта. К такому блоку можно подключать по несколько кабелей без потери контактных рядов на межмодульные промежутки.

А где же секреты? В каждом разьеме. Это уникальный кодовый пластинчатый ключ, закрепляемый на прямой и ответной частях разъема, который не позволит включить кабель не в свое гнездо, даже если вы этого захотите.

## Мыши, коты и другие

Компьютерная мышь — хорошо известное устройство. Менее известны его заменители — не такие, как сорокакнопочная мышь Powermouse 100, а качественно иные устройства.

Одно из них — «кот», появившийся довольно давно, но не получивший широкого распространения. Фактически это перевернутая вверх ногами мышь: устройство, из верхней плоскости которого выступает часть находящегося внутри шара. Вращая шар ладонью, перемещают курсор по экрану.

Другой вариант, пока не удостоенный «бытового» названия — вмонтированный в край клавиатуры цилиндр. Его можно вращать кончиками пальцев и нажимать. Это устройство удобно для использования в портативных компьютерах, не имеющих достаточно свободного места для вмонтирования «кота» и зачастую эксплуатируемых на коленях, по которым не очень-то удобно катать мышь.

Кстати, необходимость большой плоскости для работы не единственная особенность мыши. Даже самая маленькая мышка весит фактически несколько килограммов — ведь она перемещается вместе с управляющей ею рукой, причем руку приходится держать на весу! Сначала это незаметно, но попробуйте почертить час-другой...

## И время, и деньги

В то время, когда мы только начинали борьбу с отечественными бюрократами, американские электронщики работали над устройствами, облегчающими жизнь своим.

Один из таких приборов фирмы ENEWS представляет из себя специализированный компьютер размером с обычную книгу, снабженный жидкокристаллическим дисплеем на несколько строк и встроенным принтером. Компьютер ориентирован только на работу с базами данных и не требует специальной подготовки. От предыдущих моделей он отличается тем, что позволяет хранить, редактировать и быстро находить запись о сотруднике по фамилии, телефонному номеру и полному почтовому адресу. Кроме того, электронная база данных может хранить небольшое текстовое сообщение для любого зарегистрированного в ней абонента.

Не менее важным для хорошей организации дела устройством, разработанным той же американской фирмой, является электронный фиксатор времени. Внешне этот прибор похож на настольные электронные часы, но помимо демонстрации времени он печатает на карточке, опущенной в прорезь в корпусе, время и дату, производит подсчет зарегистрировавшихся сотрудников и собирает связанную с этим статистику.

Тем, кто воспользуется новой оргтехникой, фирма, подтверждая старую истину «время — деньги», обещает немедленную экономию и того, и другого.

## Бейсик жил, Бейсик жив...

Отмечая 25-ю годовщину создания этого языка, фирма Microsoft сообщила, что, по ее подсчетам, на Бейсике сейчас программируют около 4 млн. человек — в 3 раза больше, чем на всех остальных языках, вместе взятых. Начиная с 1985 г. фирмой было продано более 500 тыс. экземпляров пакета КвикБейсик, и спада популярности этого языка не ожидается.

33

М. ГУТКИН, А. ИВАНОВ, С. НОВОСЕЛЬЦЕВ, С. ХРИСТОЧЕВСКИЙ  
ИПИ АН СССР, Москва

## Учебные персональные ЭВМ

### Отечественные КУВТ

**34** Опыт использования ПЭВМ в обучении, накопленный за последние годы, позволяет сделать следующие выводы.

1. Концепция поставки в учебные заведения не отдельных ПЭВМ, а комплектов учебной вычислительной техники (КУВТ) была правильной. Каждый КУВТ состоит из одного рабочего места преподавателя (РМП) и некоторого количества (от 9 до 15) рабочих мест учеников (РМУ). В состав каждого рабочего места входит ПЭВМ, а в состав РМП, кроме того, входит печатающее устройство (ПУ) и накопитель на гибких магнитных дисках (НГМД). Все рабочие места объединены между собой локальной сетью, которая, с одной стороны, обеспечивает доступ к дефицитным ресурсам — ПУ и НГМД, а с другой — облегчает преподавателю управление учебным процессом.

2. В настоящее время в нашей стране в учебном процессе используются следующие виды КУВТ: КУВТ-86, УКНЦ, «Корвет», «Агат», УКНЦ-01. В ряде республик и областей используются другие УПЭВМ и КУВТ, но ввиду их сравнительной малочисленности в данной статье они не рассматриваются.

3. Совместимость между различными моделями КУВТ как на аппаратном (разные разъемы), так и на программном уровне (разные системы команд и разные операционные системы) практически отсутствует. Теоретически возможен лишь перенос на магнитофонной

кассете исходных текстов программ, написанных на языке Бейсик.

Рассмотрим характеристики перечисленных КУВТ.

### КУВТ-86

КУВТ-86 включает в себя одну ПЭВМ типа ДВК-2М (РМП) и 15 ПЭВМ БК-0010Ш (РМУ), объединенных локальной сетью типа общая шина через последовательный интерфейс типа СТЫК-С2. По ряду параметров КУВТ-86 не удовлетворяет не только внутрисоюзному уровню развития средств учебной вычислительной техники (не говоря о международном), но и ГОСТ 27201—87, регламентирующему основные требования к ПЭВМ.

В компьютере ДВК-2М используется 16-разрядная одноплатная ЭВМ «Электроника-80» на основе микропроцессора КМ1801ВМ1 и 56К байт ОЗУ, плата контроллера дискового, системный блок, спаренный дисковод НГМД-6022 и профессиональный алфавитно-цифровой дисплей «Электроника-15ИЭ00013». Монитор не допускает графических режимов, контроллер монитора представляет собой просто последовательный интерфейс, видеоОЗУ отсутствует. Имеется контроллер параллельного интерфейса типа ИРПР-М.

В БК-0010Ш используется одноплатная ЭВМ на основе микропроцессора КМ1801ВМ1 с производительностью

300 тыс. оп/с. 32К байт ОЗУ, из которых видеоОЗУ занимает 16К байт, а 16К байт предоставляется программе пользователя. Почти 32К байт ПЗУ (8К) заняты системным монитором и драйверами, около 24К байт занимает интерпретатор-компилятор языка Бейсик, часть которого может быть отключена через системный слот и заменена ПЗУ тестов. Объем памяти, предоставляемой пользователю, может быть увеличен за счет части видеоОЗУ (12К байт), однако при этом графика, по сути дела, исключается. В качестве ВЗУ предусматривается бытовой кассетный магнитофон. Контроллер НМЛ допускает запись/чтение со скоростью 1200 бод. Контроллер монитора позволяет подключать как черно-белый («Видео»), так и цветной монитор, но пользователю доступен только выход «Видео» (выход аналогового RGB не выведен на внешний разъем). Имеются интерфейсы: параллельный асинхронный на 16 линий разделенного ввода/вывода, последовательный — в виде отдельного устройства, подключаемого через системный слот (поставляется в составе КУВТ для организации локальной сети). Есть разъем системной шины. БК-0010Ш допускает графические режимы: одноцветный с  $512 \times 256$  точками и  $256 \times 256$  точками с четырьмя цветами.

Ниже перечислены основные недостатки КУВТ-86.

1. Использование принципиально разных машин в качестве РМП и РМУ. Объединяет ДВК-2М и БК-0010Ш по сути дела только одинаковый микропроцессор, используемый в обеих конструкциях. Разработчики не позаботились даже о соответствии адресов внешних устройств. Это, а также принципиально различные карты памяти и отсутствие у РМП графических режимов привели практически к полной несовместимости ПЭВМ между собой. По сути дела, РМП используется лишь как файлсервер, непомерно дорогой для таких целей, не обеспечивающий обмен информацией между рабочими местами и других функций локальной сети.

2. На рабочих местах используются компьютеры, которые в лучшем случае могут быть рассмотрены как игровые

(и то с параметрами, далекими от достигнутого в этой области уровня). Слишком малое ОЗУ (недостаточное даже для Бейсика) создает трудности для непрофессионального программиста.

3. Форматы записи данных на дискете присущи только ДВК-2М и не совместимы ни с одним стандартным форматом.

4. Надежность КУВТ-86 не выдерживает никакой критики. Наиболее часты отказы дисководов; регулярны сбои клавиатуры (дребезг контактов).

## КУВТ УКНЦ

КУВТ УКНЦ состоит из однотипных ПЭВМ «Электроника МС0511», объединенных в локальную сеть. Базовый класс предусматривает одно РМП и от 3 до 12 РМУ. Каждая ПЭВМ построена по двухпроцессорной двухмагистральной архитектуре. В качестве центрального (ЦП) и периферийного (ПП — по сути дела процессор ввода/вывода) процессоров используются микропроцессоры КМ1801ВМ2, связанные через скоростной байтовый параллельный интерфейс и регистры. Совмещенная 16-разрядная шина адреса/данных позволяет каждому из процессоров адресовать 64К байт. Имеется приоритетная система прерываний. Наличие у БИС КМ1801ВМ2 двух режимов работы: «USER» и «HALT» позволяет включать в адресное пространство различные области памяти и организовывать арбитраж. Для решения задач арбитража используется также буфер магистрали ПП. Производительность ЦП (по ТУ) 700 тыс. оп/с, ПП — 500 тыс. оп/с.

Объем ОЗУ составляет 192К байт, из них 64К принадлежат ЦП, 32К — ПП, 96К — видеоОЗУ. Управление каждым ОЗУ осуществляется своей БИС управления. Из-за ограниченности адресного пространства каждого из процессоров часть памяти, в том числе видеоОЗУ, адресуется через регистры.

Системное ПЗУ объемом 32К байт принадлежит адресному пространству ПП и содержит программу начального загрузчика и тесты.

Имеется 3 системных слота расширения, один из которых предназначен для подключения контроллера локальной сети.

Контроллер НГМД реализован в виде отдельного устройства, подключаемого через слот. Форматы 5.25" дискеты совместимы с форматами ЭВМ серии PDP — 11 фирмы DEC. Контроллер НМЛ позволяет осуществлять запись/считывание информации со скоростью 1200 бод.

Контроллер клавиатуры обслуживает матрицу 8×12 клавиш встроенной клавиатуры. Имеются 10 функциональных клавиш.

Контроллер монитора позволяет подключать как черно-белый («Видео»), так и цветной (аналоговый RGBY, Y — яркость строки) мониторы.

ПЭВМ включает в себя таймер и генератор звука, работающий в полосе частот 60 Гц — 16 кГц.

36 Сеть типа «Кольцо» передает информацию в байтовом формате со скоростью 57,6К бод.

Графические режимы позволяют адресовать экран до 640×288 точек 8 цветами из 16, время закрашивания экрана около 1 с. Доступен текстовый режим: 26 строк по 80 символов в строке.

КУВТ стандартно комплектуется мониторами «Электроника-6105», сдвоенным дисководом НГМД-6022 емкостью 2×400К байт и принтером ROBOTRON — 6329.01.

С точки зрения пользователя несомненными достоинствами КУВТ УКНЦ являются: унификация составляющих его ПЭВМ (в отличие от КУВТ-86); удобная для программирования, полная и симметричная система команд процессора; ориентация на интегральные схемы большой степени интеграции (полузаказные БИС серии 1515); малое количество корпусов.

Основные недостатки КУВТ УКНЦ:

1. Несбалансированность тепловых режимов (частые сбои и выход из строя в результате перегрева).

2. Неотработанность технологии некоторых комплектующих, в частности контроллера НГМД.

3. Использование несовершенной периферии НГМД-6022. Частые обращения к дискам, характерные для ПЭВМ DEC-архитектуры, требуют гораздо более надежных дисководов.

4. Хотя изначальная идея разделения функций центрального и периферийного процессоров несомненно здравая и соответствует тенденциям мирового развития, ее конкретная реализация в УКНЦ принесла больше осложнений, чем пользы. Во-первых, качественно иные функции ПП предполагают, что его структура и принципы работы принципиально отличаются от структуры и принципов ЦП. Поэтому совершенно неоправданно использование двух совершенно одинаковых процессоров. Во-вторых, мультипроцессорная система требует более серьезной проработки вопросов арбитража и распределения ресурсов, чем организация промежуточного буфера и регистрового доступа к страницам ОЗУ.

5. Своеобразие архитектуры и вызванная этим уникальность ПЭВМ привели к тому, что УКНЦ оказался практически «голым». Все программное обеспечение, включая поддержку работы с локальной сетью, необходимо разрабатывать заново. Использование профессиональных высокопроизводительных ЭВМ для разработки ПО УКНЦ также невозможно по той же самой причине. Более того, для создания прикладных пакетов, отличных от тривиальных, разработчики из языков программирования могут использовать только ассемблер.

6. Хотя микропроцессоры, используемые в ПЭВМ, могут работать с тактовой частотой 9 мГц и достигать при этом скорости выполнения регистровых операций 1.2 млн. оп/с, но реальное быстродействие компьютера не превышает 400 000 оп/с. Для повышения скорости работы программ разработчики рекомендуют записывать программы в ПЗУ и распараллеливать их работу с использованием одновременно двух процессоров, однако такой режим не является штатным и на практике нереален.

7. Невозможность независимого вывода на экран текстовой и графической информации крайне затрудняет создание прикладных программ учебного назначения.

#### КУВТ «Корвет»

КУВТ состоит из одного РМП типа ПК8020 и 12 РМУ ПК8010, объединен-

ных локальной сетью; сдвоенного дисководы и принтера западного производства.

В ПЭВМ использованы микропроцессор КР580ВМ80А с производительностью 625 тыс. оп/с и карта памяти с регистровым переключением страниц. Реализованы режимы прерываний, таймер, различные контроллеры.

ПЭВМ предоставляет программе пользователя 64К байта ОЗУ; видеоОЗУ, подразделяемое на графическое (ГЗУ) объемом 192К или 48К в зависимости от используемых микросхем памяти, и алфавитно-цифровое (АЦЗУ) объемом 1К×9 бит; ПЗУ объемом 24К байт, из которых 8К — заняты системой (драйверы), а 16К — интерпретатором Бейсика.

Контроллер НГМД 5.25" обслуживает дисководы с емкостью данных на неформатированной дискете до 1М байт (800К байт на форматированной).

Контроллер НМЛ допускает запись на кассетный магнитофон со скоростью 1200 бод.

Контроллер монитора обеспечивает подключение черно-белого монитора по выходу «Видео» и цветного — по выходу цифрового RGBI (соотношение цветов: В: G: R: I=1:2:4:8). Выход аналогового RGBI реализован, но не доступен пользователю.

Реализованы и доступны параллельный (ИРПР-М) и последовательный (ИРПС) и Стык-С2 (выведены оба). На плате имеются выходы для двухкоординатных джойстиков (не выведены на внешний разъем).

Имеется несколько графических режимов, до 512×256 точек 8 цветов (3 плана по 16К). Время закрасивания области экрана — 2 сек.

Существует текстовый режим, допускающий 80 символов в строке; формат знакоместа: 8×11 точек. Так как графическое ЗУ и алфавитно-цифровое ЗУ физически разнесены, совмещенные графического и текстового режимов не составляет проблем.

Локальная сеть (KERMIT) типа «Общая шина» реализует байториентированный протокол. Скорость передачи по сети 19.6К бод.

ПЭВМ содержит 126 корпусов микросхем, из которых БИС — 62 корпуса

(4 корпуса — программируемые логические матрицы (ПЛМ)).

Достоинства «Корвета» заключаются, во-первых, в его пользовательских качествах: полная совместимость с пакетами, разработанными в рамках операционных систем СР/М и Микро ДОС; наличие встроенного (в отличие от УКНЦ) интерпретатора Бейсика; щадящий режим работы с дисковым накопителем. Во-вторых, постоянно идущая модификация модели, что в условиях быстро меняющейся ситуации гарантирует жизнеспособность модели в течение некоторого времени.

Несомненным достоинством является совместимость ПЭВМ по форматам данных на внешних носителях со множеством отечественных и зарубежных ПЭВМ.

Можно отметить следующие недостатки:

1. Большое количество корпусов и недостаточная автоматизация производства.

2. Низкая надежность, в основном обусловленная некачественной сборкой в заводских условиях.

3. Некоторая недоработанность базового программного обеспечения (работа с локальной сетью), что вызывает определенные трудности при создании программ, работающих с локальной сетью.

4. Наличие импортной периферии (дисководы, принтер), закупаемой за валюту.

#### КУВТ «Агат»

Поставляемые в школы комплекты ПЭВМ «Агат» представляют собой 13 ПЭВМ (каждая с дисководом ИЗОТ) и матричное печатающее устройство (D100). Компьютеры не объединены локальной сетью.

Каждая ПЭВМ состоит из системного блока (с дисководом), клавиатуры и монитора.

В качестве центрального процессора используется, как правило, импортный микропроцессор MCS6502 с производительностью около 500 тыс. оп/с (утверждается, что система команд этого микропроцессора может быть эмулирована на БИС 588 серии, однако за счет значи-

тельного снижения скорости).

Адресное пространство, организованное в страницы по 16К, составляет 64К байта. Выбор страницы памяти осуществляется через регистры.

В модели, поставляемой в школы, объем ОЗУ — 64К байт, из которых видеоОЗУ занимает от 2К до 8К.

Системное ПЗУ составляет всего 2К байта и состоит из программы начального загрузчика и драйверов, однако существует режим эмулирования ПЗУ из части ОЗУ (до 32К байт). В ПЭВМ, поставляемой в школы, такое «псевдоПЗУ» составляет 12К байт и содержит программу «Монитор» и интерпретатор языка Бейсик.

ВЗУ представляет собой НГМД ЕС-5088 производства НРБ, обеспечивающее объем данных на 5.25" дискете 146К байт на неформатированной дискете 5.25" (109.4К на форматированной).

КНМЛ — бытовой, скорость обмена не менее 500 бод.

ПЭВМ «Агат» имеет контроллеры интерфейсов: параллельного (16 линий ввода/вывода) и последовательного (асинхронный, модулированный и немодулированный каналы).

ПЭВМ реализует следующие графические режимы: 64×64 точек — 16 цветов, 128×128 точек — 8 цветов или 256×256 черно-белых точек. Текстовые режимы: 32×32 точки 8 цветов или 64×32 черно-белых (знакоместо составляет 7×8 точек). Основными достоинствами ПЭВМ являются простая система команд, достаточно гибкая архитектура, наличие НГМД на каждом РМУ.

Основные недостатки:

1. Ненадежность работы как самой ПЭВМ (перегрев, сбой в работе контроллера ОЗУ), так и периферии, в особенности болгарских дисководов.

2. Низкое качество НГМД привело к тому, что программное обеспечение, изготовленное на одной ПЭВМ, не всегда идет на другой.

3. Используемая реализация языка Бейсик не совместима с реализациями на других ПЭВМ.

4. Из-за отсутствия локальной сети во время урока преподаватель вынужден постоянно бегать от одной машины к

другой для устранения сбоев и перезагрузки системы.

5. Цена компьютера столь велика, что обычная школа не в состоянии приобрести класс без помощи шефов.

6. Стоимость обслуживания настолько велика, что недоступна для большинства школ.

Напрашивается вывод о том, что «Агат» по своей сути не предназначен для задач образования. Его положительные стороны зачеркиваются недостатками, имеющими принципиальное значение для этой сферы деятельности.

## КУВТ УКНЦ-01

Название КУВТ УКНЦ-01, на наш взгляд, некорректно, так как этот КУВТ разработан не в г. Зеленограде, а на заводе «Экситон» (г. Павлов Посад). Архитектурно он не имеет почти ничего общего (за исключением системы команд) с КУВТ УКНЦ и программно с ним несовместим. КУВТ УКНЦ-01 построен на базе ПЭВМ БК-0011 («Электроника МС0513»), объединенных локальной сетью «Школок» производительностью 57.6К бод. Базовый класс предусматривает одно РМП и двенадцать РМУ.

ПЭВМ «Электроника МС0513» («Электроника БК-0011М») построена на основе микропроцессора с системой команд ряда ПЭВМ PDP — 11 фирмы DEC. Основой архитектуры БК-0011 является совместимость с программным обеспечением ПЭВМ БК-0010, что привело к определенной несовместимости с программным обеспечением DEC (в частности, в ПЭВМ БК-0011 сохранены несовместимые с DEC адреса внешних устройств).

По сравнению с БК-0010 в архитектуре БК-0011 расширено (до 48К байт) адресное пространство, доступное программам пользователя, увеличен до 128К байт объем ОЗУ, добавлены палитры в графике, применена более совершенная и производительная локальная вычислительная сеть, обеспечен доступ к НГМД со стандартными форматами данных на дискете, усовершенствован «защитый» в ПЗУ Бейсик. Несомненным достоинством КУВТ на



основе БК-0011 следует считать взаимозаменяемость входящих в него ПЭВМ (в отличие от КУВТ-86).

В БК-0011 используется операционная система типа RT-11SJ. Программное обеспечение пополнено некоторыми распространяемыми системными программами, разработанными в СССР (в частности, программой форматирования документации DOC, разработка ИНЭУМ).

Поставляемое программное обеспечение содержит языки высокого уровня Паскаль и Бейсик. Бейсик поставляется в двух вариантах, в значительной степени несовместимых друг с другом: «защитный» в ПЗУ (ROM-BASIC) и «дисковый». Само по себе это не является большим недостатком, так как «дисковый» Бейсик предназначен только для демонстрации совместимости с программными продуктами фирмы DEC и не предназначен для учебных целей (в частности, потому, что в нем используется код КОИ-7 и отсутствуют средства работы с графикой).

Не лишен существенных недостатков и ROM-BASIC. Низкая скорость заполнения текстового массива (составляющая в расчете на двухсимвольный элемент примерно 0.8 с/символ), по сути, исключает возможность использования в прикладных программах текстовых

переменных. Мал объем ОЗУ, выделяемый Бейсиком программе пользователя, — 14К байт, что затрудняет создание больших прикладных программ. Ни в одном из документов нет ссылки ни на ГОСТ 27787—88 «Язык программирования Бейсик», ни на международный стандарт ISO X6373—84. Не указано, как каждая реализация соотносится с действующим ГОСТом.

Таким образом, большую часть существующего программного обеспечения ПЭВМ БК-0011 нельзя считать патентно чистой, что ограничивает сферу возможного применения и поставок ПО.

Надежность серийных образцов зависит от множества факторов. Существуют, однако, недостатки конструкции, потенциально снижающие их надежность. В ПЭВМ БК-0011 применена небуферизованная общая шина, выведенная на внешний разъем. Это означает, что нештатные ситуации, возникающие при подключении внешних устройств (неисправность периферии, подключение к невыключенной ПЭВМ, случайные КЗ), будут, как правило, приводить к неисправностям, обнаружить и устранить которые будет невозможно силами не только пользователей, но и профессионалов.

В документации на ПЭВМ указано,

39

Сравнительные характеристики отечественных КУВТ

Параметры	КУВТ-86	УКНЦ	«Корвет»	«Агат»	УКНЦ-01
Быстродействие центрального процессора млн. оп/с	0.3	0.6 (1.2 при работе из ПЗУ на 2 процессорах)	0.625	1	0.7
Разрядность процессора	16	16	8	8	16
Объем ОЗУ, Кб	16	56(192)	64	64	48(64)
Емкость НГМД, Кб	2×200	2×400	2×800	146 на каждом РМУ	2×400
Объем ВидеоОЗУ, Кб	16	3×32	48—192	2—8	8
Количество точек на экране	512×256 256×256	640×288	512×256	64×64 128×128 256×256	512×256 256×256
Количество цветов	4 при 256× ×256	8 из 16	8	Соответственно 16, 8, 4/6	8
Скорость локальной сети, бит/с	9600	57 000	19 200	Локальная сеть отсутствует	57 600

К сожалению, данные, приведенные в таблице, недостаточно информативны. До сих пор отсутствуют критерии, позволяющие сравнивать ПЭВМ различных типов не по формальным параметрам, приведенным выше, а интегрально — по совокупности программно-аппаратных средств, максимально удовлетворяющих требованиям пользователя.

что в состав КУВТ входят дисководы типа НГМД-6022. Эти дисководы не рассчитаны на работу с форматами 400К байт на одной дискете, используемыми в КУВТ. Согласно другим материалам, КУВТ будут комплектоваться другими дисководами: НГМД-5309, которые удовлетворяют необходимым требованиям (что будет поставляться реально — не ясно).

Хотя ПЭВМ БК-0011 соответствует ГОСТ, определяющему общие требования к ПЭВМ (ГОСТ 27201—87), она не соответствует ГОСТу на расположение клавиш (ГОСТ 14289—88). С практической точки зрения это означает определенные неудобства для пользователя, привыкшего к стандартному расположению клавиш (УКНЦ, «Корвет», «Ямаха» и др.). Наиболее неудобно при работе расположение клавиши, временно переключающей большие/малые буквы. Большие неудобства вызывает отсутствие кнопки RESET и отсутствие автоповтора клавиш.

Таким образом, будучи исходно предназначенными для работы в системах управления экспериментом и контроля технологического оборудования, ПЭВМ с архитектурой, близкой к компьютерам фирмы DEC, плохо предназначены для использования в сфере образования. Хотя КУВТ УКНЦ-01, несомненно, является изделием более высокого класса, чем КУВТ-86, его «происхождение», а также ряд конструктивных недоработок не позволяют рассматривать этот КУВТ как перспективный (хотя его наверняка одобряют пользователи, привыкшие к программному обеспечению БК-0010).

## Зарубежные учебные ПЭВМ

За рубежом в образовании используются практически все виды персональных ЭВМ, исключение составляют лишь мощные графические станции и компьютеры-игрушки. Специально для целей обучения создан ряд ПЭВМ, такие, как «Archimedes A305» (фирм BBC и Acorn, Великобритания), Apple-IGS (фирма Apple, США). Кроме того, в учебном процессе все более широко применяются ПЭВМ типа IBM PC. Наиболее характер-

ным примером является класс Нанорезо II (Франция), состоящий из одной ПЭВМ типа IBM PC/XT и некоторого (от 10 до 20) количества ПЭВМ типа IBM PC. Все они объединены локальной сетью со скоростью передачи данных до 1М бод.

## УПЭВМ «Архимед»

Широко известная учебными компьютерами фирма Acorn Computers приступила во второй половине 1987 г. к выпуску семейства 32-разрядных ПЭВМ «Archimedes» («Архимед»), состоящему из двух серий: А300 — для использования в обучении и в быту; А400 — для профессиональной деятельности. В настоящее время выпускаются четыре модели: А305, А310, А410, А440. Фирма подчеркивает, что обучение — основная сфера применения новой ПЭВМ, и проводит соответствующую политику, включающую установление для школ льготных тарифов.

В компьютерах семейства используется микропроцессор, построенный по RISC технологии (reduced instruction set computer — компьютер с сокращенным набором команд), получивший название ARM (Acorn RISC Machine) с системой команд, во многом аналогичной системе команд микропроцессора 6502. Система однократных команд ARM состоит в основном из операций обмена между регистрами и ОЗУ. Быстрая обработка прерываний позволяет реализовать все функции ввода/вывода через микропроцессор, не используя дополнительного оборудования.

Первый вариант ARM был разработан в 1985 г. и имел производительность 3 MIPS (3 млн. простых операций в секунду). Современный микропроцессор имеет производительность 10 MIPS, содержит 27 тыс. транзисторов и занимает площадь 5×5 мм в 84- контактном пластмассовом корпусе. Разрабатывается микропроцессор с производительностью 17 MIPS. Следует отметить, что простота конструкции и высокая производительность ARM во многом достигнута благодаря перераспределению функций между микропроцессором и БИС управления памятью MEMC. На

MEMC возложены функции общей синхронизации, контроллера режима прямого доступа к памяти, формирования физического адреса и мультиплексирования микросхем ОЗУ и др.

Скорость «Архимеда» оценивается в 6 млн. оп/с.

Кроме центрального микропроцессора используются еще 3 сопроцессора: управления ОЗУ — MEMC, управления видеопамятью — VIDC и ввода/вывода — ИОС. MEMC адресует до 4М байт ОЗУ с тремя уровнями защиты памяти: «супервизор», «операционная система», «пользователь». MEMC выполняет также роль контроллера прямого доступа для управления буферами видеоданных, звука и курсора и осуществляет генерацию всех синхросигналов. В функции MEMC входит также преобразование 26-битового адреса ARM в 22-битовый адрес и арбитраж между ARM и VIDC. MEMC мультиплексирует до 32 БИС ОЗУ.

Объем ОЗУ в существующих моделях составляет от 0.5М байт до 4М байт в зависимости от модели (в А305 — 0.5М байт) ПЗУ — 0,5М байт, имеется также CMOS — ОЗУ объемом 240 байт и 16 байт для часов реального времени с питанием от литиевых батарей.

Системное ПЗУ размером 512К байт в настоящее время содержит: монитор операционной системы MOS, дисковую файловую систему ADFS, сетевую файловую систему ANFS, BBC Basic V, Basic Editor программу работы с «окнами» Desk Top Manager и несколько кодовых наборов символов.

Каждая из моделей комплектуется смонтированным в системный блок 3.5" дисководом для 1М байтных (неформатированных) дискет. Возможно подключение второго дисковода или диска типа «Винчестер» емкостью 20М байт (в ПЭВМ серии А300 для подключения «Винчестера» требуется дополнительный контроллер). Кроме того, имеется доступ к 20М байтному «Винчестеру» через локальную сеть.

ПЭВМ имеет контроллеры последовательного (RS—423) и параллельного (Centronics) интерфейса.

Видеопроцессор VIDC поддерживает 18 экранных режимов с максимальным числом цветов 256 из палитры — 4096

для мониторов с обычной скоростью сканирования и 3 режима для высокоскоростных мониторов. Всего поддерживается 22 экранных режима, в том числе 3 — с высоким разрешением (640×512 точек). Число цветов — до 256 из палитры 4096. Имеется система программных спрайтов. Скорость передачи на дисплей может изменяться программно переключением частоты передачи от 8 мГц до 24 мГц. Использование в микропроцессоре 32-разрядного циклического сдвигового регистра обеспечивает необходимый для графики высокого разрешения набор битовых операций.

ПЭВМ имеет канал стереозвуча на 8 голосов.

Клавиатура (расширенная IBM PC) содержит 103 клавиши, из которых 12 программируются пользователем. Монитор может подключаться как через аналоговый, так и через цифровой RGB входы. Имеется байтовый интерфейс принтера (Centronics), трехклавишная «мышь».

В машинах серии А300 пользователю предоставляется один 64-контактный системный расширитель магистрали (слот), допускающий 2 слота расширения, в А400 — 4(6) системных слотов.

«Архимед» поддерживает традиционную для Acorn локальную сеть ECONET с сетевым интерфейсом RS—423 (скорость передачи до 350К бод), представляющую пользователю доступ к периферии в режиме разделения времени. Следует отметить, что PS—423 распространен в основном в Великобритании.

Из дополнительно поставляемых устройств можно выделить плату сопроцессора с Intel—80186 для поддержки программного обеспечения фирмы IBM, а также сопроцессор арифметики с плавающей точкой (для серии А300 использование этой платы недопустимо).

«Архимед» рекламируется как ПЭВМ, имеющая вдвое большую производительность: чем VAX—780 и сопоставимая с последними 32-разрядными моделями, построенными на Intel—80386 и Motorola—68020.

Результаты тестов показывают, что «Архимед» превосходит по производительности значительно более дорогие 32-битные ПЭВМ (типа IBM PC/AT 386)

и чуть ли не на порядок быстрее сравнимых с ним по стоимости 16-битных.

«Архимед» был выпущен на рынок недавно, в конце 1987 г., но благодаря усилиям около 40 независимых фирм — производителей ПО уже к началу 1988 г. «Архимед» был достаточно хорошо оснащен программными средствами. С появлением машин у пользователей и с развитием инструментальных средств объем каталога в ближайшее время должен увеличиться на порядок. Уже сейчас имеются трансляторы языков ЛИСП, Пролог, Смолток и др.

Помимо чисто программных средств для «Архимеда» разработан ряд дополнительных периферийных программно-аппаратных модулей.

В списке обучающих программ и пакетов, занимающих почти половину каталога прикладного программного обеспечения, есть программы для всех возрастов — от обучения чтению пятилетних детей до сложных пакетов по физике, биологии, экономике для студентов. Практически все учебные программы созданы фирмами Longman, Scetlander и RESORCE. Следует отметить, что большинство обучающих программ — это ситуационные модели, ролевые игры, требующие от обучаемого принятия решений и демонстрирующие ближайшие и отдаленные результаты этих решений, что позволяет, не забывая голову зазубриванием несущественных частных, понять суть изучаемых проблем и их взаимосвязи.

## Apple-II

В настоящее время наиболее распространенными компьютерами, применяемыми в школах США (более 50 % парка машин), являются ПЭВМ семейства Apple-II. Несмотря на «возраст» (первая модель Apple-II была разработана более 11 лет назад), компьютеры этого класса сохраняют лидирующее положение в основном за счет относительно низкой цены и развитости программного обеспечения. В настоящее время семейство Apple-II состоит из 5 моделей: Apple-II, Apple-II+, Apple-IIe, Apple-IIc и Apple-IIGS. Появившаяся в конце 1986 г. модель Apple-IIGS

отличают такие качества, как открытость архитектуры, использование наряду с 8-разрядной 16-разрядной арифметики, использование технологии монтажа на поверхности. Относительно невысокая стоимость (цена базовой модели \$999), доступность богатейшего программного обеспечения Apple-II и другие достоинства позволяют компьютерам серии Apple до сих пор занимать одно из ведущих мест среди ПЭВМ для учебных и бытовых применений.

Apple-IIGS состоит из системного блока, выносной клавиатуры, «мыши», дисплея, утилит операционной системы ProDOS 8/16 и документации. В качестве центрального процессора используется микросхема 65C816 производства Western Design Center. Это 16-разрядная БИС, имеющая, кроме того, режим эмуляции системы команд 8-разрядного микропроцессора 6502, что позволяет использовать программное обеспечение Apple-II без каких-либо переделок. Тактовая частота работы 65C816 составляет 2.8 МГц, для обеспечения программной совместимости с Apple-II добавлен режим работы с частотой 1 МГц.

Адресное пространство микропроцессора составляет 16М байт (256 банков по 64К байта), однако системная шина Apple-II обеспечивает доступ лишь к 8М байтам. Стандартная поставка включает в себя 320К байт ОЗУ, из которых 128К байт «быстрого» (2.5 МГц) ОЗУ принадлежит ЦП, 128К байт «медленного» (1 МГц) ОЗУ — БИС Mega-II, осуществляющей функции ввода/вывода и управления, а 64К байт представляют собой специализированное ОЗУ звукового синтезатора. Это существенно превышает объем ОЗУ в Apple-IIe (128К байт). Объем встроенного ПЗУ также больше, чем в любой другой ПЭВМ серии Apple (128К байт), и включает в себя Applesoft Бейсик, инструментальные средства, функции обмена с монитором, клавиатурой, портами ввода/вывода, а также драйвер локальной сети Apple—Talk.

Контроллер дисководов обеспечивает работу с 800К байтными 3.5", а также (для совместимости с Apple-II) с 143К байтными 5.25" дискетами. Поддержки «Винчестера» в стандартной поставке не

предусмотрено, однако фирма Apple продает соответствующую аппаратуру в виде отдельного продукта.

Имеются 2 последовательных порта ввода/вывода, «мышь», выходы на монохромный и цветной мониторы (последний может подключаться как через RGB, так и через видеовход). В стандартную поставку входит монохромный монитор.

Клавиатура модифицирована по сравнению с Apple-II для обеспечения большего удобства пользователя и совместимости с клавиатурой Macintosh. «Защитные» в ПЗУ функции обслуживания клавиатуры включают в себя все системные входы из применяемого в Macintosh набора Control Panel, а также дают возможность переключать тактовую частоту 1/2.8 мГц. Возможны 7 вариантов кодировки клавиш для неанглоязычных стран.

Принципиально новой особенностью Apple-IIGS по сравнению с другими ПЭВМ серии Apple-II является наличие 7 слотов расширения шины Apple-IIe и слота расширения ОЗУ до 8М байт.

Встроенное расширение ОЗУ делает невозможным использование ряда разработанных для ранних моделей интерфейсных плат, например платы прямого доступа к памяти. Следует отметить, что с появлением ПЭВМ типа Apple-IIGS (а также Mac-SE, Mac-II, Commodor Amiga—2000 и др.) снова проявилась тенденция к открытой, «гибкой» архитектуре, позволяющей пользователю варьировать ресурсы в широких пределах в зависимости от своих нужд и не менять компьютер, как только он перестал удовлетворять возросшим потребностям. Дешивизна моделей с минимальной комплектацией, возможность наращивания ресурсов по мере роста сложности решаемых задач, доступность богатого программного обеспечения (особенно в области учебного использования ПЭВМ) вызывают интерес к возможностям применения этих ПЭВМ в различных областях.

Разъемы и кабели унифицированы, т. е. кабель для подключения к ПЭВМ принтера может быть использован для подключения к ней модема и наоборот, а также для подключения принтера непосредственно к модему. Порт для под-

ключения 3.5" и 5.25" дисководов (Smart Port) может быть использован для других блочных операций, например, для обмена с «Винчестером» или «файлсервером» (устройством, организующим доступ к ресурсам внешней памяти). «Винчестер», кроме того, может быть подключен через локальную сеть Apple-Talk, через SCSI — порт или специализированную интерфейсную плату.

Apple-IIGS поддерживает все текстовые и графические режимы ПЭВМ серии Apple: текстовые режимы на 40 и 80 символов в строке (24 строки, 6×8 точек в знакоместе), графические режимы с низким, высоким (280×192) и двойным (560×192) разрешением, комбинации текстовых и графических режимов. Добавлены два новых режима высокого разрешения: 320×200 точек с 16 цветами и 640×200 точек с 4 цветами.

Особенностью Apple-IIGS, отличающей ее от аналогичных ПЭВМ, является наличие звукового спецпроцессора фирмы Ensoniq с 64К байтами памяти, что позволяет записывать в цифровом виде и воспроизводить одновременно до 32 голосов, а также высококачественную речь и пение. Возможен цифровой синтез звуков с введением данных из файла в виде частотных гармоник, нотного текста или непосредственно с музыкального инструмента.

Технологическим успехом фирмы Apple следует считать высокую степень интеграции, достигнутую в Apple-IIGS. Большинство «обрамления», на которое в Apple-II уходило 70 корпусов МС малой интеграции, а в Apple-IIe — 20 корпусов, в Apple-IIGS сведено в одной микросхеме SMD — технологии (технология монтажа на поверхности). Это означает, что Apple-IIGS является более надежной ПЭВМ по сравнению со своими предшественниками, однако в случае поломки практически неремонтоспособной — микросхема может быть заменена только вместе с печатной платой. Технология SMD, требующая высокой роботизации производства, удешевляет конструкцию и одновременно страхует от несанкционированного копирования аппаратуры мелкими производителями.

## КУВТ на основе IBM PC совместимых моделей

Типовой КУВТ на основе IBM PC состоит из одной ПЭВМ IBM PC/XT (или PC/AT) с жестким диском емкостью не менее 20М байт (РМП) и некоторого количества (до 10) IBM PC/XT с НГМД 5 1/4". Все ПЭВМ объединены локальной сетью типа Arcnet или Ethernet.

Производительности центрального процессора (8088 с тактовой частотой 4.77 МГц) и емкостью оперативной памяти (не менее 512К байт) в настоящее время достаточно для выполнения большинства задач учебной информатики как для среднего, так и для высшего образования.

Проблемы для таких КУВТ в основном связаны с недостаточностью графических средств. Цветные мониторы CGA обладают низкой разрешающей способностью (640×200 точек в монохромном режиме и 320×200 точек при 4 цветах). В текстовом режиме знакоместо размером 8×8 и промежутки между текстовыми строками в одну строку развертки существенно затрудняют восприятие и оказывают вредное воздействие на зрение. Отсутствие аппаратной реализации средств динамической графики затрудняет программирование мультипликации. Существенным недостатком является также невозможность загрузки знакомест из ОЗУ. Это приводит к серьезным затруднениям при использовании национальных алфавитов.

К достоинствам таких КУВТ следует отнести, во-первых, наличие огромного количества программного обеспечения для самых различных уровней образо-

вания, во-вторых, быструю локальную сеть (до 10М бод), что, несомненно, облегчает учебный процесс с точки зрения преподавателя и позволяет наиболее полно использовать все ресурсы ПЭВМ.

В СССР также ведется разработка КУВТ такого типа: «Корвет 4», «Корвет 5»; «Электроника МС-1502» и т. п.

Следует отметить, что в настоящее время существенно подешевели ПЭВМ типа IBM PC/AT на микропроцессоре 80286. Кроме того, все шире распространяются мониторы VGA, которые по своим графическим характеристикам на порядок лучше CGA. Поэтому в ближайшее время следует ожидать появления КУВТ на основе этих ПЭВМ.

## КУВТ на основе ПЭВМ PS/2

Многие недостатки предыдущего КУВТ устранены в новых КУВТ, поставляемых сейчас в учебные заведения США. В них используются новейшие ПЭВМ фирмы IBM—PS/2 (модели 25, 30 и 50). Только в 1989 г. в учебные заведения США было поставлено около 0,5 млн. таких ПЭВМ. Школы и колледжи охотно приобретают эти ПЭВМ, так как, с одной стороны, за ними стоит фирма IBM, а с другой стороны, в учебные заведения эти ПЭВМ продаются с очень существенной скидкой.

Основные характеристики PS/2 модель 30: микропроцессор 8086, 8 МГц; ОЗУ 640К байт; НГМД 3.5", 720К байт, последовательный и параллельный интерфейсы; «мышь»; цветной монитор VGA (640×480 точек при 16 цветах).

Новейшая технология производства — поверхностный монтаж, крепле-

Параметры жесткого диска

Тип диска	Количество пластин	Размер диска	Дорожек на одной стороне	Секторов на дорожку	Всего секторов	Среднее время доступа	Емкость диска, МБ
CP-4024	1	3 1/2"	627	34(35)	42 636	29 Мсек	21.8
CP-3024	1	3 1/2"	636	33(34)	41 976	27 Мсек	21.5
CP-3100	4	3 1/2"	776	33(34)	204 864	25 Мсек	104.9
CP-3204F	4	3 1/2"	1366	38(39)	415 264	16 Мсек	212.6

В таблице приведены данные нескольких жестких дисков типа «Винчестер» (фирма CONNER), устанавливаемых в современных ПЭВМ.

В скобках указано физическое количество секторов на дорожку, а вне скобок — доступное количество секторов.

ния на фиксаторах позволили полностью автоматизировать процесс производства (вплоть до упаковки) и существенно повысили надежность ПЭВМ.

Модель 50 используется в качестве РМП и отличается от 30 модели следующими характеристиками: микропроцессор 80286, 10 Мгц; ОЗУ 1М байт; НГМД 3.5" 1.44М байт; жесткий диск 40М байт.

В качестве локальной сети используется сеть Token-Ring.

Параметры НГМД

Диаметр дискеты	Количе- ство дорожек	Секторов на дорожке	Емкость сектора	Емкость дискеты
5 1/4"	40	9	512 б	360 Кб
5 1/4"	80	15	512 б	1.2 Мб
5 1/4"	80	9	512 б	720 Кб
3 1/2"	80	9	512 б	720 Кб
3 1/2"	80	18	512 б	1.44 Мб

• Все типы дискет — двусторонние. Параметры приведены для одной стороны, а емкость дискеты — для двух сторон.

Сравнительные характеристики зарубежных КУВТ

Параметры	Архимед	Apple IIGS	IBM PC	PS/2—30
Быстродействие центрального процес- сора, млн. оп/с	10	1.5	2—6	4
Разрядность процессора	32	16/8	16	16
Объем ОЗУ, Мб	0.5—4	0.32	0,64	0.64
Емкость НГМД, Мб	1.0	0.8	1.2	1.44
Емкость жесткого диска, Мб	20	Нет в стан- дартном комплекте	20 —	40 —
Объем ВидеоОЗУ, Кб			До 256	До 256
Количество точек на экране	640×512	320×200 640×200	640×480	640×480
Количество цветов	256 из 4096	Соответственно 16 и 4	16	16
Скорость локальной сети, Мбит/с	0.35		Не менее 1.0	До 10.0

## Выводы и предложения

Даже анализ характеристик на «простейшем» уровне показывает огромный разрыв между зарубежными и отечественными ПЭВМ, поэтому сравнительный анализ может лишь подсказать характер развития советских ПЭВМ и дать ориентиры оценки вновь появляющихся моделей.

Сравнивая отечественные УПЭВМ с лучшими серийно выпускаемыми зарубежными УПЭВМ, можно констатировать, что наша страна отстала в этой области по крайней мере лет на 7—10 по надежности и по стоимости и лет на 5 по остальным пользовательским характеристикам. Это особенно тревожно, так как именно надежность и стоимость являются определяющими характеристиками для УПЭВМ.

Первая и основная проблема в создании отечественных моделей УПЭВМ состоит, по-видимому, в выборе конструк-

тива и элементной базы компьютера таким образом, чтобы достичь «золотой середины» между следующими противоречивыми требованиями.

1. Разработанная конструкция должна практически полностью основываться на достигнутом в массовом производстве уровне технологии и отечественных комплектующих, что позволит относительно быстро наладить серийный выпуск.

2. Тенденции развития учебной информатики в мире свидетельствуют о том, что технологический процесс здесь идет не менее бурно, чем в других областях вычислительной техники. Учебные ПЭВМ не «кормятся объедками со стола» профессиональных ПЭВМ и больших универсальных ЭВМ, а используют собственную элементную базу. Это должно стать образцом и для нашей промышленности. Ведь в ближайшем будущем в сферу образования должно поставляться ежегодно около 800 000 компьютеров.

По нашему мнению, на первом этапе (ближайшие 2—3 года) экономически целесообразным представляется вариант, когда УПЭВМ являются точным подмножеством профессиональных ПЭВМ (по аппаратным и программным средствам). Поскольку в СССР сейчас активно разворачивается производство IBM PC совместимых моделей, то и учебные ПЭВМ должны представлять их подмножество (или по крайней мере быть абсолютно совместимыми на уровне интерфейса с пользователем). Эти УПЭВМ в основном должны будут использоваться в курсе «Основы информатики и вычислительной техники».

В дальнейшем, при росте индустрии производства ПЭВМ, возможно появление альтернативных вариантов, связанных с использованием, например, RISC-процессоров (аналогично ПЭВМ «Архимед»).

При разработке УПЭВМ следует обратить особое внимание на следующие моменты.

1. ПЭВМ должна быть надежной. Тенденция в повышении надежности за рубежом — в росте уровня интеграции, уменьшении числа корпусов микросхем, создании специализированных БИС. Особую роль играет надежность периферии, в первую очередь дисководов.

2. Взаимосвязанность характеристик: как не имеет смысла комплектовать жестким диском 8-разрядную ПЭВМ, имеющую 32—64К байт ОЗУ, так и не имеет смысла создавать ПЭВМ с 0.128—1М байтным ОЗУ, допускающую в качестве ВЗУ лишь кассетный магнитофон. Аналогичные соотношения имеют место и для других характеристик.

3. Принципиальное значение имеют видеосредства. Без использования в конфигурации видеопроцессора высокой производительности с богатством функций не может быть и речи о современной учебной ПЭВМ. В расчете на будущее следует развивать и звуковые возможности УПЭВМ.

4. ПЭВМ, используемые для информатизации образования, на самом деле должны быть профессиональными ПЭВМ, предназначенными для использования в учебном процессе. Поэтому тут уже нельзя говорить об упрощенных характеристиках для снижения цены и т. п. Именно для этих целей и представляют наибольший интерес с точки зрения перспективы ПЭВМ на RISC процессорах.

5. Вновь разрабатываемое прикладное программное обеспечение, по-видимому, должно испытать качественный скачок. На начальном этапе круг задач компьютерного обучения (машинопись, обучение основам информатики) предполагает представление информации только в текстовой форме (также используемой как программа для зашитого в ПЗУ интерпретатора), что позволяет вести разработку ПО силами отдельных программистов, часто только пользуясь средствами резидентной вычислительной среды.

В дальнейшем предполагается активное использование стандартных трансляторов, графических пакетов, баз данных и т. д., т. е. ПО, действующего только в рамках операционной системы. Это предполагает все более углубляющееся разделение труда программистов и более высокие требования к их профессиональному уровню.

6. Следует отметить, что в нашей стране еще долгое время стоимость программного обеспечения будет пренебрежимо мала по сравнению со стоимостью аппаратуры, так как электроника стоит дорого, а тиражирование программного обеспечения осуществляется практически бесплатно. В связи с этим затраты на разработку программного обеспечения всегда будут экономически оправданы, если при этом будет достигнуто снижение стоимости аппаратуры.



## Графические средства системы Пролог-Д

Практически все современные учебные системы программирования предполагают использование компьютерной графики, она является мощным дидактическим средством. В традиционной компьютерной графике построение изображения на экране интерпретируется серией команд. Таким образом, всякое изображение ассоциируется с алгоритмом его построения и служит для иллюстрации алгоритмического подхода к работе ЭВМ. Однако эта концепция не укладывается в принципы декларативного программирования, принятые в системах логического программирования, к которым относится Пролог-Д. Если последовать концепции логического программирования, то необходимым становится описание элементов рисунка в виде совокупности графических объектов, соотношений и связей между ними. В этом случае описание последовательности действий художника-исполнителя становится излишним. (А может ли настоящий художник быть исполнителем?)

В системе Пролог-Д определен набор графических примитивов, отображающих графические объекты и построенных таким образом, что с точки зрения синтаксиса каждый из них может быть только *целью* и принимает значение «истина» (выполняется), если на экране появляется графическое изображение объекта. При записи в правиле нескольких графических примитивов и выполнении данного правила на экране появляется объединение графических образов в той последовательности, как они описаны в данном правиле.

В системе Пролог-Д БК-0010 предусмотрен один встроенный графический предикат ОТР ( $x, y, z, t, c$ ). Он выполняется, если на экране генерируется отрезок цвета  $c$ , соединяющий точки  $(x, y)$  и  $(z, t)$ . Аргументы  $x, y, z, t$  должны быть целыми или переменными, связанными с целыми, причем  $x, z$  должны быть в пределах от 0 до 256, а  $y, t$  — в пределах от 0 до 512.

В качестве примера приведем описание угла, вершина которого находится в точке  $(x, y)$ .

```
угол(x, y) <- ОТР(x, y, 10, 10, 1),
                ОТР(x, y, 50, 50, 1);
?угол(100, 100);
```

Сначала будет нарисован отрезок, соединяющий точки  $(100, 100)$  и  $(10, 10)$ , а затем отрезок, соединяющий точки  $(100, 100)$  и

$(50, 50)$ . Если бы пятым аргументом предикатов ОТР было число, равное нулю, то точки отрезков были бы невидимы.

Не обязательно, чтобы описание всей картинке было записано в одном предложении. Часть описания может быть выделена в виде отдельного предложения. Программу предыдущего примера можно модифицировать:

```
угол(x, y) <- ОТР(x, y, 10, 10, 1),
                продолжение(x, y);
продолжение(x, y) <- ОТР(x, y, 50, 50, 1);
?угол(100, 100);
```

Новая программа выполняет те же самые функции. 47

Система Пролог-Д допускает возможность использования переменных в графических примитивах. В качестве примера приведем описание вектора, выходящего из точки А с координатами  $(x, y)$  в точку В с координатами  $(s, t)$ :

```
вектор(A(x, y), B(s, t)) <- ОТР(x, y, s, t, 1);
```

Необходимо отметить особенность графических объектов, описываемых с помощью переменных. В процессе работы системы может оказаться, что какая-то переменная в описании графического примитива не определена. В этом случае графический примитив все равно будет выполнен, однако переменная принимает все допустимые для нее значения. Иными словами, на экране появится геометрическое место точек, задаваемое уравнением графического объекта. В качестве примера приведем вопрос

```
?ОТР(0, 0, x, 0, 1);
```

В результате ответа на этот вопрос на экране появится треугольник белого цвета. Причина этого в том, что величина абсциссы второй точки не определена, в этом случае абсцисса должна быть любым числом, в допустимых пределах. Как правило, область допустимых значений ограничена размерами экрана.

Языковые средства Пролога-Д обеспечивают возможность наращивать определения, естественным путем поддерживают структурность описания объекта. В качестве примера приведем описание построения домика. Домик можно определить как треугольник и квадрат, совмещенные одной стороной. Квад-

рат можно определить посредством четырех отрезков:

$$\begin{aligned} \text{квадр}(x, y, z, t) &< \text{ОТР}(x, y, x, t, 1), \\ &\text{ОТР}(x, y, z, y, 1), \text{ОТР}(x, t, z, t, 1), \\ &\text{ОТР}(z, y, z, t, 1); \end{aligned}$$

В этом выражении нет нарушения синтаксиса, однако длина этого предложения больше, чем допустимые 64 символа, и, чтобы оно было выполнено в рамках системы Пролог-Д БК-0010, необходимо разделить выражение, например, на две части:

$$\begin{aligned} \text{квадр}(x, y, z, t) &< \text{ОТР}(x, y, x, t, 1), \\ &\text{ОТР}(x, y, z, y, 1), \text{ч2}(x, y, z, t); \\ \text{ч2}(x, y, z, t) &< \text{ОТР}(x, t, z, t, 1), \\ &\text{ОТР}(z, y, z, t, 1); \end{aligned}$$

Аналогично определяется треугольник:

$$\begin{aligned} \text{треуг}(x, y, z, t) &< \text{ОТР}(x, y, t, y, 1), \\ \text{СЛ}(x, t, r), \text{ДЕ}(r, 2, f), \text{ОТР}(x, y, f, z, 1), \\ &\text{ОТР}(f, z, t, y, 1); \end{aligned}$$

(Предикат с именем ДЕ — предикат деления первого аргумента на второй, частное в третьем, а предикат с именем СЛ — сложение первого аргумента со вторым, результат в третьем аргументе). К сожалению, и это определение не входит в прокрустово ложе 64 символов. Вот окончательный вариант разделения:

$$\begin{aligned} \text{треуг}(x, y, z, t) &< \text{ОТР}(x, y, t, y, 1), \\ &\text{ДЕ2}(x, t, f), \text{угол}(x, y, z, t, f); \\ \text{угол}(x, y, z, t, f) &< \text{ОТР}(x, y, f, z, 1), \\ &\text{ОТР}(f, z, t, y, 1); \\ \text{ДЕ2}(x, t, f) &< \text{СЛ}(x, t, r), \text{ДЕ}(r, 2, f); \end{aligned}$$

Определение дома длиной 20 единиц, высотой этажа 10 единиц и высотой крыши 20 единиц имеет вид

$$\begin{aligned} \text{дом}(x, y) &< \text{крыша}(x, y, r, f), \text{этаж}(x, y, f); \\ \text{этаж}(x, y, f) &< \text{СЛ}(y, 20, z), \text{квадр}(x, y, f, z); \\ \text{крыша}(x, y, r, f) &< \text{СЛ}(r, 10, y), \\ &\text{СЛ}(x, 20, f), \text{треуг}(x, y, r, f); \end{aligned}$$

Вновь необходимо разделить это описание на две части: в одной части определяется крыша, а в другой этаж. (Это тот случай, когда дом приходится начинать строить с крыши.)

Использование рекурсивных определений дает возможность записать базу знаний более лаконично. Рекурсия может быть использована и для создания динамически изменяющегося графического объекта. Для этого на одном и том же месте последовательно фиксируется образ объекта так, что цвет его элементов попеременно меняется от цвета фона до цвета, определяемого в базе знаний.

Пример — описание летящей птицы, машу-

щей крыльями. В предикате *взмах* описан взмах вниз и затем взмах вверх. Первому положению соответствует горизонтальное положение отрезка белого цвета, затем этот отрезок гасится, становится цветом фона. Положению «вверх» соответствует угол, состоящий из двух отрезков белого цвета, которые затем гасятся. Каждое из понятий «вверх» и «вниз» описывается отдельно, при этом производятся арифметические операции, необходимые для вычисления координат начала и конца отрезков. Периодическое повторение взмахов вверх и вниз осуществляется с помощью рекурсивного обращения к одному и тому же предикату *птица*. (Внимание! В некоторых вариантах транслятора не предусмотрена обработка возникающей в данном примере эффекта концевой рекурсии типа петли. Следствием этого может быть аварийный останов после нескольких десятков взмахов.) Полное описание птицы, машущей крыльями, приведено ниже.

$$\begin{aligned} \text{птица}(x, y) &< \text{взмах}(x, y), \text{птица}(x, y); \\ \text{взмах}(x, y) &< \text{вниз}(x, y, 1), \text{вниз}(x, y, 0), \\ &\text{вверх}(x, y, 1), \text{вверх}(x, y, 0); \\ \text{вниз}(x, y, c) &< \text{сдв}(x, y, z, t, u, v), \\ &\text{ОТР}(z, y, u, y, 1); \\ \text{вверх}(x, y) &< \text{сдв}(x, y, z, t, u, v), \\ &\text{ОТР}(x, y, z, t, 1), \text{ОТР}(x, y, u, v, 1); \\ \text{сдв}(x, y, z, t, u, v) &< \text{сдп}(x, y, z, t), \\ &\text{сдл}(x, y, u, v); \\ \text{сдп}(x, y, z, t) &< \text{СЛ}(x, 5, z), \text{СЛ}(t, 5, y); \\ \text{сдл}(x, y, z, t) &< \text{СЛ}(z, 5, x), \text{СЛ}(t, 5, y); \end{aligned}$$

Декларативность языка позволяет создавать достаточно сложные картины, используя известные принципы декомпозиции графического объекта на части и последующее их описание.

Например, картина, содержащая дом и летящую птицу, есть пример и описания сложного синтетического объекта, и использования рекурсии для определения динамического изменения объекта.

$$\begin{aligned} \text{дом}(x, y) &< \text{крыша}(x, y, r, f), \text{этаж}(x, y, f); \\ \text{этаж}(x, y, f) &< \text{СЛ}(y, 20, z), \text{квадр}(x, y, f, z); \\ \text{крыша}(x, y, r, f) &< \text{СЛ}(r, 10, y), \text{СЛ}(x, 20, f), \\ &\text{треуг}(x, y, r, f); \\ \text{треуг}(x, y, z, t) &< \text{ОТР}(x, y, t, y, 1), \\ &\text{ДЕ2}(x, t, f), \text{угол}(x, y, z, t, f); \\ \text{угол}(x, y, z, t, f) &< \text{ОТР}(x, y, f, z, 1), \\ &\text{ОТР}(f, z, t, y, 1); \\ \text{ДЕ2}(x, t, f) &< \text{СЛ}(x, t, r), \text{ДЕ}(r, 2, f); \\ \text{квадр}(x, y, z, t) &< \text{ОТР}(x, y, x, t, 1), \\ &\text{ОТР}(x, y, z, y, 1), \text{ч2}(x, y, z, t); \\ \text{ч2}(x, y, z, t) &< \text{ОТР}(x, t, z, t, 1), \\ &\text{ОТР}(z, y, z, t, 1); \\ \text{птица}(x, y) &< \text{взмах}(x, y), \text{птица}(x, y); \\ \text{взмах}(x, y) &< \text{вниз}(x, y, 1), \text{вниз}(x, y, 0), \\ &\text{вверх}(x, y, 1), \text{вверх}(x, y, 0); \\ \text{вниз}(x, y, c) &< \text{сдв}(x, y, z, t, u, v), \\ &\text{ОТР}(z, y, u, y, 1); \\ \text{вверх}(x, y) &< \text{сдв}(x, y, z, t, u, v), \\ &\text{ОТР}(x, y, z, t, 1), \text{ОТР}(x, y, u, v, 1); \\ \text{сдв}(x, y, z, t, u, v) &< \text{сдп}(x, y, z, t), \end{aligned}$$

$\text{сдл}(x, y, z, t) \leftarrow \text{СЛ}(x, S, z), \text{СЛ}(t, S, y);$   
 $\text{сдл}(x, y, z, t) \leftarrow \text{СЛ}(z, S, x), \text{СЛ}(t, S, y);$

В ответ на вопрос:

?дом(70, 110), птица(120, 50);

на экране будет построен дом и нарисована птица, машущая крыльями. Обратите внимание на одно обстоятельство: предикат *птица* должен быть описан после предиката *дом*, так как в нем содержится концевая рекурсия типа петли.

В более мощной системе Пролог-Д MSX число встроенных графических предикатов больше. Полный их перечень приведен в таблице. Все аргументы — целые или переменные, конкретизированные целыми.

Функция	Действие
ФОН (Т, F)	Установка цвета текста и фона. Переменная конкретизируется целым, соответствующим установленному цвету
ТОЧКА (X, Y, C)	Генерация точки на экране с координатами X, Y и цветом C
ЛИНИЯ (X, Y, U, V, C)	Генерация линии из точки с координатами X, Y в точку с координатами U, V и цветом C
ОКРУЖНОСТЬ (X, Y, R, C)	Генерация окружности радиусом R с центром в X, Y

В качестве примера приведем описание двух синих окружностей с центром в точке с координатами 100 и 100, радиусами 10 и 50:

синие-окружности $\leftarrow$ ОКРУЖНОСТЬ(100,  
 100, 10, 5), ОКРУЖНОСТЬ(100, 0, 5);  
 ?синие-окружности;

Сначала будет нарисована окружность с радиусом 10, а затем окружность с радиусом 50.

Не обязательно, чтобы описание всей картинки было записано в одном предложении. Часть описания может быть выделена в виде отдельного предложения. Программу предыдущего примера можно модифицировать:

синие-окружности $\leftarrow$ ОКРУЖНОСТЬ(100,  
 100, 10, 5), продолжение;  
 продолжение $\leftarrow$ ОКРУЖНОСТЬ(100, 100, 50, 5);  
 ?синие-окружности;

Новая программа выполняет те же функции. Использование переменных в графических

примитивах аналогично использованию в системе Пролог-Д БК-0010.

Обратившись к опыту художников-примитивистов, нарисуем дерево в виде ствола — отрезка линии и кроны — окружности.

дерево(x, y, h, r)  $\leftarrow$  СЛОЖЕНИЕ(z, h, y),  
 ЛИНИЯ(x, y, x, z, 14), СЛОЖЕНИЕ(z, r, t),  
 ОКРУЖНОСТЬ(x, t, r, 3);

Если к этому определению в базе знаний задать вопрос

?дерево(50, 50, 40, 10);

то на экране появится одинокое дерево с координатами комеля 50, 50, высотой 40 и радиусом кроны 10.

Языковые средства Пролога-Д обеспечивают возможность наращивания определения, естественным путем поддерживают структурность описания объекта. Использование рекурсивных определений дает возможность записать базу знаний более лаконично.

Построим поляну с деревьями.

сад(1, x, y, h, r)  $\leftarrow$  дерево(x, y, h, r);  
 сад(m, x, y, h, r)  $\leftarrow$  смещение(x, y, z, t),  
 дерево(z, t, h, r), СЛОЖЕНИЕ(n, 1, m),  
 сад(n, z, t, h, r);  
 смещение(x, y, z, t)  $\leftarrow$  СЛОЖЕНИЕ(x, 10, z),  
 СЛОЖЕНИЕ(y, 10, t);  
 поляна $\leftarrow$  ОКРУЖНОСТЬ(v, 100, 50, 2),  
 сад(6, 100, 150, 50, 10),  
 сад(5, 85, 172, 50, 10);

Если к этой базе знаний задать вопрос:

?поляна;

то на экране появится искомое изображение.

Поляна изображена с помощью примитива ОКРУЖНОСТЬ, в описании которого не определена одна из координат центра. Это значит, что на экране должны быть нарисованы все возможные окружности, центры которых лежат на горизонтальной прямой, а радиус равен 50. В этом примере использована рекурсия для определения объекта «сад». В нашем понимании сад — несколько деревьев, расположенных на некотором расстоянии друг от друга. Чтобы нарисовать сад, необходимо воспользоваться рекурсивным определением. Оно содержит начальную, или базовую, ситуацию и описание некоторого рекурсивного соотношения. В примере:

базовой ситуацией будет та, когда в саду одно дерево;

рекурсивное определение получится тогда, когда будет несколько деревьев, смещенных относительно друг друга (смещение задано в виде специальной функции, именно она и определяет взаимное расположение деревьев).

### Упражнения.

1. Напишите на языке Пролог-Д базу знаний, описывающую прямоугольный треугольник.

2. Используя рекурсивное определение,

напишите базу знаний, описывающую многоэтажный дом.

3. Опишите на языке Пролог-Д построение улицы без учета и с учетом перспективы.

## В. ТИЩЕНКО

# Наладка простейшего компьютера

Если монтаж выполнен правильно и все детали исправны, то компьютер наладивать не придется, он заработает сразу. Однако выполнить сотни соединений, не сделав ни одной ошибки, практически невозможно. Всегда оказывается несколько пропущенных или неправильно соединенных проводников. Поэтому прежде всего необходимо проверить все соединения в монтаже общей шины и модулей. Помните, что ошибочное соединение может полностью вывести из строя модуль или привести в негодность его основные детали! Проверку выполняют при помощи омметра или простого пробника, состоящего из последовательно соединенных лампочки на 2,5 В и элемента питания 373.

Проверку начинают с разъемов общей шины, подключив один конец пробника (омметра) к гнезду А1 любого разъема, а второй конец поочередно подключая к гнездам А1 остальных разъемов; по степени свечения лампочки (показаниям омметра) судят о правильности и надежности соединений. Аналогично проверяют параллельное соединение друг с другом гнезд В1, А2, В2 и т. д.

После проверки монтажа общей шины тщательно проверьте монтаж всех модулей, отмечая цветным карандашом на принципиальной электрической схеме каждое проверенное соединение.

Затем проверяется блок питания. Включив его в сеть, проверьте вольтметром все три выдаваемых напряжения (+5 В, —5 В, +12 В). Если напряжения соответствуют норме, то штыревой разъем блока питания включите в гнездовой разъем общей шины и вольтметром проверьте наличие и величину питающих напряжений на гнездах шоры питания (А1, В1, А2 и В2) всех разъемов общей шины. Если отклонений от нормы нет, в один из разъемов общей шины включите разъем модуля ввода, вывода и управления, а на плате этого модуля штыревые разъемы Х2 и Х3 включите в гнездовые. При этом должны засветиться

те из светодиодов VD1—VD24, тумблеры которых (SA1—SA24) находятся в верхнем (по схеме) положении. Установите все тумблеры SA1—SA24 в нижнее положение, а затем, переключая каждый тумблер из нижнего положения в верхнее, а затем снова в нижнее, проверьте свечение и погасание каждого светодиода.

Одновременно вольтметром проверяется появление напряжения +5 В на соответствующих гнездах шины адреса и данных тех разъемов общей шины, в которые предполагается включать модули памяти и микропроцессора. При установке тумблера SA1 в верхнее положение должен загореться светодиод VD1 и появиться напряжение +5 В на гнездах А22 (ША15 шины адреса); при переключении тумблера SA2 должен загореться светодиод VD2 и появиться напряжение на гнездах В22 (ША14) и т. д.

Если какой-либо светодиод не загорается, нужно проверить его исправность, подключив параллельно ему вольтметр. Если напряжение превышает 2 В, значит, светодиод неисправен или неправильно подключен (анод ошибочно принят за катод). В случае отсутствия напряжения на светодиоде проверьте исправность транзистора, к которому он подключен, а также замерьте напряжение на базе транзистора. Оно должно составлять +0,2 В, когда соответствующий тумблер установлен в верхнее положение.

Убедившись в исправной работе узлов светодиодной индикации и тумблерного ввода, следует перейти к проверке узла управления. Замерьте напряжение на выводе А5 (СБР) шины управления разъема модуля микропроцессора. Оно должно находиться в пределах от +2,5 В до +3,5 В. Нажмите кнопку SA25 (СБР). Напряжение должно упасть до нуля. Многократным нажатием кнопки SA25 проверьте четкость и надежность сигнала СБР, осуществляющего сброс счетчика адреса микропроцессора в исходное состояние.

Следующий этап проверки — определение

правильности очень ответственного процесса переключения (захвата) шин адреса и данных для использования их микропроцессором или памятью (при ее программировании). В целях проверки можно воспользоваться тумблером SA1 в качестве источника управляющего сигнала, выдающего (взамен отсутствующего микропроцессора) высокий (1) или низкий (0) логический уровень. Для этого соедините проводником вывод A22 (ША15) с выводом A7 (ПЗХ) на разъеме модуля микропроцессора общей шины. При установке тумблера SA1 в верхнее положение светодиод VD25 (ПЗХ) должен загореться, а при установке в нижнее положение — погаснуть.

Затем проверьте цепи захвата шин: подключив вольтметр к гнездам B1 (OB) и B7 (ЗХ) разъема микропроцессора общей шины, переводите из одного положения в другое переключатель SA26 (ЗХ). При этом вольтметр должен показывать четкое изменение напряжения от нуля до +5 В.

Теперь можно перейти к проверке цепей, участвующих в программировании памяти. Подключите вольтметр к гнездам B1 (OB) и A9 (BK) разъема памяти общей шины. Он должен показывать напряжение около 3 В. Нажмите кнопку SA29 (ЧТ); напряжение должно упасть до нуля. Такой же результат должен наблюдаться и при нажатии кнопки SA30 (ЗП). Переключите щуп вольтметра из гнезда A9 (BK) в гнездо B9 (Ч/З); напряжение должно быть около 5 В. Нажмите кнопку SA30 (ЗП); напряжение должно упасть до нуля. При нажатии кнопки SA29 (ЧТ) показания вольтметра изменяться не должны.

Затем проверяется наличие тактовых сигналов Ф1 и Ф2 на выводах B4 (Ф1) и A4 (Ф2) разъема микропроцессора общей шины. Эту проверку лучше всего выполнить с помощью осциллографа, позволяющего определить частоту, длительность и форму сигналов. Если осциллограф отсутствует, то приблизительную оценку сигналов Ф1 и Ф2 можно произвести, подключая сигналы Ф1 и Ф2 к видеосулителю телевизора через емкость 10 мкф или к антенному гнезду коротковолнового приемника через емкость 50 пф. По характеру темных полос на экране телевизора или по звуку в радиоприемнике судят о наличии тактовых сигналов Ф1 и Ф2.

Наступила очередь модуля памяти. Выключите блок питания, вставьте разъем модуля памяти в разъем общей шины и снова включите питание.

Все тумблеры SA1—SA24 установите в нижнее положение. Светодиоды VD1—VD24 должны погаснуть. Нажмите и отпустите

кнопку SA30 (ЗП), записывая этим в начальную (нулевую) ячейку памяти, имеющую адрес 0000000000000000, двоичный код 00000000. Тумблер SA16 переводите в верхнее положение, устанавливая этим следующий адрес ячейки памяти (000000000000001). С помощью тумблеров SA17—SA24 наберите двоичный код 01010101 и снова кратковременно нажмите кнопку SA30 (ЗП). Тумблерами SA15 и SA16 установите адрес 0000000000000010, а с помощью тумблеров SA17—SA24 — двоичный код 10101010 и опять кратковременно нажмите SA30 (ЗП). Затем так же занесите код 11111111 по адресу 0000000000000011.

В результате в четыре первых ячейки памяти оказываются записанными такие двоичные коды:

```
00000000
01010101
10101010
11111111
```

51

Теперь нужно проверить правильность выполненной записи. Для этого извлеките штыревой разъем X3 из гнездового, отключая этим тумблеры SA17—SA24 от шин. При помощи тумблеров SA1—SA16 наберите адрес нулевой ячейки памяти (0000000000000000) и нажмите кнопку SA29 (ЧТ). Светодиоды VD17—VD24 должны отобразить двоичный код (00000000) нулевой ячейки памяти (остаться темными). Изменяя адрес при помощи тумблеров SA1—SA16 и нажимая кнопку SA29 (ЧТ), проверьте, какие коды записались в первую, вторую и третью ячейки памяти.

Проверив модуль памяти, переходите к самому ответственному этапу налаживания — проверке модуля микропроцессора. Для этого выключите питание, разъедините разъемы X2 и X3, установленные на модуле управления, снимите временные проводники с гнезд разъемов и включите модуль микропроцессора в разъем общей шины. После этого можно включить питание. Тумблер SA27 установите в верхнее положение (Автом.), а тумблер SA26 (ЗХ) — в такое положение, при котором светодиод VD25 (ПЗХ) оказывается погашенным.

Частое мигание светодиодов старших разрядов шины адреса (VD1—VD3) свидетельствует об исправной работе микропроцессора, который беспорядочно перебирает адреса различных ячеек памяти. Если переключатель SA27 перевести в нижнее положение (Пошаг.), то мигание светодиодов прекратится, микропроцессор приостановит свою работу и выдаст случайный адрес (VD1—VD16) и двоичный код (VD17—VD24)

ячейки памяти, соответствующей адресу останова.

Испытав работу микропроцессора, переходят к программированию памяти. Тумблер SA26 (3X) нужно перевести в положение, при котором загорается светодиод VD25 (ПЗХ).

Убедившись в устойчивом свечении светодиода VD25 (ПЗХ), включите штыревые разъемы X2 и X3 в гнездовые и в описанном выше порядке (см. проверку модуля памяти) в первые шесть ячеек памяти запишите двоичные коды диагностической программы, приведенной в таблице.

Адрес	Данные
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0
0 0 0	
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 0 1 0 1 0 1 0
0 0 1	
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 1 0 1 0 1 0 1
0 1 0	
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 1 0 0 0 0 1 1
0 1 1	
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0
1 0 0	
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0
1 0 1	

Затем разъедините разъем X3, вынув штыревую часть из гнездовой. Тумблерами SA1—SA16 последовательно установите адреса, указанные в левой части таблицы, и, нажимая кнопку SA29 (ЧТ), убедитесь в том, что в ячейках памяти верно записаны двоичные коды данных.

Разъедините разъем X2.

Тумблер SA27 установите в положение «Пошаг.», а тумблер SA26 (3X) — в положение, при котором гаснет светодиод VD25 (ПЗХ). Нажмите кнопку SA25 (СБР). При этом должны погаснуть все светодиоды VD1—VD24. Последовательно нажимайте SA28 (ШАГ). При каждом нажатии свето-

диоды адреса (VD1—VD16) и данных (VD17—VD24) должны отображать очередную строку таблицы, а после шести нажатий снова вернуться к отображению первой строки. Диагностическая программа составлена так, что после выполнения последней строки микропроцессор автоматически возвращается к выполнению первой строки и бесконечно совершает этот цикл.

Убедившись в таком циклическом выполнении программы, переключатель SA27 переведите в положение «Автом.». После этого начинается автоматическое выполнение микропроцессором программы в пределах первых шести адресов; мигание светодиодов старших разрядов адреса незаметно, все светодиоды светятся равномерным, но неполным свечением.

Теперь компьютер готов к работе. Он позволяет:

записывать данные в любую из 2048 ячеек памяти;

читать данные, содержащиеся в этих ячейках;

выполнять в пошаговом или автоматическом режиме программы, составленные на машинном языке и записанные в память;

производить наглядный обмен данными с внешними устройствами (реле, переключателями, звуковыми и световыми устройствами);

постепенно усложнять и совершенствовать его структуры путем подключения дополнительных модулей.

Основным назначением простейшего компьютера является наглядная демонстрация принципа работы его модулей и освоение азов программирования путем составления и прогона несложных программ на машинном языке.

Вопросы практической работы на простейшем компьютере, его программирование, а также рекомендации по дальнейшему совершенствованию будут изложены в следующих статьях.

## Графический планшет

Компьютер в школе уже не редкость. Но чаще всего десяток машин с колоссальной вычислительной способностью большую часть времени простаивают, используют их лишь для изучения основ информатики. Использование компьютеров на уроках по другим предметам сдерживается в основном неумением программировать. Предлагаемое устройство облегчит программирование, автоматизировав создание изображений. Описываемый вариант предназначен для подключения к ПЭВМ «Агат», но возможно использование и с другими типами ЭВМ.

Механическая часть представляет собой устройство типа «планшет», преобразующее координаты положения рисуемого элемента на рабочем поле в линейно изменяющиеся сопротивления резисторов R1 и R2. В рисуемом элементе имеется кнопка, срабатывающая от нажатия при рисовании. Подключается устройство к разъему «Пульт» ПЭВМ «Агат».

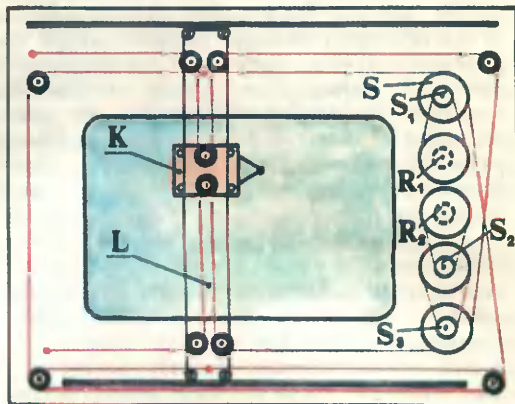
Обслуживающая программа обеспечивает: точное повторение на экране рисунка, выполняемого на рабочем поле прибора, любым из 8 цветов на фоне любого цвета; четыре вида графики: высокого ( $256 \times 256$ ), среднего ( $128 \times 128$ ), низкого ( $64 \times 64$ ) и инфранизкого ( $32 \times 32$ ) разрешения; полную имитацию рисования. При наложении устройства на экран дисплея оно действует подобно «световому перу»;

смену цвета, стирание изображения с одновременной сменой фона (осуществляется нажатием соответствующих клавиш в паузах рисования);

запись изображения в память в компактном виде (три ячейки на вектор) с последующей записью на магнитный носитель в виде В-файла;

воспроизведение любой части изображения с возможностью перемещения его в любую часть экрана. Создается кодовая подпрограмма, которая позволит закрашивать замкнутую область любой формы.

Основанием механической части устройства является прямоугольная пластина из прозрачного оргстекла толщиной 5 мм. Кинематическая схема устройства приведена на рисунке. Вертикальная рейка L с подвижной кареткой K может совершать параллельные перемещения. Это достигается применением системы шкивов и канатиков. Бесконечный канатик, обходя



четыре шкива, расположенные в углах пластины, прикрепляется к концам рейки в двух точках. Нижние по схеме шкивы имеют двойную канавку, и канатик обходит их дважды. Верхний правый шкив двойной. Он передает вращение на другой шкив, насаженный на ось резистора R1, сопротивление которого пропорционально координате X положения каретки на рабочем поле. С помощью другой системы шкивов вертикальное перемещение каретки передается резистору R2. Концы рейки и каретка снабжены роликами, представляющими собой миниатюрные шариковые подшипники. Все детали, в том числе и шкивы, изготавливаются из оргстекла. Канавки вытачиваются на токарном станке или с помощью бытовой электродрели. В центры шкивов запрессовываются при нагревании шариковые подшипники. Диаметры всех шкивов желательно брать большими, это уменьшает сопротивление. Диаметры больших шкивов все одинаковы (у автора 9 см). Диаметр шкива S3 в 2 раза меньше диаметра S. Диаметры S1 и S2 подбираются в зависимости от размеров рабочего поля. Если предполагается работать методом наложения устройства на экран дисплея, то размеры поля должны совпадать с размерами изображения на экране. В любом случае диаметр их определяется следующим образом: сначала определяется активная длина шкива, насаженного на ось резистора. Для этого резистор подключается вместо одного из резисторов пульта. Вводят и запускают программу 1 и, вращая шкив с помощью канатика, определяют изменение его длины при измене-

Программа 1

```
10 HOME
20 HTAB 10: VTAB 16: PRINT PDL(0); " "
30 HTAB 20: VTAB 16: PRINT PDL(1); " "
40 GOTO 20
```

нии одного из чисел на экране от 1 до 255. Отношение диаметров шкивов S:S1 (S:S2) должно быть равно отношению соответствующего размера рабочего поля (по горизонтали или вертикали) и полученной активной длины. Все детали механизмов передачи закрываются корпусом из декоративной фанеры или пластика, оставляя окно для рабочего поля. Рисующий элемент вставляют в петлю из упругой и тонкой проволоки, прикрепленную к каретке. Изготовлен он из корпуса авторучки или фломастера подходящих размеров. В корпусе укрепляется рисующий узел от чертежного набора с грифелем. Узел должен свободно перемещаться в отверстии на 1—2 мм при нажатии и возвращаться обратно при отпускании. Внутри корпуса устанавливается микровыключатель, срабатывающий при нажатии на рисующий элемент при проведении линий. Микровыключатель и резисторы (R1 и R2) подключаются вместо кнопки и резисторов пульта. Резисторы должны быть обязательно группы А и иметь сопротивление 110 кОм. Лучше работают резисторы старых типов с большим радиусом дуги.

Программа 2 является настроечной, но может быть использована и на практике для занесения рисунков в ОЗУ экрана. В этом случае изображение может быть записано в виде В-файла данной страницы. Полная программа с применением кодовых подпрограмм и других сервисных возможностей имеется у автора.

Устройство может быть использовано с другими ПЭВМ, имеющими средства графики, но для этого нужно добавить электронную схему АЦП. Вариант такой схемы дол-

Программа 2

```
10 HOME: HTAB 6: VTAB 10:
PRINT "ВВЕДИТЕ РЕЖИМ РАБОТЫ
(ГВР - 1, ГСР - 2, ГНР - 4,
КУРСОР - 8)": INPUT R
15 PRINT "НАЧАЛЬНЫЙ ЦВЕТ ЗАПИСИ (0-7)"
18 INPUT C: COLOR=C: RIBBON=C
20 IF R=1 THEN HGR=1: COLOR=15: GOTO 90
30 IF R=2 THEN MGR=1: GOTO 60
40 IF R=4 THEN GR=4: GOTO 60
50 IF R=8 THEN 180
55 GOTO 10
60 COLOR=C
90 X1=PDL(0)/R: Y1=PDL(1)/R
100 X=PDL(0)/R: Y=PDL(1)/R
120 PLOT X1,Y1 TO X,Y: X1=X: Y1=Y
140 IF PEEK(BC061)<200 THEN 100
145 X1=PDL(0)/R: Y1=PDL(1)/R
150 S=PEEK(BC000): IF S<127 THEN 140
160 S=S-127
170 IF S>47 AND S<58 THEN COLOR=S-49:
C=S-49
172 IF S=56 THEN COLOR=15
173 IF S=71 OR S=103 THEN GOSUB 300
175 POKE BC012,00: GOTO 90
180 TEXT=4: HOME: INVERSE: RIBBON=C
190 X=PDL(0)/R: Y=PDL(1)/R
200 HTAB H: VTAB Y: PRINT " "
210 IF PEEK(BC061)<200 THEN 190
220 S=PEEK(BC000): IF S<127 THEN 210
225 S=S-127
230 IF S>48 AND S<57 THEN RIBBON=S-49:
GOTO 210
235 IF S=71 OR S=103 THEN HOME
240 POKE BC012,00: GOTO 210
300 FOR I=0 TO 255/R
310 PLOT I,I TO 255,I
320 NEXT I: RETURN
```

Назначение клавиш управления программой  
0-7 - задание цвета;  
F - стирание изображения, установка  
фона текущего цвета

жен быть опубликован в журнале «Микропроцессорные средства и системы» (1989, № 5).

При тщательном изготовлении механической части устройства оно позволит приблизить ввод графики к естественному рисованию и даже вводить рукописный текст.

В. АРТАМОНОВ

## Музыка КУВТ «Корвет»

В языке Бейсик КУВТ «Корвет» нет специальных средств, позволяющих программировать мелодии. Однако с помощью оператора ВЕЕР этот пробел можно устранить.

После выполнения процедур начальной инициализации в системной области памяти ПК8020 появляются четыре 8-разрядных ячейки, работая с которыми можно при по-

мощи встроенного динамика генерировать широкий по высоте и длительности диапазон звуков. Это ячейки со следующими адресами:

F715, F716 — высота сигнала BELL;

F717, F718 — длительность сигнала BELL.

Первоначально в первые две ячейки заносится число 25A0, а во вторые — число



0800, что заставляет динамик генерировать звук частотой 800 Гц и длительностью 0,25 с.

Указанные выше факты позволили установить следующие соотношения: чтобы динамик генерировал звук с частотой F (Гц) и длительностью D (с), необходимо в ячейки с адресами F715, F716 записать число  $A=9632 \times 800/F$ , а в ячейки с адресами F717, F718 — число  $B=2048 \times 4 \times D$ . При этом по адресам F715 и F717 записываются младшие разряды указанных чисел.

Полученные формулы носят эмпирический характер, однако прослушивание профессиональными музыкантами мелодий, исполняемых компьютером, подтвердило их правильность.

Хотя соотношения для частоты и длительности уже позволяют программировать музыку, необходимый при этом перевод нот в форму многоразрядных чисел оказывается делом затруднительным. Поэтому возникла идея написать прикладную программу, имитирующую оператор PLAY расширенного Бейсика IBM PC, который предусматривает применение специального музыкального языка для программирования мелодий.

В силу особенностей ПК8020 удалось реализовать не все из 19 подкоманд оператора PLAY, однако наиболее важные были запрограммированы. Кроме того, исполнение мелодий в режиме реального времени, как это делается в операторе PLAY, вызвало затруднения из-за относительно медленной работы интерпретатора языка Бейсик ПК8020. Поэтому были написаны две программы: PLAY и MELOD.

Программа PLAY — своего рода компилятор, который каждую ноту представляет в виде четырех целых десятичных чисел и вместе с названием мелодии записывает на ГМД в файл с именем, выбранным пользователем. Она «понимает» следующие команды:

нота — звучание указанной (C, D, E, F, G, A, B) ноты в нужной октаве с диэзом (знак +), бемолем (знак —) либо чистой (знак пробела);

O (октава) — задание номера от 0 до 7;

N (номер) — звучание ноты с номером от 01 до 17;

L (длительность) — задание длительности всех последующих нот от целой 01 до шестидесятичетвертой 64. Как вариант этот параметр может следовать за конкретной нотой, указывая ее длительность;

P (длительность) — пауза. Параметр указывается, как и в команде L;

T (число ударов) — задание темпа произведения от 30 до 260;

MN — исполнение нон легато (обычное);

ML — исполнение легато;

MS — исполнение стаккато.

Внешне эти команды аналогичны подкомандам оператора PLAY. Однако есть ряд отличий.

Команда «.» в явном виде отсутствует. Чтобы увеличить длительность звучания ноты в число раз, кратное 1,5, необходимо показатель кратности поставить в 5—6-ю позиции команды «нота» (например, D+0801); в 6—7-ю позиции подкоманды «N» (например, N050801); в 4—5-ю позиции подкоманд «L», «P» (например, L0802 или P1601).

Ноты B— и A+, A— и G+, G— и F+, E— и D+, D— и C+ в подкоманде «N» имеют разные номера, хотя при исполнении мелодии звучат одинаково.

Мелодии могут записываться в восьми октавах, из которых две лежат ниже первой (номера 0 и 1), а пять — выше.

Программа MELOD озвучивает с помощью оператора ВЕЕР откомпилированные программой PLAY мелодии. Для этого указанный пользователем файл с мелодией считывается в оперативную память и перед каждым использованием оператора ВЕЕР в ячейки F715 — F718 заносятся целые числа, характеризующие очередную ноту.

Некоторую сложность представляет реализация пауз при исполнении мелодии. Специальные средства для этого отсутствуют; однако если в ячейки с адресами F715 и F716 записать число 2, то будут генерироваться импульсы с частотой 3,9 МГц, т. е. не воспроизводимые динамиком и не слышимые ухом. Это обстоятельство и было использовано.

Для иллюстрации приведем команды для исполнения на ПК8020 песни «Тонкая рябина».

```
T76.02.E 02.G 04.B 02.03.C 04.02.B 04.
A ,F01,D+02,F+04.03.C 02.02.B 04.A 04,
G ,P01.E 04.G 04.B 04.03.E 02.D 04.
D 04,C ,P01.D 02.C 04.02,B 02.A 04,
B 04.E ,F01.E 04.G 04.B 04.03.E 02.
D 04,D 04,C ,P01,D 02,C 04.02,B 02,
A 04,B 04.03.E 01
```

Ниже приведены также полные тексты программ PLAY и MELOD.

```
5 REM *****"PLAY"*****
10 REM ПРЕПРОЦЕССОР МЕЛОДИИ
15 REM АВТОР АРТАМОНОВ В.С.
17 REM *****
20 CLS:PCLS:LOCATE 20,2,1:PRINT
  "ПРОЦЕССОР МЕЛОДИИ":CLEAR 2000
22 PRINT "ВВЕДИТЕ УНИКАЛЬНОЕ ИМЯ
  ФАЙ
ЛА ДЛЯ ХРАНЕНИЯ МЕЛОДИИ,"
24 PRINT "ПОСЛЕ ПОЯВЛЕНИЯ НА ЭКРАНЕ
  ОК
  НАБЕРИТЕ:GOTO 27<BK>:"
25 INPUT NM$:OPEN "O",#1,"FILE.BAS"
  :NM
```

```

$="52 OPEN "+CHR$(34)+"O"+CHR$(34)
      +",#1,"+C
HR$(34)+NM$(34)+CHR$(34)
      :PRINT #1,NM$:C
LOSE 1
26 MERGE "FILE.BAS"
27 PRINT "НАЗВАНИЕ МЕЛОДИИ:":INPUT
      M$:
PRINT "АВТОР:":INPUT A$
29 DIM B(17)
30 B(1)=65.405:B(2)=69.41:B(3)=69.41
      :B(
4)=73.415:B(5)=77.91
35 B(6)=77.91:B(7)=82.405:B(8)=87.305
      :B(
9)=92.6525:B(10)=92.6525
40 B(11)=98!:B(12)=104!:B(13)=104!
      :B(
14)=110!:B(15)=116.735
      :B(16)=11
6.735:B(17)=123.47
45 DEF FNH(A$,S$)=
      (IN
STR(S$,LEFT$(A$,1))-1)*16+
      INSTR(S$,
RIGHT$(A$,1))-1
50 S$="0123456789ABCDEF"
52 OPEN "O",#1,"AAAAAA"
54 PRINT #1,M$,A$
55 O=2:T=2048*24:REM ПО УМОЛЧАНИЮ
      2-Я
      ОКТАВА,ТЕМП-LARGHISSIMO
57 KF=1:REM ПО УМОЛЧАНИЮ
      ДЛИ
ТЕЛЬНОСТЬ РАВНА 1
58 Z=1:REM ПО УМОЛЧАНИЮ
      ПРИ
ЕМ- LEGATO.
59 GOSUB 1000:NT=0
60 LOCATE 1,15,0:PRINT SPC(60)
      :LO
CATE 1,15,0:INPUT "КОМАНДА";
      AG$:IF AG
$="***" OR AG$=""
      THEN 999
65 G$=LEFT$(AG$,1)
70 IF G$="O" THEN GOSUB 100:GOTO 60
72 IF G$="T" THEN GOSUB 200:GOTO 60
74 IF G$="N" THEN GOSUB 300:GOTO 60
76 IF G$="L" THEN GOSUB 400:GOTO 60
78 IF G$="P" THEN GOSUB 600:GOTO 60
80 IF G$="M" THEN GOSUB 500:GOTO 60
85 GOSUB 2000:GOTO 60
100 REM П/П ВЫЧИСЛЕНИЯ ОКТАВЫ.
110 O=LEN(AG$)-1:O$=RIGHT$(AG$,O)
      :O
=VAL(O$)
120 IF O<0 THEN O=0
130 IF O>7 THEN O=7
140 RETURN
200 REM П/П ВЫЧИСЛЕНИЯ ТАКТА.
210 T=LEN(AG$)-1:T$=RIGHT$(AG$,T)
      :T
=VAL(T$)
220 IF T<0 THEN T=40
230 T=2048*240/T
235 T=T*4
240 RETURN
300 REM П/П ОЗВУЧИВАНИЯ НОТ.
310 W=LEN(AG$)-1:W$=RIGHT$(AG$,W)
      :N
=VAL(LEFT$(W$,2))
320 KN=VAL(MID$(W$,3,2))

```

```

321 IF KN=0 THEN 330
322 IF KN<1 THEN KN=1
324 IF KN>64 THEN KN=64
330 IF N<1 THEN N=1
340 IF N>17 THEN N=17
350 KT=VAL(MID$(W$,5,2))*1.5
360 IF KN=0 THEN DL=T*Z/KN
      :P
UZ=T*(1-Z)/KN:GOTO 380
370 DL=T*Z*KF:PUZ=T*(1-Z)*KF
380 IF KT=0 THEN DL=DL*KT
382 DL$=HEX$(DL):IF LEN(DL$)<4
      TH
EN DL$="0"+DL$
384 IF LEN(DL$)<4 THEN
      DL
$="0"+DL$
386 PRINT #1,STR$(FNH(LEFT$(DL$,2),
      S
$)):PRINT #1,STR$(FNH(RIGHT$(
      (DL$,2),
S$))
388 F=9632*800/(B(N)*(2^(O+1))):
      F$
=HEX$(F):IF LEN(F$)<4
      THEN F$=
"0"+F$
389 IF LEN(F$)<4 THEN F$="0"+F$
390 PRINT #1,STR$(FNH(LEFT$(F$,2),
      S
$)):PRINT #1,STR$(FNH(RIGHT$(
      (F$,2),S
$))
391 IF PUZ=0 THEN NT=NT+4:GOTO 396
392 PUZ$=HEX$(PUZ):IF LEN(PUZ$)<4
      TH
EN PUZ$="0"+PUZ$
393 IF LEN(PUZ$)<4 THEN
      PU
Z$="0"+PUZ$
394 PRINT #1,STR$(FNH(LEFT$(
      PU
Z$,2),S$)):PRINT #1,STR$(
      FNH(RIG
HT$(PUZ$,2),S$))
395 PRINT #1,STR$(0):
      PR
INT #1,STR$(2):NT=NT+8
396 LOCATE 40,15:PRINT SPC(5):
      LO
CATE 40,15:PRINT NTO4:FOR
      IN=1TO99
:NEXTIN:RETURN
400 REM П/П ОПРЕДЕЛЕНИЯ ДЛИТ.
410 L=LEN(AG$)-1:L$=RIGHT$(AG$,L)
      :L
=VAL(LEFT$(L$,2))
420 IF L<1 THEN L=1
440 IF L>64 THEN L=64
445 LT=VAL(MID$(L$,3,2))*1.5
450 KF=1/L:IF LT=0 THEN KF=KF*LT
455 RETURN
500 REM П/П ПРИЕМА ИЗВЛ. ЗВУКА.
505 REM STACCATO
510 IF AG$="MS" THEN Z=3/4:RETURN
515 REM LEGATO
520 IF AG$="ML" THEN Z=1:RETURN
525 REM NON LEGATO
530 IF AG$="MN" THEN Z=7/8:RETURN
540 Z=1:PRINT "НЕВЕРНАЯ КОМАНДА !"
550 RETURN
600 REM П/П ОПРЕД. ПАУЗ В СЕК.
610 P=LEN(AG$)-1:P$=RIGHT$(AG$,P)
      :P

```

```

=VAL(LEFT$(P$,2))
620 IF P<1 THEN P=1
630 IF P>64 THEN P=64
640 PZ=T/P
642 PT=VAL(MID$(P$,3,2))*1.5
      I
F PT>0 THEN PZ=PZ*PT
643 PZ$=HEX$(PZ):IF LEN(PZ$)<4
      TH
EN PZ$="0"+PZ$
645 IF LEN(PZ$)<4 THEN
      PZ
$="0"+PZ$
650 PRINT #1,STR$(FNH(LEFT$
      (P
Z$,2),S$)):PRINT #1,STR$
      (FNH(RIG
HT$(PZ$,2),S$))
655 PRINT #1,STR$(0):
      PR
INT #1,STR$(2):NT=NT+4
660 LOCATE 40,15:PRINT SPC(5):
      LO
CATE 40,15:PRINT NTO4:FOR
      IN=1TO99:
NEXTIN:RETURN
999 CLOSE 1:END
1000 REM П/П ИНСТРУКЦИЯ.
1005 CLS:LOCATE 25,1,1:PRINT
ИНСТРУКЦИЯ":PRINT
      "ВЫ МОЖ
ЕТЕ ВВОДИТЬ СЛЕДУЮЩИЕ
      КОМАНДЫ:"
1010 PRINT "1. ОX,ГДЕ X-НОМЕР
      О
КТАВЫ;":PRINT "2. ТХХХ,ГДЕ
      ХХХ-ТЕМ
П МЕЛОДИИ;":
1015 PRINT "3. NXXYZZ,ГДЕ ХХ-
      Н
ОМЕР НОТЫ,УУ-ДЛИТЕЛЬНОСТЬ,ZZ-
      УВЕЛИЧЕ
НИЕ ДЛИТЕЛЬНОСТИ В ZZ*
      1.5 РАЗ;":
1020 PRINT "4. LXXZZ,ГДЕ ХХ-
      Д
ЛИТЕЛЬНОСТЬ ПО УМОЛЧАНИЮ,ZZ-
      УВЕЛИЧЕ
НИЕ ДЛИТЕЛЬНОСТИ В ZZ*
      1.5 РАЗ;":
1025 PRINT "5. PXXZZ,ГДЕ ХХ-
      Д
ЛИТЕЛЬНОСТЬ ПАУЗ,ZZ-
      УВЕЛИЧЕ
НИЕ ДЛИТЕЛЬНОСТИ В ZZ*1.5
      РАЗ;":
1030 PRINT "6. MS,ML,MN-ПРИЕМЫ
      И
ЗВЛЕЧЕНИЯ ЗВУКА(СТАССАТО,
      LEGATO,
NON LEGATO);":
1032 PRINT "7. YXZZUU,ГДЕ У-
      О
БОЗНАЧЕНИЕ НОТЫ,Х(+,-,ПРОБЕЛ),
      ZZ-ДЛИТ
ЕЛЬНОСТЬ,УУ-УВЕЛИЧЕНИЕ
      ДЛИТЕЛЬНОСТИ
В UU*1.5 РАЗ."
1040 RETURN
2000 REM П/П ОПРЕД. НОМЕРА НОТЫ.
2010 Q$="C C+D-D D+E-E F F+
      G

```

```

-G G+A-A A+B-B "
2015 G$=G$+MID$(AG$,2,1):
      N
N=INSTR(Q$,G$)
2020 IF NN=0 THEN LOCATE 20,15:
      P
PRINT "НЕВЕРНАЯ КОМАНДА !":FOR
      IN=1TO5
0:NEXTIN:RETURN
2025 NN=(NN+1)/2:NN$=STR$(NN):
      N
N$=MID$(NN$,2,LEN(NN$)-1):IF
      LEN(NN$
)<2 THEN NN$="0"+NN$
2030 G$=MID$(AG$,3,2):GG=VAL(G$):
      G
$=STR$(GG):
      G$=MID$
(G$,2,LEN(G$)-1):
      IF LEN(G$)<2
THEN G$="0"+G$
2035 DD$=MID$(AG$,5,2):DD=VAL(DD$):
      D
D$=STR$(DD):
      DD$=MID
$(DD$,2,LEN(DD$)-1):IF
      LEN(DD$)<2 TH
EN DD$="0"+DD$
2040 AG$="N"+NN$+G$+DD$:G$="N"
2045 LOCATE 20,15:PRINT AG$:
      F
ORIN=1TO50:NEXTIN
2050 GOSUB 300
2060 RETURN
3 REM *****"MELOD"*****
4 REM ПРОГРАММА
      ПРОИ
ГРЫВАНИЯ МЕЛОДИИ.
5 REM АВТОР АРТАМОНОВ В.С.
6 REM *****
7 GOSUB 100:MERGE "FILE.BAS"
9 CLEAR 2000
17 Z=1:PRINT "КОЭФФИЦИЕНТ
      ДЛИ
ТЕЛЬНОСТИ(1 ИЛИ ЧИСЛО
      МЕНЬШЕЕ 1
5)":INPUT Z
20 I=1:DIM M(5000):INPUT #1,M$
22 PCLS:CLS:PRINT M$
24 IF EOF(1) THEN 30
26 INPUT #1,M(I)
28 I=I+1:GOTO 24
30 I=I-1:FOR J=1 TO I STEP 4
40 A=M(J):B=M(J+1):
      C=M
(J+2):D=M(J+3)
45 A=A*Z
47 IF A>255 THEN A=255
50 POKE &HF718,A
60 POKE &HF717,B
70 POKE &HF716,C
80 POKE &HF715,D
90 BEEP
95 NEXT
99 CLOSE 1:END
100 REM ПОДПРОГРАММА С ИНСТРУКЦИЯМИ.
110 PCLS:CLS:LOCATE 20,2,1:PRINT
      "9
ЛЕКТРОННЫЙ ОРКЕСТР"

```

```

120 PRINT "ВВЕДИТЕ ИМЯ ФАЙЛА С
      МЕ
ЛОДИЯ",":PRINT "ПОСЛЕ ПОЯВЛЕНИЯ
      НА ЭКРАНЕ
Е "Ок" ВВЕДИТЕ:GOTO9<БК>"
130 INPUT NM$
140 NM$="10 OPEN "+CHR$(34)+"
      "I

```

```

"+CHR$(34)+",#1,"+CHR$(34)+
      NM$+CHR$(
(34)
150 OPEN "O",#1,"FILE.BAS"
155 PRINT #1,NM$:CLOSE 1
160 RETURN

```

Е. ЕРЕМИН

## Улучшение Турбо-Паскаля для «Ямахи»

58 Широкое распространение получил Турбо-Паскаль. Он очень удобен в обращении и как нельзя лучше подходит для обучения программированию. К сожалению, в дошедшей до нас второй версии Турбо-Паскаля для «Ямахи» не работает опция компилятора U+, обеспечивающая прерывание программы с клавиатуры. Приходится применять обычный режим трансляции, в результате которого обучаемый порой не может прекратить выполнение своей программы и найти допущенную ошибку. Анализ показал, что причина этого недостатка крайне проста: для опроса клавиатуры после исполнения каждого оператора система Турбо генерирует команду RST 38, которая в «Ямахе» применяется для других целей. Поэтому при компиляции нужно включать в программу другую аналогичную команду, например RST 28, которая в стандарте MSX свободна. В файле TURBO.COM достаточно изменить всего четыре байта (последнее изменение не яв-

ляется обязательным и рекомендуется лишь для слабо подготовленной аудитории):

Адрес	Старое значение	Новое значение	Комментарий
39C 3A2	38 39	28 29	Занесение команды перехода по RST28
5282	FF	EF	Замена кода RST38 на RST28 в компиляторе
4479	35	3D	Установка режима 0+ по умолчанию

После внесения указанных изменений нажатие комбинаций клавиш CTRL+STOP или CTRL+C будет останавливать исполнение программы.

С. ПУСТОВОЙТОВ

## Об одной особенности КУВТ-86

Работающие с КУВТ-86 часто попадают в следующую ситуацию: на ДВК происходит останов по адресу 2 с выходом в монитор или выход из загруженной программы в командный монитор KMON. Эти сбои вызваны прерываниями по сети при включении или выключении БК-0010 или нарушении контакта между блоком ИРПС и БК. Обычно эти прерывания замаскированы, но программы NET.SAV и RIGA.SAV (режимы 0 и 8) снимают с них запреты. После выхода из этих программ прерывания остаются разрешенными, что и приводит к сбоям, так как стандартными средствами RT-11 они не обрабатываются.

Предлагаемая программа маскирует эти

прерывания и полностью устраняет сбой подобного рода. Написанная на MACRO-11, она занимает после трансляции два блока. Ее удобно поместить на резидентный диск и вызывать командой R после каждого использования программ NET или RIGA.

```

MOV #24.,R0
MOV #176500,R1
18: BIC #100,(R1)+
TST (R1)+
SOB R0,18
CLR R0
EMT 350

```

Адрес для справок: 310168, Харьков, а/я 2996, компьютерный клуб «JNP».

**Кафедра автоматики и процессов управления  
Ленинградского электротехнического института  
им. В. И. Ульянова (Ленина)**

**предлагает**

**Вниманию пользователей УМК**

Предлагается дополнительное аппаратное, программное и методическое обеспечение для учебных микропроцессорных комплексов (УМК) производства ВЭФ, которое позволяет:

использовать УМК в качестве контроллера для управления технологическим оборудованием или его макетом;

подключать к УМК дисплей с внешним интерфейсом типа С2 (ИРПС);

пользоваться при программировании на УМК резидентным экранным редактором и транслятором с языка ассемблера;

объединять в локальную сеть с радиальной структурой до 8 УМК;

использовать в сети в качестве координирующей микроЭВМ типа МС 0511, ДВК, при этом появляется возможность хранения программ УМК на НГМД и распечатки листингов на принтере.

Реализация указанных возможностей обеспечивается аппаратно:

внешним модулем УМК (устанавливается в разъем на лицевой панели), содержащим программируемое устройство ввода-вывода дискретной информации на 20 каналов, аналого-цифровой преобразователь, трехканальный программируемый таймер, блок приоритетных прерываний, два канала обмена информацией в последовательном коде, дополнительные ОЗУ и ПЗУ;

модулем сетевого концентратора на 8 каналов ИРПС;

программно:

резидентным программным обеспечением, содержащим драйвер терминала, экранный редактор и ассемблер;

программами сетевой поддержки.

На базе разработанных средств реализован лабораторный практикум по изучению микропроцессорного комплекта БИС серии КР580, основ программирования в кодах процессора КР580ИК80 и на языке ассемблера. Методическое обеспечение лабораторного практикума содержит практически все справочные материалы, необходимые для изучения технических средств и для разработки и отладки программ, благодаря чему отсутствует необходимость в использовании дополнительной литературы.

**Вниманию пользователей КУВТ  
«Электроника МС 0202»**

Предлагается сетевое программное обеспечение «Корнет» для комплекса учебной вычислительной техники УКНЦ «Электроника МС 0202».

«Корнет» позволяет группе из 8 пользователей получать практические навыки работы с дисковой операционной системой (ФОДОС или RT-11), языками программирования, СУБД, пакетами прикладных программ и т. д.

**Отличительные особенности:**

виртуализация дискового устройства рабочего места ученика (РМУ) (виртуальный диск — физически отсутствующее, но логически доступное благодаря программно-аппаратной эмуляции псевдодискового устройство);

серверный режим работы рабочего места преподавателя (РМП).

**Возможности:**

начальная загрузка («холодный» старт) РМУ с одного из виртуальных дисков; обслуживание поступающих одновременно с разных РМУ программных запросов на чтение-запись виртуальных дисков. Каждому ученику доступны по

чтению все 8 виртуальных дисков; по записи (во избежание искажения каталога из-за одновременного его обновления) доступен только свой (системный).

**Состав:**

драйвер виртуального диска;  
драйвер антидисковый;  
сервер виртуального диска;  
инструкция по установке.

***Комплекс аппаратно-программных средств для  
распределения систем контроля и управления  
технологическими объектами и оборудованием***

**на базе однокристалльных микроЭВМ и персональных компьютеров,  
совместимых с IBM PC/XT/AT**

**В состав комплекса входят:**

одноплатные свободно программируемые контроллеры;  
адаптеры связи;  
устройства ввода-вывода аналоговых и дискретных сигналов;  
средства обработки и представления информации.

**Области применения:**

распределенные системы сбора и обработки данных о состоянии технологического оборудования;  
локальные системы контроля и управления;  
распределенные системы комплексной автоматизации;  
автоматизированные испытательные комплексы;  
системы контроля и диагностики процессов, материалов и машин;  
автоматизированные системы промышленных исследований.

**В комплексе обеспечены:**

совместимость аппаратно-программных средств между собой, с персональными компьютерами типа IBM PC/XT/AT и со стандартными схемами ТТЛ-уровня;  
модульность и открытость программного обеспечения;  
самодиагностика всех видов устройств;  
возможность безвахтенного обслуживания;  
оптимальность аппаратно-программных и конструктивных решений для каждой конкретной задачи;  
минимальная стоимость при максимальном удовлетворении требований заказчика;  
простота и надежность в эксплуатации.

**Предоставляемые услуги:**

исследование объектов управления;  
разработка критериев и оптимальных алгоритмов сбора информации, контроля и управления;  
разработка специализированных аппаратных средств;  
разработка проблемно-ориентированного программного обеспечения;  
комплексирование и поставка систем требуемой конфигурации.

**Технические характеристики комплекса:**

процессор КМ1816ВЕ48(51) или UB8840;  
ПЗУ (ППЗУ) от 3 до 64К байт;  
ОЗУ программ и данных от 1 до 64К байт;  
программируемый таймер;  
последовательный канал связи ИРПС (RS-232);  
параллельный канал связи — 24, 32 разряда;  
система прерываний — от 2 до 8 источников;  
тактовая частота — 6, 8, 12 МГц.

**Заявки направлять по адресу: 197022, Ленинград, ул. профессора Попова, 5,  
ЛЭТИ им. В. И. Ульянова (Ленина), кафедра АиПУ.**

**Телефон: 234-37-98.**

Ю. ЗАЛЫЦМАН

## Архитектура и ассемблер БК-0010

### Архитектура БК

#### Системные регистры

Совершая путешествие по нашему «городу» БК-0010, мы отметили, что ПЗУ тянется почти до конца адресного пространства — до адреса 177600. А что же дальше? Дальше, с адреса 177600 и до конца, по адрес 177776, располагается так называемая *область системных регистров*.

Системные регистры микроЭВМ служат для связи с внешними устройствами и некоторых других целей. Каждый регистр занимает в адресном пространстве одно слово, следовательно, имеет 16д разрядов, с нулевого по пятнадцатый. Простой арифметический подсчет показывает, что в области адресов 177600 — 177776 содержится 100 слов, значит, там можно разместить 100 (64д) регистров. Но далеко не все адреса этой области использованы в нашем компьютере. Мы рассмотрим имеющиеся в БК-0010 системные регистры в порядке возрастания их адресов. При этом нужно учитывать, что и в имеющихся регистрах далеко не все разряды (биты) используются. Некоторые разряды хранят только определенную, строго фиксированную информацию, которую нельзя изменять. В других разрядах информацию меняет только сама ЭВМ, а программист может ее лишь читать и использовать результаты чтения (говорят, что такие разряды *доступны по чтению*). В третьи разряды регистров можно, наоборот, только записать информацию, а прочесть записанное нельзя (разряды, *доступные по записи*). Наконец, есть разряды, в которые можно записывать информацию и читать ее (они доступны по записи и чтению).

Чем системные регистры (например, до-

ступные по записи и чтению) отличаются от ячеек ОЗУ? Отличие существенное — информацию в ОЗУ может записывать, читать и использовать только ЦП. Разряды же системных регистров могут быть связаны непосредственно с внешними устройствами — клавиатурой, магнитофоном, телеграфной линией, — словом, с внешним миром, могут принимать оттуда сигналы или, наоборот, передавать. Это, образно говоря, те руки и ноги, с помощью которых ЦП может преобразовать свои «мысли» — информацию — в конкретные действия или получить информацию извне. Есть, правда, регистры, казалось бы ни с чем не связанные, например таймер. Но это лишь иллюзия: таймер тоже связан с внешним миром — через физическое время!

А что же отсутствующие регистры? Это резерв для расширения нашей микро-ЭВМ в будущем; так сказать, пустующие пока «участки под застройку» (к слову сказать, в машинах серии ДВК, а также в БК-0010Ш, работающих в составе комплекта учебной вычислительной техники КУВТ-86, часть из них уже используется).

Итак, приступим к прогулке по последнему «переулку» адресного пространства. Напомним, что каждый регистр имеет 16 разрядов, нумеруемых справа налево, причем нумерация начинается с нуля. Номера разрядов мы будем использовать десятичные. По ходу перечисления разрядов регистров мы будем также указывать, для чего можно использовать тот или иной разряд при программировании (как их использует ЭВМ — другой вопрос).

**1. Регистр состояния клавиатуры** (адрес 177660). Используются только два разряда: разряд 06 — маска прерываний от клавиатуры. Если состояние разряда 0, то прерывание разрешено, 1 — запрещено. Разряд доступен по записи и чтению. Можно исполь-

зовать, чтобы программно отключить клавиатуру ЭВМ;

разряд 07 — флаг состояния клавиатуры. Устанавливается в 1 при поступлении новых данных в *регистр данных клавиатуры* (см. ниже), например при нажатии очередной клавиши; в 0 — при чтении кода из регистра данных. Разряд доступен только по чтению. Если запретить автоматическое чтение из регистра данных, то этот разряд можно использовать, чтобы определить, была ли нажата клавиша за истекший промежуток времени. В обычных условиях, когда данные читаются сразу после нажатия клавиши, состояние разряда всегда 0, «поймать» единицу в нем нельзя.

**2. Регистр данных клавиатуры** (адрес 177662). Используются семь младших разрядов.

Разряды 00—06 — буфер кода нажатой клавиши. При нажатии клавиши туда заносится семиразрядный код клавиши, на который не влияют клавиши **НР** (**AP2** для БК-0010.01) и **РУС**, т. е. код этот — только в регистре **ЛАТ**. Полный код клавиатуры получается в БК-0010 программным путем. Разряды доступны только по чтению. Широко используются для чтения кода клавиатуры независимо от регистра **РУС** — **ЛАТ**, для ввода команд без прерывания работы программы (в том числе и в программах на языках высокого уровня) и т. п.

**3. Регистр смещения** (адрес 177664). Используются девять разрядов.

Разряды 00—07 отображают адрес начала экранного ОЗУ, которое организовано по типу рулона, причем единица содержимого регистра соответствует одной телевизионной строке дисплея, или 100 байтам (64d) экранного ОЗУ. В исходном состоянии, когда началу экрана соответствует адрес 40000, содержимое этих разрядов — 330. По мере сдвига экрана содержимое меняется, причем из-за того, что одна строка текста содержит в БК-0010 12 (10d) телевизионных строк, с каждым сдвигом экрана содержимое регистра также меняется на 12. В начале экрана, как известно, имеется служебная строка, занимающая в экранном ОЗУ 2000 байтов (каждая строка текста — 1200 байтов). Эти тонкости приводят к тому, что число сдвигов, необходимое для того, чтобы начало экрана получило снова адрес 40000, некратно числу строк текста (24d), и содержимое младших разрядов данного регистра практически все время разное. Разряды доступны по записи и чтению и могут использоваться для плавного (а не на строку текста) сдвига изображения на экране по вертикали. Служебная строка при этом, од-

нако, тоже смещается, занимая положение в середине экрана, поэтому данный способ требует специальных приемов получения изображения на экране.

Нужно отметить, что в режиме РП экран организован совсем иначе, чем описано, служебная строка (начало экрана) всегда имеет адрес 70000, рулонное смещение не используется, а регистр всегда содержит константу 230.

Разряд 09 — задание режима расширенной памяти (0 в данном разряде) или обычного режима (1) (в «Руководстве системного программиста» ошибочно указано наоборот). Разряд доступен по записи и чтению. Практически может использоваться для обнаружения режима РП, включить же РП, просто записав сюда 0, нельзя.

**4. Регистры системного таймера** (адреса 177706, 177710, 177712). Эти три регистра не описаны ни в одном руководстве по БК-0010, поэтому они заслуживают подробного рассмотрения.

Прежде всего, что такое системный таймер? Это специальный счетчик, работающий самостоятельно и независимо от ЦП. Если запустить такой счетчик, а потом в какие-либо фиксированные моменты времени читать его показания, то по разности показаний можно определить прошедший отрезок времени.

Итак, наш таймер имеет три регистра. Первый из них — *буфер константы цикла таймера* (адрес 177706) доступен по записи и чтению. Второй — *счетчик* (адрес 177710), собственно таймер, доступен только по чтению. Третий — *регистр управления* (адрес 177712) доступен по записи и чтению.

Рассмотрим, как работает таймер. Если мы запишем в регистр 177706 какое-либо число (а максимальное число, записываемое в 16-разрядный регистр, равно 65535d, или 177777), а затем подадим определенную последовательность команд (запишем в регистр 177712 некоторую последовательность кодов), то таймер начнет работать — число из 177706 переписывается в 177710 и там начнется вычитание из него в определенном темпе по единице — обратный счет времени. Как только содержимое 177710 станет равно нулю, в него снова будет переписано число из 177706, снова начнется вычитание и т. д. Задавая в 177706 различные числа, можно изменять *время цикла таймера* — это «тонкая регулировка» хода наших часов. Но есть и грубая — в зависимости от записанного в 177712 кода можно замедлять ход в 4, 16 и 64 раза. Что это за коды? Прежде всего, хотя в регистре 177712 доступны для записи все восемь младших разрядов, рабочими являются только три из них:



разряд 04 — пуск таймера;  
 разряд 05 — множитель 16;  
 разряд 06 — множитель 4.

При работе таймера отсчитанный промежуток времени определяется умножением разности показаний счетчика за истекшее время на те множители, в разряды которых записаны единицы, и на единичный период времени. Этот период в среднем равен 42,9 мкс и может для разных экземпляров БК варьировать в пределах нескольких процентов.

Что же касается разряда 04, то при работе таймера там всегда должна быть единица, а если обнулить этот разряд (или весь регистр 177712), то таймер остановится и «сбросится на начало», т. е. в регистр 177710 переписывается константа из 177706. В неиспользуемых разрядах 177712 должны быть нули.

Рассмотрим пример. Пусть мы, запуская таймер, записали в регистр 177712 код 160 (единицы в разрядах 04, 05 и 06) и получили разность показаний таймера за отсчитанный промежуток времени, равную 154432, или 55578д. Умножим:  $55578 \times 4 \times 16 \times 42,9 = 152594956,8$  мкс, или примерно 152,6 с.

Естественно, чтобы отсчитать такую разность, необходимо иметь в регистре 177706 число, большее 154432. Приведем табличку максимальных интервалов времени, которые в состоянии отсчитать таймер в пределах одного цикла.

Код в 177712	Множитель	Максимальное время цикла, с
20	1д	2,812
120	4д	11,25
60	16д	45,00
160	64д	180,0

Как ясно из изложенного, максимальный цикл задается числом 177777, занесенным в регистр 177706 (кстати, если просто обнулить 177706, то после первого же вычитания произойдет «заем» в старшем разряде и в регистре окажется число... 177777).

Итак, таймер позволяет отсчитывать промежутки времени до 3 минут. А если надо больше? Тогда придется написать программу, которая могла бы периодически опрашивать регистр 177710, отмечать моменты перехода его «через ноль» и считать число переходов. Цикл при этом целесообразно задать таким, чтобы он был кратен какой-либо единице времени. Попробуем, например, задать время цикла равным 1 минуте.

Ясно, что нам подходит лишь код 160 —

все другие дают время цикла менее 1 мин. Подсчитаем константу.  $1 \text{ мин} = 60000000 \text{ мкс}$ ,  $(60000000/64)/42,9 = 21853$  (округленно). Переведем в восьмеричную систему:  $21853_{10} = 52535_8$ . Остается записать полученное число в 177706, а код 160 — в 177712, и счетчик таймера будет выполнять цикл длительностью 1 минуту. Подобные эксперименты удобно проводить в МСД, используя для непрерывного чтения регистра 177710 команду Ц.

Для чего еще может понадобиться таймер кроме отсчета времени? Это идеальный генератор случайных чисел, если только обращения к нему производятся не программой, а человеком (например, по нажатию клавиши). Такой генератор будет давать не псевдослучайные, а истинно случайные числа, так как момент нажатия клавиши оператором зависит от очень многих факторов и угадать показание таймера в этот момент невозможно. Если же к таймеру обращается программа, последовательность будет далека от случайной: ведь и таймер, и процессор используют одну и ту же тактовую частоту, поэтому их деятельность коррелирует.

Еще несколько замечаний. Системный таймер работает независимо от процессора, поэтому его счетчик считает время независимо от того, стоит процессор или работает, есть в БК работающая программа или нет. Таймер одинаково работает в Бейсике, Фокале, с программами в кодах, запущенными из МСД и ПМ и даже в случае, если БК-0010 работает с магнитофоном. Но опрос регистра 177710 необходимо проводить не реже, чем 2 раза за цикл таймера, иначе можно не выявить переход цикла через 0, и показания «часов» исказятся. Таймер, в свою очередь, никак не влияет на работу программы пользователя, в том числе и на скорость ее исполнения, своей деятельностью (или бездействием).

И наконец, одно не совсем приятное замечание. Поскольку таймер по техническим условиям в состав БК-0010 не входит, то на некоторых БК его регистры могут быть неисправны, что не является основанием для предъявления претензий заводу-изготовителю...

При выходе в Бейсик или Фокал таймер отключается, поэтому запускать его нужно в том режиме, в котором он будет работать.

5. Регистр порта ввода-вывода (адрес 177714). Используются все 16 разрядов, они доступны по записи и чтению, но с особенностью: данные, записываемые по адресу 177714, пишутся в регистр вывода, а читаются данные по этому адресу из регистра ввода. Прочитать данные, записанные в регистр вывода, невозможно, поэтому, если не-

обходимо сохранить к ним доступ, нужно их записать не только в порт, но и в специально зарезервированную для этого ячейку памяти (адрес 256).

Порт вывода инвертирует подаваемый на него код, т. е. при записи в него нулей на выходе оказываются единицы и наоборот. То же самое и в порте ввода — при подаче на него от внешних устройств логических единиц программно читаются нули. При «привязке» к порту внешних устройств это нужно учитывать.

**6. Регистр системных внешних устройств** (адрес 177716). Используются все 16 разрядов, но для разных целей.

Разряды 0—3 предназначены для внутренних целей ЭВМ, служат для задания режима работы процессора, доступны только по чтению.

Разряды 4—7 предназначены для управления системными внешними устройствами ЭВМ (магнитофоном, телеграфным (ТЛГ) каналом, клавиатурой, встроенным динамиком). Они представляют собой как бы внутренний порт ввода-вывода и организованы совершенно так же, как только что описанный внешний порт; это два отдельных регистра — ввода и вывода, поэтому запись и чтение информации по этим разрядам имеют разное значение.

Разряды регистра ввода предназначены: 04 — для чтения информационного сигнала с ТЛГ-линии;

05 — для чтения информации с магнитофона;

06 — индикатор нажатия клавиши. Если нажата любая клавиша (кроме **СТОП**, **НР**, **ПР**, **СУ**, **СТР**, **ЗАГЛ**), то в этом разряде 0, если ни одна из клавиш не нажата — 1. Широко используется в программах для индикации нажатия клавиши, ввода информации «на ходу», без остановки программы, организации циклов автоповтора и т. п. В самой ЭВМ используется для организации режима «повтор»;

07 — для чтения сигнала готовности с ТЛГ-линии. Он, как и 04, может использоваться не только по прямому назначению, но и для других целей, это дополнительный канал связи ЭВМ с внешними устройствами. Для задействования ТЛГ-ввода-вывода на плате ЭВМ необходимо установить перемычки.

Разряды регистра вывода предназначены: 04 — для передачи информации на ТЛГ-линию, его исходное состояние — 1;

05 — для передачи сигнала на магнитофон (при записи) либо сигнала готовности на ТЛГ-линию (позтому одновременный обмен информацией с магнитофоном и ТЛГ-линией невозможен), исходное состояние — 0.

Разряд 06 служит для передачи информации на магнитофон (при записи) и сигнала на пьезодинамик ЭВМ (при записи на МЛ и нажатии клавиши). Очень широко используется в программах для создания звуковых эффектов. Если с определенной периодичностью записывать в этот разряд чередующиеся 0 и 1, мы услышим звук, воспроизводимый как встроенным пьезодинамиком, так и подключенным к выходу БК «Магнитофон» любым усилителем низкой частоты. Исходное состояние — 0.

Тут необходимо пояснить, почему для выдачи информации на магнитофон используются два разряда — 05 и 06. Дело в том, что в БК-0010 для записи на магнитофон принят двухуровневый сигнал; при записи «нуля» сигнал идет с обоих разрядов и имеет большую амплитуду, а при записи «единицы» только с разряда 06, и амплитуда его меньше. Это сделано для коррекции частотной характеристики тракта записи магнитофона и получения более качественных записей. Кстати, это одна из причин, по которым ни одна схема «регенерации» (т. е. прямой перезаписи с магнитофона на магнитофон) не обеспечит такого же качества, как перезапись через БК.

Разряд 07 служит для дистанционного управления двигателем магнитофона — при записи туда нуля двигатель включается, а при записи единицы отключается. Управление двигателем осуществляется через установленное на плате БК электромагнитное реле.

Разряды 08—15 предназначены для задания адреса запуска системы, доступны только по чтению. Адрес запуска в БК-0010 принят равным 100000, он и задан в старшем байте регистра 177716, причем считается, что при чтении адреса запуска его младший байт равен 0.

*Продолжение следует*

## Ассемблер БК

**5. Автодекрементная косвенная адресация.** Если сказать, что *декремент* — это уменьшение чего-либо, то этот способ будет вполне понятен, не правда ли? Его отличие от автоинкрементной косвенной адресации в двух пунктах: во-первых, адрес регистра-указателя не увеличивается, а уменьшается; во-вторых, это происходит не после выполнения команды, а до, на что намекает и его запись: — (RN). Знак «—» стоит, в отличие от «+», перед именем регистра. В остальном сходство с автоинкрементным способом полное, включая и то, что при операциях с байтом модификация числа в регистре де-

лается на 1, а со словом — на 2, и то, что регистр SP всегда модифицируется только на 2. Поясним этот способ примером.

Пусть в регистре R2 число 1002. Тогда команда

```
CLR    -(R2)
```

очистит ячейку 1000, т. е. ту, адрес которой на 2 меньше, чем записанное в регистре число, а содержимое регистра, естественно, станет равно тоже 1000 — оно стало таким еще до очистки ячейки. Видимо, дальнейших примеров не нужно, по аналогии с автоинкрементным способом все и так ясно. Этот способ позволяет делать, в общем, то же, что и автоинкрементный, но только если мы работаем с массивом, то просматривать или пересылать его при этом будем «с конца», а перед началом операции в регистре должен быть не адрес конца массива, а адрес следующей за ним ячейки. Однако этот способ позволяет делать и нечто более интересное. Пусть в R2 записано число 2004, а по адресу 2002 — число 1254. Команда

```
MOV    -(R2), -(R2)
```

поместит число 1254 в ячейку 2000, т. е. как бы передвинет его в памяти с большего адреса на меньший! Разумеется, по адресу 2002 число 1254 сохранится тоже. Как это произошло? Вначале содержимое R2 уменьшилось на 2, стало равно 2002, и регистр указал на наше число 1254. Число мы «поймали». Теперь оно будет записано по адресу второго операнда, но его указатель — опять тот же регистр, и опять вначале его содержимое уменьшится на 2, станет равно 2000, а потом уже туда будет записано число. Нетрудно видеть, что таким образом можно «сдвигать» массивы в памяти, удалив в середине массива какой-либо элемент. Возникает вопрос: где же хранятся при таких «фокусах» адреса операндов, ведь они модифицируются и в то же время сохраняются до конца команды? Ответ на него выходит за рамки программирования — промежуточные данные хранятся в особых буферных регистрах процессора, недоступных программно и поэтому нас не интересующих. Довольно знать, что команда выполняется именно так, а не иначе.

А теперь — маленькое развлечение и одновременно — интересный пример по практике использования автодекрементной адресации. Запишем команду

```
MOV    -(PC), -(PC)
```

Ее код — 14747; для большего эффекта по-

3 Информатика и образование № 6

местим ее в «конец» ОЗУ, в ячейку 77776 (с помощью директив МСД это сделать просто, и никакого ассемблера не требуется: 77776A14747И), а затем передадим на нее управление, т. е. запустим «программу» по адресу этой команды: 77776. Что случилось?! Экран мгновенно стал полосатым, а ЭВМ... Она «мертва»: не отвечает на нажатие клавиш и даже на такое «мощное лечебное средство», как клавиша СТОП, никак не реагирует! Только системный сброс или выключение, а затем включение питания «приводит ее в чувство». Что же это за бомба, которая мгновенно парализовала все средства ЭВМ?

Рассмотрим, что происходит при выполнении данной команды. После запуска в регистре PC — адрес следующей команды, т. е. число 100 000. Но при обращении к первому операнду содержимое PC уменьшается на 2, давая адрес 77776, а там — код самой команды, т. е. число 14 747. А куда оно перешлетя? По адресу второго операнда, но перед тем содержимое PC уменьшится еще на 2 и станет равно 77774, т. е. укажет уже на следующую («нижележащую») ячейку памяти. А потом, раз в регистре PC адрес 77774, туда и будет передано управление, а тем снова та же команда, тот же «смертельный» код! Команда как бы размножается в памяти ЭВМ, причем с огромной скоростью: чтобы заполнить собой всю память, ей нужно лишь 0,2 секунды! А почему ЭВМ отказывает? Это станет ясно, если вы вспомните, что в зоне адресов 0—777 размещается системная область и стек ЭВМ, как бы ее собственная память, «мозг». А если стереть содержимое мозга любого существа, пусть даже такого простого, как ЭВМ, — наступит если не смерть, то безумие... Так что же это все-таки за команда, зачем мы потратили на ее разбор столько времени и для чего она может пригодиться?

Вы, наверное, слышали о компьютерных вирусах, но еще не встречались ни с одним из них, не правда ли? Так вот, позвольте представить: MOV — (PC), — (PC) — простейший вирус, вызванное им заболевание развивается молниеносно и безусловно смертельно. Эта команда уничтожает всю информацию, хранимую в ОЗУ по адресам меньше, чем адрес самой команды. А зачем может понадобиться такая страшная команда, ведь вирусы — это безусловное зло? Не совсем так! Существует ряд случаев, когда нужно срочно уничтожить хранимую в ОЗУ информацию. Не касаясь даже таких экзотических вещей, как секретность, государственные тайны и т. п., можно придумать массу поводов для такого акта. Например, если вы хотите, чтобы ваша игра давала после запуска только определенное число попыток

пройти лабиринт, а потом ее нужно было снова «грузить» с МЛ (это затруднит нахождение решений), поместите в любом месте программы вирус, и, когда число попыток исчерпано, пусть он «оживет». Более сложно, но возможно реализовать активизацию вируса, например, по истечении заданного времени, или после определенного числа копирований программы на МЛ, или при попытке что-либо изменить в программе. Как видите, даже вирусы можно заставить приносить пользу. Но это еще не всё. Изучая вирусы, создавая их и борясь с ними, программисты совершенствуются уже не в своем ремесле, а в своем искусстве. Это тоже приносит пользу, хотя и является обычно слабым утешением и отнюдь не покрывает вред, приносимый вирусами.

Но вернемся к способам адресации.

6. Автодекрементная двойная косвенная адресация. Записывается как @ — (RN). Если сделать поправки, уже нам известные, что декремент — это уменьшение, выполняющееся до обращения к числу, на которое указывает операнд, то из автоинкрементной двойной косвенной адресации легко получается автодекрементная двойная косвенная. Даже не детализируя дальше, приведем лишь пример. Пусть в регистре R2 — число 1002, по адресу 1000 — число 1254, а по адресу 1254 — число 3333. Тогда команда

```
MOV @-(R2),R4
```

запишет в R4 число 3333. Данный способ может быть применен для тех же целей, что и автоинкрементная двойная косвенная адресация, но дает возможность, например, просматривать первую таблицу-указатель с конца.

К сожалению, такого, казалось бы, логичного способа, как «двойная косвенная адресация», который мог бы записываться, например, как @(RN), не существует; а ведь для чего-нибудь он бы пригодился... Однако мы можем «обмануть» ассемблер и все-таки задать адрес таким образом! Как именно — скажем в дальнейшем.

Отметим также, что все уже рассмотренные способы адресации «вписываются» в формат самой команды и, таким образом, команда (оператор и операнды) при переводе в машинный код занимает одно слово. Способы, которые мы рассмотрим далее, таким преимуществом не обладают, они требуют задания некоторых дополнительных параметров, в формат одного слова не укладываемыхся. Соответствующие команды занимают два, а то и три машинных слова, но эта «растрата» памяти компенсируется предоставляемой программисту дополнительной гибкостью и универсальностью средств

программирования на ассемблере.

Прежде чем перейти к описанию других методов адресации (а мы исчерпали едва половину их разнообразия!), введем еще два общих обозначения:

«X» — восьмеричное число;

«MET» — произвольное имя метки.

7. Индексная адресация. Это разновидность косвенной адресации через регистр. Записывается X(RN) или MET(RN). Стоящее перед регистром восьмеричное число X или имя метки MET называется *индексом*; он означает *смещение* от адреса, который содержится в регистре, и становится при трансляции вторым или третьим словом команды. Восьмеричное число может быть записано как без знака, так, в последних версиях ассемблеров, и со знаком «—» (в этом случае оно автоматически переводится при трансляции в дополнительный код), а вместо имени метки ассемблер тоже подставляет число — ее абсолютный адрес, полученный в результате трансляции и компоновки. Таким образом, при данном способе адресации адрес ячейки, к которой мы обращаемся, определяется как сумма смещения (индекса) и содержимого регистра RN (так называемой *базы*). Приведем примеры. Пусть в регистре число 2000, тогда команда

```
CLR 1000(R2)
```

обнулит слово по адресу 3000, а команда

```
CLR -1000(R2)
```

обнулит слово по адресу 1000. Этот способ может быть использован, если нужно выполнить действия с несколькими ячейками, отстоящими от заданного адреса на фиксированные смещения. Например, если задать в регистре R2 адрес какого-либо байта экранного ОЗУ, то команда

```
CLR 100(R2)
```

сотрет байт, находящийся в следующей телевизионной строке, снизу от заданного, а команда

```
CLR 1200(R2)
```

сотрет байт, лежащий под заданным в 100 строках.

Еще интереснее случай применения в качестве индекса имени метки. Предположим, что нам нужно вместо заносимых в регистр R2 чисел, например, 0, 1, 2, ..., 7, подставлять значения 4, 3, 2, 1, 0, 7, 6, 5 соответственно. Эта задача называется *перекодировкой*. Способ ее решения «в лоб» —

проверять каждый раз содержимое регистра на соответствие всем значениям ряда по очереди. Он ужасно неэкономичен как по времени, так и по занимаемому объему ОЗУ (представьте, что возможных кодов — 1000). Лучше применить *табличный* метод. Напишем программу, состоящую всего из одной (!) команды

```
MOV TAB(R2),R2
```

и составим таблицу перекодировки, которая в нашем примере будет иметь вид

```
TAB: .B: 4, 3, 2, 1, 0, 7, 6, 5
```

(забегая вперед, поясним, что команда .B: позволяет записать ряд чисел в последовательные байты памяти начиная с текущего адреса, т. е. в нашем случае с адреса метки TAB).

Задача решена! Пусть, например, в регистре R2 число 3. Адрес первого операнда команды будет вычислен как сумма адреса метки TAB и содержимого регистра. Отсчитаем начиная с нуля третий байт таблицы — это число 1. Оно и будет занесено нашей командой вместо исходного числа 3 в регистр R2 — перекодировка выполнена.

Круг аналогичных задач весьма широк — от замены символов текста на другие (например, если набор символов, которые печатает принтер, ограничен и не соответствует набору символов БК-0010) до вычисления значений произвольной функции по аргументу табличным методом.

**8. Индексная косвенная адресация.** Записывается как @X(RN) или @MET(RN). По смыслу близка к предыдущей, но в ячейке, определяемой суммой индекса и содержимого регистра, теперь не само число-операнд, а его адрес. С таким явлением — двойной косвенностью — мы уже встречались, поэтому приводить примеры не будем, а лучше выполним обещание — покажем, как реализовать двойную косвенную адресацию через регистр, которой в наборе команд ЭВМ вроде бы нет. Как мы уже говорили, логично было бы представить запись такой адресации в виде @(RN), а мы вместо этого запишем: @0(RN). Вот теперь цель достигнута — мы обратились к ячейке с двойной косвенностью, без модификации содержимого регистра-указателя. Правда, за этот «обман» придется заплатить — команда занимает целых два (или три) машинных слова, хотя содержимое второго (или третьего) слова — лишь ноль...

Индексная косвенная адресация применяется редко, но без нее иногда почти невозможно обойтись, например, когда нужно не просто перекодировать последовательность чисел, а сравнить перекодированную

последовательность с некоторой второй последовательностью (возможно, тоже перекодированной!), не меняя ее. С такой необходимостью вы непременно столкнетесь, если захотите написать программу сортировки по алфавиту русских слов — ведь последовательность символов русского алфавита не совпадает с порядком нарастания их кодов.

**9. Непосредственная адресация.** Этот способ позволяет в качестве одного из операндов записать число (константу). Число это записывается как #X или #MET. Значок #, называемый на жаргоне программистов «решетка», в ассемблере как раз и обозначает число. Восмеричное число-операнд может быть без знака или же со знаком «—», а имя метки ассемблер при трансляции и компоновке заменяет на ее абсолютный адрес, как в двух предыдущих способах. Полученный при этом числовой операнд становится вторым или третьим словом команды, состоящей из двух или трех слов.

Ясно, что при этом способе часть команд, запись которых формально возможна, теряет смысл. Например, практически бесполезна такая команда, как CLR #1000. Что тут мы можем обнулить? Только второе слово самой команды, а зачем? Тем не менее такая команда существует и будет работать (если не находится в ПЗУ). Или, допустим, команда MOV #1000, #2000. Она всего лишь перешлет второе слово команды по адресу третьего слова. Трудно вообразить программу, в которой это требуется... Но пора, кажется, перейти к примерам более практического свойства. Пожалуйста:

MOV	#1000,R2	Запись в R2
:	:	числа 1000
MOVB	#40,R0	Запись в R0
:	:	числа 40

Обратите внимание, что во втором примере число 40 записываем в регистр R0 с помощью «байтовой» команды. Как это понимать, разве мы можем адресоваться не ко всему регистру, а к его части? Да, можем. Если при регистровой адресации применена байтовая команда, она оперирует с младшим байтом регистра. Раз так, то старший байт регистра при этом меняться не должен? Не тут-то было! Если с младшим байтом оперирует команда MOVБ, происходит не всегда полезная вещь, называемая распространением знака, — все биты старшего байта регистра принимают значение знакового разряда младшего байта (т. е. разряда 07). Например, в приведенном примере (MOVБ #40, R0) все разряды старшего байта R0 обнулятся, хоть бы мы и не преследовали

67

такой цели! А после команды MOVБ # 377, R0 в регистре R0 окажется не 377, а 177777 (во все разряды старшего байта R0 будут занесены единицы). В отличие от этого после команды MOV # 377, R0 в R0 окажется число 377, как и можно было ожидать. Распространение знака возникает, если число в младший байт регистра будет записано с применением не только этого, а и любого иного способа адресации и команды MOVБ. Это, видимо, единственный случай, когда младший байт слова оказывает какое-то влияние на старший. При байтовых операциях с ячейками ОЗУ такого явления, конечно, не бывает. Продолжим примеры. Нижеследующая запись уже в какой-то степени напоминает собой осмысленный текст программы.

```
CLR @#156      Очистка слова
;              по адресу 156
CLRB @#45      Очистка байта
;              по адресу 45
MOV @#100112,R2  Запись содержи-
;              жимого слова
;              по адресу
;              100112 в R2
MOV #137,@#1000  Запись числа
;              137 по адресу
;              1000
```

```
MOV #TEX, R2    Запись адреса
MOVБ #101, (R2)+ метки TEX в R2
MOVБ #102, (R2)+ и запись в пос-
MOVБ #103, (R2)+ ледовательные
;              ячейки памяти,
;              начиная с метки
;              TEX, чисел 101,
;              102 и 103 (ко-
;              дов символов А,
;              В, С)
```

В процессе программирования часто приходится с помощью непосредственной адресации засылать в регистры (реже — в ячейки памяти) коды символов. Для этого надо их помнить наизусть или по крайней мере все время иметь под рукой таблицу кодов. Различные версии ассемблеров (в том числе, конечно, и МИКРО.10К) предоставляют пользователю дополнительное удобство — возможность записи вместо кодов символов самих символов. Это тоже разновидность непосредственной адресации. Символы (они могут быть любыми) должны быть заключены в апострофы, и их число не должно быть больше двух.

```
MOV 'a', R0     Примеры записи в
MOV '12', R0    регистр R0 кодов
MOV 'A=', R0    символов а, 1 и 2,
;              А и =
```

Если в апострофах при такой адресации записан один символ, то его код заносится в младший байт регистра, если два — код первого заносится в младший байт, а второго — в старший.

**10. Абсолютная адресация.** Этот способ прост и прямолинеен: указывается абсолютный адрес ячейки памяти, к которой мы обращаемся. Записывается как @#X. Восьмеричное число X становится при трансляции вторым или третьим словом команды. Приведем примеры.

Данный способ широко применяется, особенно для обращения к ячейкам, адреса которых лежат вне модуля самой программы, например по адресам системной области или ПЗУ. Обращаться же к ячейкам, находящимся в ОЗУ пользователя или тем более в самой программе, обычно удобнее с помощью *относительной* адресации, которую рассмотрим далее.

**11. Относительная адресация.** Этот способ является, пожалуй, самым распространенным в двух случаях: во-первых, при задании адресов переходов и обращений к подпрограммам (что мы рассмотрим в свое время); во-вторых, при обращении к ячейкам памяти (переменным), расположенным в ОЗУ пользователя, особенно в пределах модуля самой программы. Записывается как МЕТ. При таком обращении ассемблер во время трансляции вычисляет *смещение*, т. е. разность между адресом текущей команды и адресом метки МЕТ, и записывает это смещение вторым или третьим словом команды. При исполнении программы адрес операнда вычисляется как сумма текущего содержимого счетчика команд РС и смещения. Смещение вычисляется в дополнительном коде, с учетом знака, поэтому такая адресация возможна как вперед, так и назад по отношению к адресу текущей команды. Не вдаваясь в эти тонкости, которые больше волнуют создателей ассемблер-систем, чем их пользователей, можно сказать, что относительная адресация — это прямое обращение к ячейке памяти, помеченной меткой МЕТ. Для ассемблеров, имеющих механизм локальных меток, необходимо иметь в виду, что адресация (во всех случаях, когда в составе операндов употребляется имя метки) возможна только с использованием обычных меток (начинающихся с буквы), а локальные метки пригодны лишь для указания адресов переходов в операторах ветвления и цикла. Это связано в основном с тем, что отличать локальную метку (начинающуюся с цифры) от числа (например, индекса) ассемблер не может. Приведем примеры.

```
CLR A1          Адресация с
MOV R2, BR2     использованием
MOV PCB, (R5)+  меток A1, BR2,
```

MOV #377, R7  
 MOV R7, R4 PCW, R7

Отметим особо, что метки, упоминаемые в операндах (при любом способе адресации с их использованием), должны быть *определены*, т. е. эти метки должны присутствовать в тексте программы, помечая собой какие-либо строки (либо должны быть определены оператором прямого присваивания, о котором речь пойдет дальше). При использовании имен не определенных (отсутствующих) меток правильная компоновка программы будет невозможна.

**12. Относительная косвенная адресация.** Аналогична предыдущему способу, но по адресу, помеченному меткой, находится не само число-операнд, а, как обычно в случаях косвенной адресации, его адрес. Записывается как @MET. Применяется весьма редко, например для передачи управления по адресу, который перед этим занесен в определенную ячейку памяти. Практически всегда этот способ адресации может быть заменен на другой, например регистровую косвенную адресацию.

На этом изучение способов адресации можно считать оконченным. В заключение приведем сводную таблицу всех способов. Но сначала сделаем замечание, на первый взгляд противоречащее всему вышесказанному. Так вот, теоретически возможных способов адресации всего восемь. В общем виде все способы так или иначе используют один из регистров общего назначения, т. е. являются разновидностями регистровой адресации. Но ведь мы рассмотрели не 8, а 12 способов, и в четырех последних имена регистров не фигурируют! Дело в том, что способы 9—12 являются не самостоятельными, а лишь разновидностями первых восьми. Они выделены потому, что имеют большое практическое значение, и по этой же причине в языке ассемблера для них введены особые, «нестандартные», обозначения. Между тем смысл этих способов сводится к тому, что они в качестве регистра-указателя просто-напросто используют регистр R7, или PC (счетчик команд). Проиллюстрируем это совершенно банальное, но звучащее не совсем обычно утверждение, примером. Как мы уже знаем, непосредственная адресация выглядит как #X, где X — число (для простоты не будем касаться меток). Возьмем конкретную команду:

MOV #1000, R2

При трансляции ассемблер заносит число 1000 вторым словом команды. Но мы уже сталкивались с примером, когда, применяя

регистр PC и автоинкрементную адресацию, можно было прочитать записанную после команды константу! Итак, непосредственная адресация — это никакой не новый способ, а все та же адресация с автоинкрементом PC. Вместо команды MOV #1000, R2 вполне можно просто записать

MOV (PC)+, R2  
 -#1000

(здесь #1000 — запись числа 1000 по текущему адресу).

Совершенно так же можно «развенчать» и остальные три способа — абсолютной, относительной и относительной косвенной адресации. Предоставляем это сделать читателю. Словом, это не способы адресации, как таковые, а просто удобные приемы программирования на языке ассемблера. Спо-

#### \* Основные способы адресации

Номер (код)	Наименование	Обозначение на ассемблере
0	Регистровая (прямая)	RN
1	Регистровая косвенная	@RN
2	Автоинкрементная косвенная	(RN)+
3	Автоинкрементная двойная косвенная	@(RN)+
4	Автодекрементная косвенная	-(RN)
5	Автодекрементная двойная косвенная	@-(RN)
6	Индексная	X (RN); MET (RN)
7	Индексная косвенная	@ X (RN); @ MET (RN)

#### Способы адресации с использованием PC, имеющие отдельные названия и обозначения на языке ассемблера

Номер (код)	Наименование	Обозначение на ассемблере	
		Стандартное	Специальное
2	Непосредственная	(PC)+	# X
3	Абсолютная	@(PC)+	@ #X
6	Относительная	MET(PC)	MET
7	Относительная косвенная	@MET(PC)	@MET

ру нет, транслятор при этом делает за нас немалую работу, например: вычисляет адреса меток, смещение, заносит эти данные по адресу нужного слова команды и т. д. И тем не менее, строго говоря, это все-таки не спо-

собы адресации, а просто их частные случаи с использованием РС. Так мы их и приведем в таблице.

Как видите, ничего принципиально нового в последних четырех способах адресации нет. Но их так называемая стандартная запись для программирующих на ассемблере выглядит в высшей степени необычно и почти никогда не применяется.

Можно ли изобрести другие способы с применением РС? Да, конечно. Но они практического значения не имеют и потому либо не применяются, либо мы их применяем, даже не задумываясь, что у них есть столь «именитые родичи», имеющие отдель-

ные «титуты» и «гербы». Продемонстрируем такие «открытия». Допустим, мы по аналогии захотели создать способ с применением РС, имеющий код 0. Что мы получим? Например: MOV PC,R4. Что же тут нового? Или, например, код 1: MOV PC,R4 — так мы просто запишем в R4 код следующей команды. Это нужно? Мягко говоря, не часто... Код 4: MOV a—(PC),—(PC) — страшный вирус, мы с ним уже знакомы. И наконец, код 5: MOV a—(PC),R4 переписет в R4 число из адреса, равного коду текущей команды, и заикнется! Чепуха, и ничего более.

*Продолжение следует*

## Квазиподпрограммы

Известно, что для многократного выполнения строго фиксированных операций программным образом лучше всего в тексте основной программы использовать набор соответствующих специализированных подпрограмм. Однако при этом возникает целый ряд вопросов и затруднений, связанных со способом хранения, модификации и передачи параметров и данных, необходимых для их работы. Оптимизировать программу, имеющую в своем составе большой набор таких специализированных подпрограмм, позволяет описываемый ниже метод. Чтобы сделать его максимально понятным и наглядным, рассмотрим в качестве примера решение следующей задачи:

Составить программу на Бейсике БК или другой ПЭВМ, которая на экране рисует изображение почтового конверта (рис. 1), ожидает нажатия любой клавиши, затем, очистив экран, «переворачивает» конверт (рис. 2) и при новом нажатии клавиши «открывает» его (рис. 3). Точки изображе-

15:(100, 180); 19:(135, 111); 4:(150, 200); 8:(140, 130); 12:(140, 160); 16:(140, 180); 20:(100, 50).

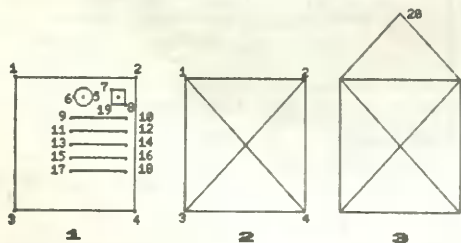
Для решения задачи удобно организовать и выполнить программным образом несколько стандартных операций. Обозначим каждую своей буквой: задать точку — Т; провести линию — Л; очертить рамку — Р; нарисовать окружность — О; закрасить изображение — З; ждать нажатия клавиши — Ж; очистить экран — Ч; закончить работу — К.

Если теперь все необходимые параметры и данные записать в одном или нескольких списках оператора DATA, а подпрограммы, нуждающиеся в них, снабдить операторами READ для считывания, то, очевидно, основная программа превратится просто в совокупность функциональных операторов операторов GOSUB и READ.

На этом этапе построения программы можно исключить из текста... все операторы GOSUB. Действительно, их функции могут взять на себя те буквы, которыми мы обозначили стандартные операции, и последовательности команд Бейсика, выполняющие эти операции.

Условные обозначения операций вместе с относящимися к ним параметрами назовем квазиподпрограммами. Таким образом, в нашем представлении каждая квазиподпрограмма имеет свое имя-идентификатор, однако в отличие от идентификаторов, допускаемых языком Бейсик, никаких ограничений на их количество и состав символов в них не налагается. Более того, их можно записывать без кавычек и при необходимости в нескольких списках оператора DATA.

Все операторы DATA со списками имен



ния имеют следующие координаты: 1:(50, 100); 5:(115, 120); 3:(100, 150); 13:(100, 170); 17:(100, 190); 2:(150, 100); 6:(105, 120); 10:(140, 150); 14:(140, 170); 18:(140, 190); 3:(50, 200); 7:(130, 110); 11:(100, 160);



квазиподпрограмм и необходимых данных разместим в конце текста программы и назовем блоком директив. Теперь останется составить фрагменты программы, которые будут непосредственно выполнять действия, заданные квазиподпрограммами (назовем совокупность этих фрагментов исполнительным блоком), и ту часть программы, которая будет вызывать эти фрагменты на исполнение (управляющий блок).

Итогом работы будет следующая программа.

```

10 DIM M(100,2)
20 '-----
30 ' УПРАВЛЯЮЩИЙ БЛОК
40 '-----
50 READ КЯ 'КЯ - ИМЯ ОЧЕРЕДНОЙ
      ПОДПРОГРАММЫ
60 IF КЯ="Т" THEN 160 'ЗАДАТЬ ТОЧКУ
70 IF КЯ="Л" THEN 230 'ЛИНИЯ
80 IF КЯ="Р" THEN 270 'РАМКА
90 IF КЯ="О" THEN 310 'КРУГ
100 IF КЯ="З" THEN 380 'ЗАКРАСКА
110 IF КЯ="Ж" THEN 420 'ЖДАТЬ КЛАВИШУ
120 IF КЯ="Ч" THEN 450 'ОЧИСТИТЬ ЭКРАН
130 IF КЯ="К" THEN 480 'КОНЕЦ РАБОТЫ
140 GOTO 50
150 '-----
160 ' ИСПОЛНИТЕЛЬНЫЙ БЛОК
170 '-----
180 '*** Т - ЗАДАТЬ ТОЧКУ *****
190 READ N,X,Y
200 M(N,1)=X
210 M(N,2)=Y
220 GOTO 50
230 '*** Л - ЛИНИЯ *****
240 READ N1,N2
250 LINE (M(N1,1),M(N1,2))-
      (M(N2,1),M(N2,2))
260 GOTO 50
270 '*** Р - РАМКА *****
280 READ N1,N2
290 LINE (M(N1,1),M(N1,2))-
      (M(N2,1),M(N2,2)),,B
300 GOTO 50
310 '*** О - КРУГ *****
320 READ N1,N2
330 DX=M(N1,1)-M(N2,1)
340 DY=M(N1,2)-M(N2,2)
350 R=SQR(DX*DX+DY*DY)
360 CIRCLE (M(N1,1),M(N1,2)),R
370 GOTO 50

```

```

380 '*** З - ЗАКРАСКА *****
390 READ N
400 PAINT (M(N,1),M(N,2))
410 GOTO 50
420 '*** Ж - ЖДАТЬ КЛАВИШУ *****
430 IF INKEY#="" THEN 430
440 GOTO 50
450 '*** Ч - ОЧИСТИТЬ ЭКРАН *****
460 CLS
470 GOTO 50
480 '*** К - ОКОНЧИТЬ РАБОТУ
490 END
500 '-----
510 'БЛОК ДИРЕКТИВ
520 '-----
530 DATA Ч,Т,1,50,100,Т,2,150,100,
      Т,3,50,200,Т,4,150,200,
      Т,5,115,120,Т,6,105,120,
      Т,7,130,110,Т,8,140,130,
      Т,9,100,150,Т,10,140,150,
      Т,11,100,160,Т,12,140,160,
      Т,13,100,170,Т,14,140,170,
      Т,15,100,180,Т,16,140,180,
540 DATA Т,17,100,190,Т,18,140,190,
      Т,19,135,111,Т,20,100,50,
      Р,1,4,0,5,6,Р,7,8,Л,9,10,
      Л,11,12,Л,13,14,Л,15,16,
      Л,17,18,З,19,Ж,4,Р,1,4,Л,1,4,
      Л,2,3,Ж,Л,1,20,Л,2,20,Ж
1000 DATA К

```

71

Преимущество данного метода в том, что он обеспечивает создание компактных, удобных в отладке программ, которые, в свою очередь, очень легко модифицируются: для создания новых изображений на экране достаточно изменить или расширить компактный блок директив.

Предлагаемый метод пригоден не только для графических программ: в исполнительный блок могут включаться команды, выполняющие самые разнообразные действия, а для управления ходом их выполнения можно пользоваться оператором ON.

Лучше всего этот метод оправдывает себя при написании программ линейной структуры, позволяя в 5—10 раз увеличить их объем за счет компактности хранения данных и жесткой структуры.

А. ИНГОРЬ

## Спасение программ при зависании

При использовании кодов 149—151 в программах, написанных на Фокале, БК часто зависает. Приходится перезапускать процессор, теряя содержимое ОЗУ. Однако зачистку можно избежать стирания памяти! Для этого достаточно, перезапуская процессор, держать нажатой клавишу **СТОП**. Если после перезапуска появляется отклик

в виде вопросительного знака, то программа спасена.

Этот прием основан на том, что после запуска процессора управление передается на адрес 100000 («горячий» запуск монитора), а первая расположенная здесь системная команда — **ЕМТ 14**; она заносит по вектору прерывания клавиши **СТОП** «холодный»



небольших отдельных файлов на МЛ;

ИПС обеспечивает автоматическое считывание с МЛ только тех файлов, которые относятся к выбранному ИБ, остальные файлы игнорируются.

Работа с программой ИПС.01К состоит из двух частей: подготовки ИБ на МЛ и непосредственного поиска нужной информации по модели, заданной пользователем.

При подготовке ИБ данные оформляются в виде таблицы с задаваемыми пользователем графами. Это могут быть разнообразные каталоги, ведомости, сведения об учащихся и т. п. Каждая строка таблицы должна представлять собой совокупность жестко связанных логических параметров, по каждому из которых потом можно будет провести поиск всех остальных, сцепленных с ним. Подготовленные таким образом файлы ИБ записываются на МЛ с учетом того обстоятельства, что ИПС.01К способна вести поиск не только в самом содержимом ИБ, но и по именам составляющих его файлов. В связи с этим необходимо присваивать файлам информативные имена и уже на этом этапе

закладывать возможность ускорения поиска в дальнейшем.

Процедура поиска проста: достаточно задать одну из девяти основных моделей поиска, после чего по одному или по совокупности нескольких параметров из ИБ будет выбрана вся относящаяся к ним информация. Затем она может быть распечатана на принтере или подвергнута дальнейшей обработке (дополнительной выборке, редактированию и т. п.).

Особенностью ИПС.01К является заложенная в ней возможность поиска информации не только по текстовым моделям, но и по числовым параметрам. Например, из списка людей с указанием дат их рождений можно выбрать родившихся до определенного года или после него.

Хотя ИПС.01К создавалась для работы на изолированном компьютере, оснащенном лишь магнитофоном и принтером, она легко может быть адаптирована для работы в сети КУВТ. В этом случае удобство и оперативность работы с ней возрастает на порядок.

С. КУМАНДИН

## Полку периферии пришло

Ю. А. Зальцман разработал аппаратно-программный комплекс, позволяющий вводить в память БК-0010 телевизионные изображения. Время ввода одного кадра с телевизионной камеры типа «Электроника-841» — 1,2 с. После доработки возможен ввод изображения непосредственно с эфира (через телевизор) или видеомагнитофона. Вводимый кадр отображается на экране дисплея.

Аппаратная часть выполнена на 19 микросхемах серии К155 и позволяет оцифровывать телекадр с двумя или тремя уровнями яркости, получать позитивное, негативное или оконтуренное изображение размером  $512 \times 256$  точек.

Программная часть помимо управления вводом пикселей в экранное ОЗУ обеспечивает упаковку изображения и запись его на МЛ в формате графического редактора ГРЕД.М1 для дальнейшей обработки (либо воспроизведения на экране без графического редактора).

Общий вес конструкции (без телекамеры) — 2 кг, питание от сети 220 В, потребляемая мощность менее 20 Вт. Телевизионный интерфейс можно использовать для ко-

пирования графических материалов (электронная фотография), в качестве системы технического зрения (распознавание образов,



автоматизированный контроль, ориентация и сортировка объектов, зрение роботов), для создания заставок и картин к играм, компьютерной мультипликации и т. п.

Ю. КУЗЬМИН, И. РИБУЛЕ, С. КОНДРАТЕНКО  
Латвийский государственный университет

## Использование ЭВМ при обучении русскому языку

Появление ЭВМ в школе делает возможным развитие компьютерной технологии обучения различным предметам. Какова целесообразность этого направления при обучении русскому языку?

Известно, что существует ряд проблем изучения русского языка, решение которых находится в стадии становления. Так, например, в [1] отмечается необходимость интенсификации процесса обучения. ЭВМ может сыграть здесь существенную роль. Особенно это относится к регионам, где компьютеризация школ достигла уровня, при котором возможен доступ учеников к ЭВМ не только на занятиях по информатике. Одним из таких регионов является Латвия, оснастившая компьютерами к 1989 г. свыше 60 % средних школ.

Интенсификация обучения русскому языку с помощью ЭВМ возможна прежде всего за счет автоматизации работы учителя во время урока, например при использовании программ-тестов знаний грамматики русского языка. В этом случае экономится время на выявление ошибок учеников, повышается оперативность реакции учителя на эти ошибки, появляется возможность обобщения ошибок за счет анализа результатов тестирования с помощью ЭВМ.

Автоматизация работы учителя возможна и с помощью программ-репетиторов, которые в форме диалога объясняют особенности грамматики русского языка. В отличие от традиционного урока занятия по таким программам ведутся индивидуально, исключена возможность пропуска информации по невнимательности ученика. Впервые появляется возможность совместить процесс подачи материала ученику с процессом его усвоения за счет оперативного контроля.

Еще одна форма автоматизации — применение ЭВМ для подготовки заданий ученикам путем программированного выбора упреждений. В мировой практике имеется также опыт грамматического анализа изложений и сочинений учеников.

Наконец, интенсификация учебного процесса достигается и в результате использования программ-тренажеров и программированных диалогов с ЭВМ на разнообразных темах, что развивает не только грамматические навыки учеников, но и их лексический запас. Это особенно важно для национальных школ, где русский язык является вторым изучаемым языком.

Следует отметить, что программированное обучение русскому языку представляет интерес и для зарубежных стран, применяющих компьютерную технологию обучения.

Остановимся кратко на некоторых особенностях компьютерной технологии обучения языкам. Прежде всего отметим текстовую форму диалога ученика с ЭВМ. Это ограничение исключает развитие речевого общения ученика. Следует, однако, отметить быстрый прогресс в области говорящих ЭВМ, что дает перспективу автоматизации обучения языку и на уровне речевого диалога.

С теоретической точки зрения все, что выразимо словами, можно ввести в ЭВМ в виде текста. В этом смысле возможности учебных книг и ЭВМ совпадают. Однако есть и существенные различия.

Прежде всего структура книжного текста всегда последовательна. Структура текста в ЭВМ обычно намного сложнее, поскольку доза компьютерного текста (здесь она называется дисплейным кадром — около 20 строк по 32—80 букв в каждой) зависит от ответов ученика на вопрос в этой дозе. При

верном ответе обычно выдается новый объем информации, а при ошибке может последовать разъяснение. Большое число ошибок иногда ведет к понижению уровня сложности текста и наоборот: малое число ошибок — к повышению. Конечно, все это должно быть заранее запрограммировано. Из этой особенности компьютерных текстов вытекает много других. Например, объем информации, получаемой учеником, зависит от качества его ответов. Отсюда же вытекает и то, что компьютерный учебник нельзя «пролистать», как книгу, не вникая в ее содержание, так как на каждый вопрос нужно ответить, что можно сделать, только поняв текст.

Существенной особенностью компьютерной информации является возможность «оживления» иллюстраций. ЭВМ по программе может нарисовать на экране дисплея рисунок. Изменив параметры программы, можно изменить и сам рисунок. Иногда это существенно, так как дает возможность показывать действия, связанные с изучаемыми глаголами. Другое применение — «оживление» картинок, показывающих технику произношения звуков русских слоговых сочетаний, что окажет помощь изучающим русский язык либо имеющим дефекты речи. В этом случае ученик вводит слоги или даже слова, а ЭВМ показывает на экране, как они произносятся. Сравнивая диалог ученика с ЭВМ или учителем, нужно отметить ряд особенностей.

Во-первых, учитель всегда ведет диалог экспромтом, а ЭВМ должна быть заранее запрограммирована. Это значит, что нужно заранее предусмотреть не только тему диалога и сюжет, но и возможные ответы ученика. Диалог с ЭВМ будет тем универсальнее и гибче, чем большее множество ответов предусмотрено программистом. Отсюда следует, что программистом может быть лишь опытный учитель, который знает типичные ошибки учеников, ибо все возможные ошибки предусмотреть невозможно из-за малой памяти ЭВМ, больших временных затрат и сложности программирования. Кроме того, только опытный учитель знает, какие вопросы следует задать, чтобы выяснить понимание учеником заданной темы.

Следующая особенность компьютерного диалога — однонаправленность. Если в беседе с учителем инициатива диалога может меняться, то в случае с ЭВМ ученик всегда играет пассивную роль, отвечая на ее вопросы.

Наконец, важная особенность диалога с ЭВМ — точность. Он не может строиться по принципу «Расскажи о...». Это связано с ограниченными возможностями разбора ответа ученика и с отсутствием запрограммиро-

ванных структур знаний о внешнем мире, а также с невозможностью проведения семантического анализа ответов ученика.

Наиболее близким к имитации диалога учителя с учеником пока что следует считать вариант конструкции, основанной на разборе ответа ученика (на вопрос ЭВМ нужно вводить слова ответа, а ЭВМ проверит правильность написания этих слов путем их сравнения с заданными заранее). Однако, чтобы это реализовать, программист должен знать, какие слова и в каком порядке искать в ответе учеников. Конечно, такие знания есть только у опытного педагога. Особенно это касается программирования разбора возможных ошибок ученика. Каждый вопрос должен точно предусматривать слова (или их группы), которые ожидаются в ответе ученика. Этого можно достичь разными приемами, например предложить заполнить в вопросе пропущенные буквы:

«Введите слова, в которых необходим мягкий знак:

Девоч?ка попросила у мал?чика мяч?». Правильный ответ, ожидаемый ЭВМ: «мальчик». Отдельно следует предусмотреть ошибочные слова: «девочка, мяч».

При обучении грамматике русского языка особая проблема — контроль знания правил и определений. Например, на вопрос «Что такое прилагательное?» возможен ответ: «Это часть речи, которая выражает признак предмета, изменяется по родам, числам и падежам, в предложении является определением или именной частью составного сказуемого» [2].

При анализе подобных ответов возможны следующие подходы.

Строгий подход. Если учитель настаивает на заданном определении, то в шаблоне ЭВМ должны присутствовать все слова и в том порядке, как они приведены выше. Сравнение ответа с шаблоном должно быть полным. ЭВМ прекрасно справится с этой задачей! И в то же время мало педагогов согласится с таким подходом, поскольку он означает механическое заучивание.

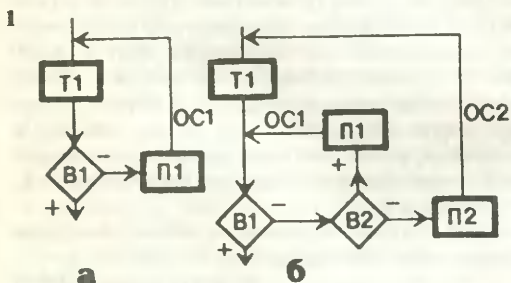
Аналитический подход. В данном случае ЭВМ ищет самые значимые слова определения, игнорируя их возможные вариации. Но набор значимых слов, конечно, зависит от ситуации и от точки зрения учителя. Например, в указанном примере значимыми словами могут быть: «часть речи», «признак предмета».

Прагматический подход. При нем вместо вопросов на знание определений даются предложения (наборы слов), содержащие элементы, правильное выделение (изменение, дополнение) которых возможно лишь в том случае, если известно проверяемое определение.

Итак, умение точно и полно сформулировать вопрос и предусмотреть возможные ответы (и ошибки) ученика — одно из важных требований к учителю русского языка, желающему применять ЭВМ.

Другим требованием является осведомленность учителя об известных структурах обучающих программ, что позволяет ему выбрать наиболее пригодную структуру. В обучающей системе «Рига», которая использовалась авторами при создании программных материалов по русскому языку, опробованы следующие структуры:

1. «Репетитор». Данную структуру можно применять для передачи элементов теоретических знаний. Схематично один из ее компонентов изображен на рис. 1, а. Здесь блок Т1 — кадр с теоретической информацией (правило, определение, общая информация и т. п.); В1 — вопрос, проверяющий понимание теоретического материала; П1 — пояснение при отрицательном (—) результате проверки; плюсом помечен переход к аналогичному кадру с теоретической информацией при положительном результате проверки понимания Т1; ОС1 — обратная связь, т. е. переход после пояснения к повтору Т1.



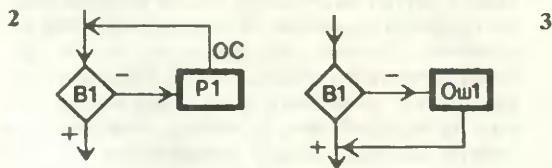
Подобная структура чревата закливанием, т. е. бесконечным повторением ОС1. Такая ситуация, немислимая при общении учителя с учеником, вполне возможна в компьютерных вариантах. Причиной закливания может быть, например, отсутствие в Т1 и П1 информации для ответа. Другой частный случай — ошибка в способе анализа компьютером ответа ученика. Рекомендуется приводить в П1 варианты правильного ответа с необходимыми объяснениями.

Структура элемента «репетитора» может быть и более сложной. На рис. 1, б показан вариант, в котором после основного вопроса В1 следует при неверном ответе вопрос В2, который, например, уточняет причину ошибочного ответа на вопрос В1, П1 — вспомогательная информация для ответа на вопрос В1, П2 — пояснения для ответа.

Следует отметить, что чем сложнее структура репетитора, тем труднее его запро-

граммировать, так как необходимо проверить все комбинации возможных связей.

2. Обучающий тест (рис. 2). В обучающем тесте задается вопрос В1 и, если получен верный ответ, на экране появляется следующий вопрос. Если ответ неверен, выдается рекомендуемый ответ Р1, и вопрос задается до тех пор, пока правильный ответ не будет



заучен. В конце обучающего теста можно попасть, только ответив на все вопросы.

3. Экзаменующий тест (рис. 3). Его цель — проверка знания ответов на заданный комплекс вопросов и запоминание всех ошибок. Отличается от обучающего теста тем, что при неверном ответе на вопрос В1 фиксируется ошибка Ош1. В конце теста сообщаются все неверные ответы, например в виде перечня ошибок Ош1, Ош2, ..., ОшN либо текстов с указанием того, что именно не знает ученик. Например: «Вы не знаете падежных окончаний существительных». Если есть способ вычисления оценки, то она также может быть выдана автоматически.

Особая проблема, возникающая в тестах, связана с тем, какие именно ошибки нужно в них фиксировать. Мы полагаем, что если учителю важно знать качество усвоения изучаемого правила, то фиксировать надо ошибки, связанные именно с ним, ибо в противном случае придется выполнять сложный анализ по отделению данной ошибки от всех возможных, не связанных с изучаемым правилом. Как следствие, при таком подходе нужно будет считаться с побочными эффектами. Например, для контроля знания правила написания мягкого знака в словах *зонтик, хищник, письмо* достаточно указать ключи *нт, щн, сьм*. Но тогда ответы типа: *зонтек, хьщник, песьмо* ЭВМ признает верными! И это будет правильным с точки зрения указанных ключей. Задание в качестве ключей полных слов вызывает проблему локализации ошибки. Например, в данном неверном ответе ошибки не были связаны с незнанием правила употребления мягкого знака.

4. Тренажер. Этот вид программ позволяет в результате многократного повторения некоторых грамматических факторов, подлежащих запоминанию, добиться их полного усвоения. При этом возможно отображение гистограммы ошибок по каждому

факту, что указывает на наиболее трудные из них. Тренажеры могут быть на различные темы, например на правописание слов с разделительными знаками *ь* и *ъ*.

При создании перечисленных программ возникает закономерный вопрос о принципах их написания. Чаще всего таковые отсутствуют, и в результате появляются программы, отвечающие методике работы конкретного учителя. Они не образуют системы, и без присутствия автора их трудно использовать.

В данной работе осуществляется следующий подход. На первом этапе предлагаются комплексные экзаменующие тесты, позволяющие определить основные ошибки учеников по курсу русского языка. Таких тестов создано три. Вопросы тестов охватывают разделы программы «Морфология и орфография». В качестве исходного материала использовано пособие для занятия по русскому языку [3]. Тесты написаны на Т-языке и могут быть использованы на ЭВМ БК-0010 либо в системе «Рига» для КУВТ-86 [4].

На втором этапе, после определения ошибок, может быть предложена индивидуальная работа с обучающимися тестами на недостаточно усвоенные темы. Авторами создано 49 обучающих тестов на Т-языке. Результатом работы могут быть уточнение основных ошибок учеников и самостоятельная проработка ими трудных грамматических тем. Список обучающих тестов приведен ниже.

#### Список обучающих тестов

Непроверяемые орфограммы  
 Правописание приставок *пре-* и *при-*  
 Непроизносимые согласные  
 Удвоенные согласные  
 Правописание приставок  
 Правописание сложных слов  
 Приставки на *-з*  
 Чередование гласных в корнях *лож — лаг* и *кос — кас*  
 Разделительный мягкий и твердый знаки  
 Гласные *ы* и *и* после приставок  
 Чередование гласных в корнях слов  
 Чередование гласных *а — о* в корнях слов  
 Чередование гласных *е — и* в корнях слов  
 Род существительных  
*-Н-* и *-нн-* в существительных  
 Окончания существительных (единственное число)  
 Окончания существительных (множественное число)  
 Суффиксы *-чик-* и *-щик-* у существительных  
 Частица *не* с существительными  
 Падежные окончания прилагательных  
 Суффиксы прилагательных  
 Суффиксы прилагательных *-к-* и *-ск-*  
*-Н-* и *-нн-* в суффиксах прилагательных  
 Краткая форма прилагательных

Сравнительная форма прилагательных

Сложные прилагательные

Правописание числительных

Употребление местоимений

Отрицательные местоимения

Неопределенные местоимения

Спряжение глаголов (настоящее время)

Повелительное наклонение глагола

Суффиксы глаголов *-ыва-* и *-ева-*

Правописание причастий

Суффиксы причастий

Краткие и полные причастия

*-Н-* и *-нн-* в прилагательных и причастиях

Употребление деепричастий

Правописание наречий

Окончания наречий

Правописание *-н-* и *-нн-* в различных частях речи

Употребление мягкого знака в различных частях речи

Правописание предлогов

Правописание союзов

Употребление частицы *не*

Употребление частицы *ни*

Однородные члены

Причастные и деепричастные обороты

Сложноподчиненные предложения

Каждый учитель может самостоятельно расширить набор тестов, поскольку все они выполнены по универсальной схеме и не требуют знания программирования. Структура теста показана на рис. 4.

Пять строк для вопроса	маршрутн <input type="checkbox"/>	такси;
	трамвайн <input type="checkbox"/>	депо;

Две строки рекомендуемого ответа	маршрутное, трамвайное
--	------------------------

Одна строка ключей	тное йное
-----------------------	-----------

При работе ЭВМ представляет ученику задание в форме, представленной на рис. 5. Если ответ верный, то ЭВМ выдаст следующий вопрос; в случае ошибки ученик получит кадр такого типа, как изображенный на рис. 6. После него ЭВМ вновь покажет кадр, изображенный на рис. 5.

Введите определения, вставляя пропущенные окончания:	
маршрутн <input type="checkbox"/>	такси;
трамвайн <input type="checkbox"/>	депо;

6

Вы ответили:  
маршрутная, трамвайный  
а нужно:  
маршрутное, трамвайное  
Сравните Ваш ответ с рекомендуемым  
и нажмите ВВОД

В конце теста на экране ученика появится кадр, изображенный на рис. 7.

7

Число вопросов=14  
Число ошибок=2  
Ошибки обозначены 00001

V=00000	C=00001	D=00001
E=00000	F=00000	G=00000
H=00000	I=00000	J=00000
K=00000	L=00000	M=00000
N=00000	O=00000	

Позовите преподавателя

78

В разработанном пакете тестов соблюдены некоторые условия, позволяющие сделать перевод тестов на другие языки народов СССР. В первую очередь это касается лексики, отобранной так, чтобы не вызывать особых трудностей для понимания. Кроме того, в тестах используются общеупотребительные слова и обороты, которые способствуют совершенствованию лексического запаса учащихся. На рис. 8 приведен пример вопроса с переводом на латышский язык.

8

Введите слова, раскрывая скобки:

(авиа) почта, (юго) запад  
(aviopasts, dienvidrietumi)

На третьем этапе производится обобщение результатов тестирования. Система «Рига» позволяет автоматизировать этот процесс. В результате учителю становятся известными наиболее трудные темы по русскому языку. На основе этих данных можно проводить занятия в классе традиционным способом либо с использованием «репетиторов» и тренажеров по наиболее трудным темам. Эти занятия можно рекомендовать как факультативы для наиболее слабых учеников.

Кроме того, результаты статистического анализа трудности отдельных вопросов могут быть использованы при построении интегральных тестов, программ-репетиторов, программ-тренажеров и т. п. для задания порядка следования грамматических явлений

согласно их трудности для учащихся.

Ориентиром очередности создания разнообразных программ для ЭВМ по русскому языку может быть частота повторения грамматических явлений русского языка в лексике. На рис. 9 приведен результат анализа первой тысячи наиболее употребительных слов с точки зрения наличия в них того или иного грамматического явления. Из него видно, что прежде всего обращают на себя внимание первые десять грамматических явлений, а в них — первые пять (это правописание непроверяемых гласных и согласных, безударных проверяемых гласных

9



в корне, букв е/и в окончаниях существительных, безударных гласных в окончаниях прилагательных и букв е/и в окончаниях глаголов I и II спряжения).

Следует отметить, что при создании компьютерных программ по русскому языку нужно учитывать особенности восприятия русского языка учащимися разных национальных групп. Иными словами, программное обеспечение русского языка для различных национальных групп должно различаться.



1. Текучев А. В. Методика русского языка в средней школе. М., 1980.
2. Баранов М. Т. и др. Русский язык: Учебник для 5—6 классов. М., 1974. С. 97.

3. Греков В. Ф., Крючков С. Е., Чешко Л. А. Пособие для занятий по русскому языку. М., 1981.

4. Кузьмин Ю., Гвардина И., Кузьмина Л. Система «Рига» // Информатика и образование. 1988. № 2.

Г. ГВАРАМИЯ, И. МАРГВЕЛАШВИЛИ, Л. МОСИАШВИЛИ

Главный центр информатики Министерства народного образования ГССР

## Опыт разработки компьютерных учебных пособий по физике

При всем многообразии мнений, по существу, прослеживаются два подхода к определению места и функций компьютеров в обучении. Одни считают компьютер техническим средством, позволяющим освободить учителя от однообразной, рутинной, трудоемкой части педагогической деятельности, не способным помочь в решении творческих задач.

Другие считают, что с помощью компьютерного обучения можно формировать качественно новые знания, развивать аналитическое и творческое мышление учащихся.

Многолетний опыт по разработке и внедрению компьютерных учебных программ в общеобразовательные предметы нашего авторского коллектива Главного центра информатики Министерства народного образования Грузинской ССР дает основание считать, что истина, скорее всего, лежит посередине. В процессе нашей работы над созданием систематических учебных курсов по разным предметам математического, естественного и гуманитарного циклов вырисовывались основные принципы создания компьютерных учебных программ, разрабатывалась методика компьютерного обучения по общеобразовательным предметам. И сегодня мы твердо уверены в том, что недооценка компьютера как средства обучения в решении творческих педагогических задач вытекает вследствие неумения разработчиков учебных программ в полной мере реализовать имеющиеся возможности. Ясно, что легче написать простенькую по своей логике и функциональным связям контролирующую программу или тренажер, нежели создать на экране дисплея операционную среду, позволяющую учащемуся управлять моделью изучаемого процесса, а за-

тем исследовать и анализировать результаты своего воздействия на него.

К систематическому компьютерному курсу, являющемуся, по существу, компьютерным учебным пособием по общеобразовательному предмету, помимо общих требований, таких, как научность, системность, доступность и т. д., нами были сформулированы специфические требования: включение моделирования в содержание, индивидуализация обучения на основе обратной связи, повышение мотивации и наглядности обучения, совмещение контроля с самоконтролем, сбор статистики.

Ниже пойдет речь об одном из разработанных нами автоматизированных учебных пособий по физике, которое уже несколько лет применяется в школе. К сегодняшнему дню анализ результатов его использования в учебном процессе в основном проведен.

Автоматизированное учебное пособие по физике для VIII класса (механика) состоит из тематического и контрольно-игрового курсов.

Цель тематического курса — помочь учащемуся в качественном усвоении содержания основных физических понятий, в выработке умений и навыков применения теоретических знаний для решения конкретных учебных и практических задач. Курс состоит из одиннадцати программ, соответствующих основным темам учебного плана по физике. В зависимости от дидактических функций они делятся на контролирующие, моделирующие, демонстрационные, тренирующие и т. д. Но это деление условно, поскольку в одной и той же программе могут присутствовать контроль и обучение или демонстрация и моделирование, обучение и тренаж и т. д. Тип программы указы-

вает лишь на то, какой функции уделяется главная роль. Общими для всех тематических программ являются разветвленная структура учебного диалога, учитывающая типичные ошибки учащихся и содержащая кадры помощи; случайный выбор численных значений физических величин из банка данных; автоматизированный контроль по критерию, учитывающему степень сложности заданий.

Ниже приводится краткое описание некоторых учебных программ, реализованных на «Ямахе».

Программа «Общие сведения о механическом движении» содержит обучающий блок с вербальной информацией и соответствующим иллюстративным материалом. Она призвана закрепить знание общих понятий курса механики. Обучающие кадры чередуются с оперативными, содержащими три задания, в процессе выполнения которых учащиеся включаются в активный диалог с программой. Первое задание состоит в упорядочении таблицы, содержащей в одном столбце различные физические явления, а в другом — соответствующие им, расположенные в случайном порядке физические величины. Второе содержит вопросы из истории физики и имеет форму кроссворда. Третье задание — на определение положения тела в различных системах отсчета.

Программа «Относительность движения» представляет собой лабораторную работу, в которой важное место занимает модель изучаемого движения. Она состоит из двух задач и контрольных вопросов.

Первая задача соответствует случаю, когда движение тела и подвижной системы отсчета параллельно: наблюдая движение лодки вниз и вверх по течению реки, учащийся по значениям измеряемых величин (скорость лодки, время ее движения в обоих направлениях) вычисляет значение скорости движения. Вторая задача соответствует случаю, когда движение тела и подвижной системы отсчета взаимноперпендикулярны: по измеренным значениям векторов скорости лодки относительно берега и скорости течения реки учащийся должен выбрать курс для кратчайшей переправы на противоположный берег.

С помощью контрольных вопросов проверяется осознанность действий учащегося.

Следует отметить, что в этой программе, как и в других, содержащих управляемые модели, демонстрируются состояния изучаемой системы в соответствии с введенным ответом учащегося, как правильным, так и неправильным. Это позволяет учащемуся наглядно видеть, выполняются или нет условия задачи, получено или нет правильное

решение, при этом у него вырабатывается навык самопроверки.

Программа «Средняя скорость неравномерного движения» напоминает учащемуся содержание понятия средней скорости; в ней приводятся модели равномерного и трех неравномерных движений со средней скоростью, равной, больше или меньше скорости равномерного движения.

Программа включает две задачи. В первой учащийся строит модель неравномерного движения тела, вычисляет среднюю скорость на отдельных участках и на всем пути. Во второй задаче моделируется ситуация, в которой в зависимости от значения вычисленной учащимся средней скорости путник приходит к отплытию теплохода либо вовремя, либо рано, либо с опозданием. Показ модели, соответствующей ответу учащегося, дает ему возможность оценить правильность решения задачи.

Программа «Равноускоренное движение» состоит из нескольких заданий на определение значений физических величин для разных случаев равноускоренного движения в зависимости от направления векторов начальной скорости и ускорения друг относительно друга. Программа развивает умение представлять демонстрируемое моделью движение в разной форме: аналитической, табличной, графической.

Программа «Искусственный спутник Земли» состоит из демонстрационного блока, задачи и контрольной работы. В демонстрационной части показывается, как изменяется траектория движений искусственного спутника Земли от скорости запуска. Задача на расчет значения первой космической скорости и периода обращения спутника на выбранной высоте. После ее решения учащийся наблюдает на экране запуск спутника. Программа завершается контрольной работой, которая оценивается по пятибалльной системе.

В программе «Движение тела под действием силы тяжести» необходимо сбросить с самолета груз для мальчика, спасшегося от шторма. Учащийся должен выбрать скорость и высоту полета самолета и рассчитать координаты точки, из которой должен падать груз, чтобы попасть на остров. При желании учащийся получает необходимую помощь.

Программа «Сила трения» состоит из двух частей: в первой осуществляются контроль и коррекция теоретических знаний; во второй учащемуся предлагается задача — остановить автомобиль у перекрестка с учетом тормозного пути, зависящего от конкретных условий движения.

Педагогические цели создания контрольно-игрового курса физики — развить у учащихся

ся навыки устного решения несложных физических задач, закрепить и систематизировать знания. Очень важным мы считаем развитие у учащихся положительной мотивации и интереса к изучению физики.

Игра между учащимся и компьютером проводится в три раунда. В каждом раунде учащийся, самостоятельно решивший задачу, получает один балл, при правильном решении после получения помощи балл делится пополам между учащимся и компьютером, при повторном неправильном решении балл получает машина. Задания случайным образом выбираются из банка, в который заложены задачи по всем основным разделам и темам школьного курса механики (банк заданий открыт и может пополняться). Каждое задание начинается демонстрацией модели рассматриваемого в задаче явления или ситуации, затем выводится текст задачи и меню ответов, из которых учащийся должен выбрать правильный. В описанном тематическом курсе физики в основном применялся способ «свободного конструирования» ответа, который ближе к естественному диалогу учащегося с преподавателем (учебной программой) и поэтому является методически оправданным, несмотря на трудности автоматического распознавания правильного ответа.

В основе применяемой нами методики разработки программного обеспечения лежит проектирование программ «сверху вниз» и составление соответствующего алгоритма; написание программ «снизу вверх» с логическими комментариями и проверкой правильности на каждом уровне. Это дает возможность в процессе апробации с учащимися вносить в программу коррективы, что для

учебных курсов не только желательно, но и необходимо.

Мы привели краткое описание разработанного компьютерного учебного пособия по школьному курсу физики для того, чтобы показать читателю наш подход к созданию обучающих программ, важным принципом которого является ориентация на развитие мышления, активности и творчества учащихся. Совершенно ясно, что компьютерное обучение никогда не сможет заменить общение с мыслящим и увлеченным своей работой учителем, но этого и не требуется. Проводимые нами в течение ряда лет исследования показали, что у ребят, систематически работающих с компьютерными учебными программами по общеобразовательным предметам, отмечается умение рационально строить свои действия, однозначно ставить цель, четко формулировать решения, правильно оценивать свои знания и возможности.

Использование компьютера как средства обучения общеобразовательным предметам играет важную роль в приобщении будущего специалиста к средствам вычислительной техники, в овладении современными способами организации и обработки информации. То, что уже во многих школах республики при изучении математики, физики, русского и английского языков используются учебные пособия в виде комплектов, состоящих из записанных на магнитные диски компьютерных учебных курсов и подготовленных с помощью компьютера методических пособий по их использованию, помогает школьникам осознавать, что компьютеры — это уже не завтрашний, а сегодняшний день.

## **ВНИМАНИЮ ЧИТАТЕЛЕЙ**

**Новый адрес редакции «ИНФО»:**

**Лефортовский пер., 8.**

**Тел. 261-11-29**

**Почту просим направлять по адресу:  
119034, Москва, Смоленский бульвар, 4,  
издательство «Педагогика»,  
журнал «Информатика и образование».**

В настоящее время одна из наиболее сложных проблем — научное обоснование психолого-педагогической целесообразности использования компьютеров в обучении детей.

В НИИ общей педагогики АПН СССР на паритетных началах между АН СССР и АПН СССР создана лаборатория экспериментальных исследований процесса формирования личности школьника в условиях НТП. Одно из направлений работы лаборатории получило название «Компьютерный эксперимент». Базой для проведения научно-исследовательских работ стала СШ № 1158 Москвы. В этой статье мы хотим познакомить читателей с организацией и результатами первой части эксперимента, проведенного в 1989/90 учебном году.

Нас интересовало, какой образ компьютера складывается в сознании ребенка по мере его знакомства с этим обучающим средством; каковы преобладающие мотивы обучения детей в условиях компьютеризации; какие факторы наиболее сильно воздействуют на динамику поведения детей на уроках; при каких условиях целесообразно использовать ЭВМ на уроках гуманитарного и математического цикла; что меняется в деятельности учителя и в его поведении во время компьютерных уроков. Но больше всего нас всех интересовали три проблемы: получают ли учащиеся, общаясь с компьютером, новые возможности для своего развития? Изменяется ли роль учителя во время компьютерных уроков, наконец, какую роль в процессе компьютеризации обучения играет программное обеспечение?

Выбор объектов наблюдения был, конечно, не случаен: он обусловлен единой «точкой отсчета», позицией лаборатории в многоплановой проблеме компьютеризации образования, получившей название «Гуманитарная концепция компьютеризации учебно-воспитательного процесса школы». В концепции подчеркивается, что участники эксперимента стоят на позиции самоценности личности каждого ученика и полагают, что применение компьютеров в обучении создаст предпосылки для изменения учебной среды, внося вариативность, разнообразие информации, доступ к ней помимо учителя; что использование компьютеров значительно усилит «обратную связь» от результата деятельности учащихся; что изменится ведущая функция педагога — он выступит как организатор самостоятельной учебной деятельности детей. В концепции отмечается также, что использование компьютера обеспечит принципиаль-

но новые возможности самопознания ребенка, его самореализации. Участники эксперимента предполагают, что использование компьютера в обучении обеспечит развитие прежде всего таких качеств учащихся, как познавательная самостоятельность, активность, повысит компетентность школьника как ученика. Все эти идеи в определенной степени проверялись во время эксперимента.

Наш опыт показал: проведение компьютерных экспериментов возможно при условии интереса, поддержки и активного участия педагогического коллектива школы. В документе, разработанном в конце 1987 г. школьной администрацией и научными сотрудниками лаборатории и получившем название «Концепция развития школы № 1158 на 10 лет», отмечалось, в частности, что расширение сферы использования компьютеров в учебно-воспитательном процессе рассматривается как одно из приоритетных направлений развития школы, как одно из направлений ее экспериментальной работы.

Условно организацию и проведение эксперимента можно разделить на четыре этапа. **I этап — июнь — декабрь 1989 г.**

За это время была разработана программа компьютерных уроков; совместно с дирекцией школы выбран экспериментальный класс (VII E), проведены совещания с педагогами, которые предложили провести компьютерные уроки в феврале — апреле 1990 г. Одновременно в МГПИ им. В. И. Ленина была сформирована группа студентов физического факультета, которые под руководством канд. физ.-мат. наук, доцента кафедры математической физики Е. К. Годуновой взялись за разработку программ по алгебре, геометрии, истории, географии, русскому, английскому, французскому языкам.

**II этап — январь — февраль 1990 г.**

В этот период в дисплейном классе школы, оборудованном компьютерами «Правец-8», проходили апробация разработанных программ и обсуждение их с учителями школы.

Пионерами компьютерного обучения стали учитель математики и директор школы Г. И. Метлик, истории — О. Ю. Фаттах, географии — П. В. Дунаев, русского языка — Г. П. Машьянова, английского языка — С. Л. Кононова, французского — Л. В. Авкина.

В это же время каждым исследователем были отобраны небольшие «референтные» группы, за которыми ученые вели наблюдения во время уроков. В лаборатории под руководством психолога М. С. Жамкочян были раз-

работаны основные показатели замеров психолого-педагогической эффективности экспериментальных уроков, обсуждены вопросы анкет, опросников, определено место каждого исследователя в общей работе. Было решено, что главное внимание участников эксперимента будет сосредоточено на изучении деятельности детей и учителей во время уроков с использованием компьютеров, на новых педагогических ситуациях, которые возникают на этих уроках.

### III этап — 20 февраля — 16 апреля 1990 г.

За это время в дисплейном классе было проведено двадцать уроков с использованием компьютеров, организованы дополнительные занятия, проведены разнообразные опросы, наблюдения, тестирование, беседы как со всем классом, так и с референтными группами детей, интервью с учителями и школьниками.

### IV этап — 17 апреля — июнь 1990 г.

Это этап анализа эксперимента и обсуждения его результатов. 17 апреля в школе прошла итоговая конференция с участием преподавателей ряда факультетов МГПИ им. В. И. Ленина.

В соответствии с логикой исследования мы хотим остановиться на результатах, которые объединили в небольшие разделы: «Дети и компьютер», «Компьютерный урок», «Учитель на компьютерном уроке», «Компьютер диктует перемены».

## Дети и компьютер

То, что дети жаждут участвовать в компьютерном эксперименте, было видно «невооруженным глазом», но мы всё же спросили их об этом. Ответ был однозначный. «Да», — ответили 100 % учеников VII Е (29 человек).

В эксперименте нас интересовало, каковы причины столь восторженного отношения детей к компьютеру. Исследование показало, что причин несколько.

До того как дети первый раз сели за компьютер, в их сознании (это показали опросы, интервью) уже существует стойкое представление об ЭВМ, вызывающее желание сесть за персональный компьютер и работать. Причем это желание настолько устойчиво, что не меняется даже после неудачного опыта общения. Это подтвердили опросы: отмечая, что во время компьютерных уроков были слишком трудные задания (80 %), неинтересные программы (50 %), плохо работала техника (32 %), слишком примитивные задания (28 %) и т. д., 64 % учащихся тем не менее высоко оценили уроки, а 96 % выразили желание участвовать в подобном эксперименте в будущем году.

Положительное отношение детей к ком-

пьютерам, высокий уровень мотивации к работе на ЭВМ до ее начала объясняется экспериментальным характером учебного процесса с использованием компьютеров. Здесь находит отражение потребность ребенка в новизне. Активно-положительное отношение детей к компьютерам связано также в значительной степени с соображениями престижа. Среди ответов школьников на предложенные вопросы, были такие, как: «Я работаю на компьютере», «Я угадал», «Я сумел», «Я ответил».

В сознании детей образ компьютера складывается постепенно. Он ассоциируется с чем-то добрым, хорошим, справедливым. Справедливость связывается в первую очередь с оценкой: «Если он говорит тебе, что ты не знаешь, значит, и впрямь не знаешь».

«Говорит», «думает», «его не обманешь», «отвечает» — такие слова употребляют многие дети, говоря о компьютере. В ответах некоторых ребят компьютер описывается как инструмент, машина. «Компьютер — инструмент. Его придумали люди, чтобы он решал сложные задачи, которые человек не может решить»; «Компьютер работает только под руководством человека»; «Это инструмент, который запоминает программы. Память его беспредельна — человек не может столько помнить»; «Без человека компьютер не может думать», но с компьютером человек думает быстрее».

Дети и компьютер глазами психолога. Во время уроков было важно понять, каковы состояние, активность, настроение детей при работе с компьютером? Как изменяется активность школьника при изменении заданий? На эти вопросы дало ответы исследование М. С. Жамкочьян.

Из дневника эксперимента. В работе использовались психологические методики: САН — методика, которая измеряет три вида состояний: самочувствие, активность, настроение; биполярные шкалы. Активность также мерялась по количеству попыток в разных заданиях. Продуктивность — по количеству правильно решенных задач, по количеству баллов, набранных в заданиях, и оценке, выставляемой компьютером.

Кроме того, мы наблюдали за эмоциональными индикаторами у части детей, фиксировали вспышки радости, возмущения, депрессию. Замеры состояний проводились до и после уроков с компьютерными программами, а также на контрольных уроках истории и математики без использования ЭВМ. Всего по психологическим методикам было проведено 16 замеров. Получено 480 показателей. Что показало наше исследование?

1. Независимо от предмета, все показатели САН улучшались к концу уроков, по срав-

нению с началом (так, например, на уроке геометрии «настроение» сдвинулось на 0,5 балла, а «самочувствие» и «активность» — на 1 балл). Надо заметить, что замеры в начале урока проводились на фоне повышенного интереса и настроения, так как ожидалось занятия на компьютере, т. е. не было страха опроса, ожидания скучного урока и т. д.

Данная методика обеспечивает хороший разброс данных: при отмеченной тенденции к росту активности и других состояний у отдельных индивидов встречаются противоположные сдвиги — у одной испытуемой сдвиг по всем показателям отрицательный:  $S = -5$ ,  $A = -4$ ,  $H = -7$ , т. е. эмоциональное состояние в целом снижается. Такой случай единственный. Еще у троих испытуемых снижается либо настроение (1 случай), либо активность (2 случая). Эти же испытуемые на других уроках (история) обнаруживают ту же тенденцию. Снижение эмоционального состояния может быть ситуативно, т. е. зависеть от каких-либо конкретных состояний, а может быть личностной характеристикой (слабость нервной системы, быстрое эмоциональное насыщение, бедность эмоциональной среды). У двоих испытуемых аналогичная тенденция — к снижению обнаруживается и на уроке истории, хотя им удалось пройти программу 2 раза.

Обнаружились «лидеры» положительного сдвига. У четверых испытуемых на уроке геометрии подъем показателей вышел за 1,5–2 балла, а у одного — выше 4. Это означает, что у него до урока эмоциональное состояние выражается отрицательными баллами. Здесь можно говорить об ожидании негативных событий, об отрицательной установке. Затем в течение работы с компьютером знак эмоций меняется на противоположный. То же самое характерно для урока истории. Соответствуют ли такие самооценки реальному поведению? Мы выбрали учащегося П. для наблюдения по изложенной выше программе. Что мы увидели? Во-первых, он приходил и приступал к занятиям первый, а остановить его в конце удавалось с трудом. Все диалоги с машиной велись вслух, никаких отвлечений. Неизменно выполнял больше заданий, чем другие, не отвлекался от компьютера и просил дополнительные задания. Наблюдалась и ярко выраженная эмоциональная экспрессия (жестикуляция: разговор с самим собой, возбужденная речь и др.). Поведение было одинаково на всех компьютерных уроках, независимо от содержания и характера программ. Не изменялось отношение и от количества ошибок, и от выставаемой оценки.

В то же время представители «отрицательных сдвигов» демонстрируют заторможен-

ность в поведении, уклонение от контактов, попытки отказа от заполнения методик. Если встречались затруднения в решении заданий или отказы машины, у таких испытуемых наблюдалось пассивное ожидание. После выполнения необходимого минимума работа заканчивалась, даже если оставалось время до конца работы. Очевидно, в таком поведении проявлялись как отношение к учебным занятиям, так и личностные особенности.

2. Вторая часть результатов касалась условий эффективности работы с ЭВМ. Было установлено, что наибольшие отрицательные эмоциональные реакции вызывало прекращение и прерывание деятельности. Если хотя бы одно задание было выполнено полностью, а остальные не удавалось выполнить по разным причинам (не хватало времени, сбой, программы), то незначительно ухудшалось настроение, активность не снижалась.

Если же перерыв или сбой наступал в середине выполнения заданий, наступало состояние фрустрации (состояние стресса при невозможности достижения цели). Для него характерны внутренняя напряженность и отрицательные эмоции (возмущение, недовольство, депрессия). В психологии существует феномен «прерванное действие», открытый и исследованный в школе немецкого психолога К. Левина. Он заключается в следующем: если предложим ребенку выполнить какое-то задание, скажем решить арифметическую задачу, а затем прервать его, не дав закончить, и предложить заняться чем-нибудь другим, внутреннее напряжение будет настолько велико, что даже по прошествии времени условие задачи вплоть до цифр хорошо сохранится в памяти, тогда как дети, дорешавшие задачу до конца, впоследствии не могут вспомнить ни цифр, ни самого условия задачи. Психологи предположили, что возникает напряженность потребности, потребности завершить деятельность конечным результатом. Причем здесь речь шла о заданиях, лишенных какой-либо привлекательности. В этом смысле деятельность сама требует необходимого завершения, она обладает целостностью и самоценностью.

Эмоциональные сдвиги типа фрустрации обнаруживались у наших испытуемых и по результатам психологических исследований, и по наблюдениям преподавателей. Вот один пример: после сбоя и 10 бесполезных попыток довести хотя бы одно задание до конца испытуемая Д. в сердцах отодвигает лист с методикой, сообщает, что ей плохо, а затем все-таки отвечает, обводя только позицию 1 по каждому состоянию. По сравнению с состоянием фона, больше чем на 2 балла ухудшились и настроение, и активность, и самочувствие. Возросла лишь напряженность.

Очевидно, что этот факт имеет существенное значение для педагогической технологии обучения с использованием ЭВМ: необходимо предусматривать для каждого ребенка возможность ведения каждого задания до конца, независимо от индивидуального темпа. Возникает и другой вопрос: сохранится ли урок в его традиционной форме?

### Компьютерные уроки

На уроках школьники работали с программами по истории: «Было или не было?» (игра с ЭВМ); историческими кроссвордами, игрой «Индия и Китай в средние века», «Причины Великих географических открытий»; по географии: «Определение основных климатических показателей» (тренажер); по русскому языку: «ЭВМ угадывает задуманное слово» (дидактическая игра), игра «Четвертый лишний»; по французскому языку: «Словарная работа по теме «Спорт»; по английскому языку: тренажер «Английские слова и фразеологизмы»; по математике: «Многочлены стандартного вида, округление десятичных дробей» (тренажер), «Квадрат суммы и разности двучлена», «Параллельные прямые и треугольники (задачи)»; «Прямоугольные треугольники» (задачи); «56 задач на доказательство с одним рисунком», «Прямая обратная и противоположная теоремы».

«Дорогой друг! — начиналась анкета, обращенная к ученикам в последний день эксперимента. — Ты принял участие в 20 компьютерных уроках. Каковы твои впечатления о них?»

Анкета предлагала 11 серий вопросов. Приведем ответы детей не некоторые из них.

На вопрос о том, было ли интересно участвовать в компьютерных уроках, 64 % учащихся ответили «да», 46 % — «не на всех» и «не очень». Ответы показали, что школьникам запомнились уроки истории (96 %), русского языка (56 %), алгебры и французского языка (44 %), английского языка (32 %), геометрии (28 %). Больше всего в компьютерных уроках им понравилось (оценка давалась в баллах): работать с компьютером (4,6 балла), работать с компьютером вместе с товарищами (4 балла), то, что была удача, — «У меня всё получилось очень хорошо» (3,1 балла). На вопрос, стали ли уроки с использованием компьютера событием в их жизни, ответили «да» 48 % учащихся. Объясняя свой ответ, дети писали: «Компьютерные уроки дали новые знания о том, о чем я раньше знал только понаслышке» (56 %); «Все узнали, какой я умный» (16 %), «Я узнал, какой я умный, сообразительный» (68 %).

Исследование, включающее сопоставительный анализ нескольких пар уроков, прово-

димых с использованием компьютера и без него (уроки одного и того же педагога), проводилось по пяти параметрам. Фиксировались:

методы учителя, побуждающие ученика к действию;

реакция ученика на побуждение; педагогические санкции (поощрение, наказание, оценка, другие реакции); диалог (наличие, характер); конфликт (наличие, характер).

На основании этих наблюдений мы сделали некоторые предварительные выводы.

1. Длительная предварительная подготовка учителя совместно с программистами к уроку позволяет тщательно продумать методику включения всех учеников в деятельность, поэтапное оценивание этой деятельности, диалог ученика с учителем через компьютер.

2. Отличительная особенность компьютерного урока — деловая и очень спокойная атмосфера, общение детей друг с другом и с учителем на деловой (предметной) основе.

3. Функции оценивания и контроля («церберовские» функции учителя) переданы компьютеру — отсюда бесконфликтный характер обучения.

Компьютерные уроки продолжались, и каждый день приносил новые факты, которые нам предстояло объяснить. Так в дневнике эксперимента проявилась страничка «Парадоксы компьютерных уроков».

1. Обычная картина: звенит звонок, и наши дети, бросив все дела, мчатся на перемену. В конце компьютерного урока этого не происходит. Напротив, детей невозможно упросить закончить работу и уйти из кабинета информатики. Когда, наконец, потеряв всякое терпение, мы их попросту выгоняем, они уходят рассердившись, обидевшись смертельно...

2. На уроках без компьютеров, которых мы во время экспериментов посетили множество, мы видели, как дети, легко отвлекаются от работы, болтают о посторонних вещах. Во время компьютерных уроков дети от начала урока и до самого его конца работают много и упорно. Все разговоры посвящаются делу.

3. На обычном уроке ребенок, как правило, очень болезненно реагирует на плохую отметку. На компьютерных уроках реакция иная. Почему? Отвечая на этот вопрос, дети объясняют, что отметка поставлена справедливо, заслуженно, не то, что во время обычного урока, когда учитель ставит «2» неизвестно за что...

4. На обычном уроке дети часто (слишком часто!) с каждой мелочью обращаются к учителю. На уроках с ЭВМ они от помощи часто отказывались, ни к кому не обращались.

Стремление их к самостоятельности было очевидным и росло. Это позже показало и исследование: если работу с компьютером они оценили (по 5-и балльной системе) в 4,6 балла, то работу за ЭВМ под руководством учителя в 2,3 балла.

### Учитель на компьютерном уроке

«Каким увидел ты своего учителя во время компьютерного урока? Отличалось ли его поведение на этом уроке от обычного? Если «да», то чем?» — такой вопрос задали мы ученикам. Анализ ответов позволил нам составить коллективный портрет учителя. Итак, учитель на компьютерном уроке работал иначе, чем обычно, по мнению 91 %. В чем же главное отличие? Учитель был более спокоен (64 %), гораздо веселее, чем обычно (76 %); больше времени уделял общению с детьми — (беседовал, объяснял, подходил) (80 %); ни разу ни на кого не сердился (100 %). Суммируя сказанное школьниками, получаем, что на компьютерных уроках они увидели спокойных, молчаливых, но веселых, охотно помогающих и дружелюбных и ни разу ни на кого не рассердившихся учителей.

Поясняя свои ответы, дети писали, что больше всего их поразило то, что учитель, который на обычных уроках не закрывает рта (согласимся, что наши уроки, как правило, сплошной монолог), мало говорил: «Говорил намного меньше, и это хорошо», «Говорили они намного меньше, а вот помогли намного больше».

Не раз приходилось слышать, что компьютеры рождают «молчаливое поколение», что японцы, обнаружив это, скоро вовсе откажутся от использования компьютеров в обучении... Мы этого пока подтвердить не можем. Но мы увидели иное — появляется поколение учителей, которые умолкают на время, чтобы дать возможность детям учиться самостоятельно. Не знаем, достаточно ли это для надежды, что эти педагоги со временем примут диалог как главный метод работы, но у нас она появилась.

### Компьютер диктует перемены

В процессе эксперимента был проведен психолого-педагогический анализ преобладающих мотивов обучения в их динамике. Была разработана специальная анкета «Твой друг компьютер». Ей был придан не только исследовательский, но и развивающий характер, так как опрос должен был стимулировать фантазию учащихся, а также ориентировать их на осмысление собственных мотивов «компьютерного поведения». В анкете были вопросы, которые выявили предпочтения детей в области использования ЭВМ, а также позво-

ляли реконструировать систему ценностей подростков по отношению к окружающему миру. Чтобы проследить динамику изменений в ходе интенсивного использования компьютеров, анкетирование было проведено дважды — до и после эксперимента.

Результаты первого замера оказались достаточно неожиданными, в частности опровергли распространенное мнение о том, что современные дети теряют интерес к учебе. Подавляющее большинство опрошенных высказались за увеличение продолжительности уроков информатики. 17 из 26 учеников пожелали, чтобы время работы на ЭВМ вообще не ограничивалось. При этом игровая мотивация даже не являлась ведущей, а выступала наряду с мотивами самосовершенствования и развития. Вполне уместно было бы задать себе вопрос: действительно ли наши дети не хотят учиться или у них нет возможности учиться тому, чему хочется.

Оказалось также, что работа с ЭВМ вовсе не является просто престижной ценностью для подростков. По своей важности умение пользоваться компьютером в иерархии ценностей учащихся VII E стоит лишь на четвертом месте. Следует отметить, что вообще все так называемые престижные моменты оказались в конце системы ценностей. Имеются в виду, конечно, усредненные результаты. Но в целом наиболее важное значение в жизни современного человека, по мнению детей, будет занимать борьба за сохранение окружающей среды и умение разбираться в экологических вопросах.

Второй замер показал, что мотивация работы с ЭВМ у детей изменилась в ходе обучения в режиме «погружения», но не совсем так, как можно было предположить. В результате знакомства с информатикой и обучающими программами заметно ослабела игровая ориентация (со второго места по значимости отодвинулась на четвертое). Но интерес к ЭВМ тем не менее не упал, а вырос. Основной претензией стала обида на нехватку машин в компьютерном классе, из-за чего приходится работать по двое или становиться в очередь. Вырос интерес учащихся к диалогу с ЭВМ и к «самопознанию с помощью компьютера». Дети открыли для себя новые возможности работы с ЭВМ, не менее увлекательные, чем игра.

Вместе с тем следует отметить, что период первоначального восторга сменился более критическим отношением как к курсу информатики, так и к себе. Ребята стали замечать больше недостатков в организации учебного процесса. Им хотелось бы, чтобы появились в классе цветные мониторы и принтеры, чтобы более надежным стало техническое обслуживание машин. Более критично стали оце-



ниваться собственные возможности и навыки. Если в первый раз на вопрос «Сможешь ли ты передать записку приятелю через компьютер?» большинство ответили утвердительно, то во втором замере были не столь категоричны.

Важный момент: отвечая на вопрос о месте компьютеров в их будущей профессиональной деятельности, только единицы связывают свои мечты с конструированием ЭВМ и созданием программ и совсем никто не проявил желания иметь дело с игровыми автоматами. Большинство же подростков видят себя пользователями ЭВМ, применяющими компьютеры для работы над проблемами из других областей науки и практики. И тенденция эта усиливается во втором замере. В результате обучения складывается четко «пользовательская» познавательная ориентация учащихся на инструментальное применение ЭВМ. Но это именно то, чего требует от современного человека развитие новейшей технологии и культуры. Следует подчеркнуть, что наиболее привлекательным для подростков способом использования ЭВМ в школе (в первом и особенно во втором замерах) оказалось не создание зала игровых автоматов, а организация банка данных по всем предметам для свободного пользования. Это в значительной степени опровергает опасение, что именно игра привлекает детей к компьютеру и что она может вытеснить учебу. Работа с ЭВМ на уроке не снижает, а повышает заинтересованность детей в учебе, делает ее более осмысленной. Отвечая на вопрос «Чем уже сегодня помогает тебе информатика в учебе?», ребята отвечают, что учиться стало интереснее. Можно сказать, что в результате интенсивного использования ЭВМ развлекательная мотивация уступает место познавательной и «пользовательской».

Интересно, что обучение в режиме «погружения» отразилось на общей структуре ценностей школьников. Хотя первая и последняя позиции остались почти неизменными как некие константы, в целом система стала более «деловой» и, если можно так выразиться, социально направленной. На 2-е место по значимости вышел профессионализм в работе, а на 4-е —

участие в политической жизни. Таким образом, социальные ценности заняли ведущее место в иерархии. Вместе с тем умение работать с компьютером как самоценность отодвинулась с 4-го на 6-е место. Можно сделать вывод, что оптимально организованный процесс компьютеризации в школе повышает социальную включенность учащихся, их адаптивность к жизни в окружающем мире, понимание сути изменений, происходящих вокруг.

В конце работы мы попросили ребят из VII Е ответить на вопрос, зачем, по их мнению, проводился компьютерный эксперимент. Анализ ответов позволил выделить 6 «ключевых понятий», которыми дети обозначили цели проведенного исследования. По мере убывания они располагаются так: «интерес» (30%), «лучше» (30%), «узнать» (25%), «будущее» (15%), «учитель» (10%). Итак, дети считают, что эксперимент проводился для того, чтобы узнать, интересно ли школьникам работать с ЭВМ на уроках, и чтобы ребятам на уроках стало интереснее; чтобы лучше учить детей и чтобы все дети лучше поняли школьную программу, тему уроков, материал конкретного урока; чтобы учителя лучше узнали своих учеников и чтобы ученики лучше узнали себя; чтобы проверить обучающие программы для будущих учеников и в будущем ввести компьютерные уроки во все школы СССР; чтобы облегчить работу учителя. «Сейчас всё зависит от нас, — нимало не смущаясь написали наши дети, — понравились ли нам уроки — другие школьники тоже будут так заниматься. А уж нет — так нет!» Но компьютерные уроки детям понравились!

Желающие более подробно познакомиться с ходом эксперимента, а он продолжается, могут обращаться по адресу: 129278, Москва, ул. П. Корчагина, 7, НИИ ОП АПН СССР, комната 50, Демакова Ирина Дмитриевна. Телефон: 283-86-58.

И. ДЕМАКОВА, Е. ГОДУНОВА,  
В. БОРИСОВА, М. ЖАМКОЧЬЯН,  
Т. МАТВЕЕВА, М. ОЛЕЙНИК,  
И. ПЕТРОВА, Е. ПРОНИНА,  
В. ПЯТУНИН

**Молодежный центр  
интернационального сотрудничества  
«Квант»**

**предлагает  
программное обеспечение для БК-0010.**

**Наш девиз — лучшее качество при низких ценах.  
Наши цены в 5—10 раз ниже государственных.  
У нас вы можете заказать самые новые игровые,  
системные, обучающие и другие программы,  
схемы дополнительных устройств, расширения памяти.  
Мы разработаем программы по вашему заказу.  
Центр заключает договоры с авторами на разработку  
и тиражирование программ.  
Программы записываются как на кассеты заказчика,  
так и на кассеты центра.  
Стоимость одной программы от 25 копеек до рубля.  
Каталоги высылаются бесплатно.  
Наш адрес: 142403 Московская обл., Ногинск,  
ул. Краснослободская, 16. МЦИС «Квант».**

**Каталог BASIC программ  
для КУВТ «Корвет»**

Если вы желаете систематизировать имеющиеся у вас программы на языке программирования BASIC, «Каталог BASIC программ» окажет вам неоценимую помощь. Ваша программная библиотека будет иметь компактный вид, и вы получите возможность максимально быстрого и упрощенного доступа к любой нужной вам программе.

**«Каталог BASIC программ» — это комплект, состоящий из дискетки и описания работы с каталогом.**

Стоимость дискетки с каталогом и описанием — 120 рублей.

Заказ на «Каталог BASIC программ» можно сделать по адресу: **384002, г. Кутаиси, ул. Св. Нино, 9, Кутаисский Дом науки и техники.**

**Молодежный компьютерный центр  
«Контакт»**

предлагает пользователям персональных ЭВМ  
IBM PC, «Sinclair», «Commodore-64», «Atari», «Ямаха», Радио-  
86PK, «Микроша», «Криста»,  
«Партнер», «Львов», «Специалист», «Вектор»,  
БК-0010/0010. 01/0011, УКНЦ, «Корвет», ДВК-2/3  
широкий выбор программ и документации.  
Наш адрес: 614010, Пермь, а/я 4694.

## Модемы бывают разные

А какие именно? Приведем несколько основных стандартов на эти устройства.

Стандарт V.21 устанавливает скорость передачи данных 300 бит/с в полностью дуплексном режиме (последнее означает, что модем способен одновременно передавать и принимать данные, так же как и человек при обычном разговоре по телефону). Этот стандарт поддерживается почти всеми модемами и используется при обмене данными по низкокачественным телефонным линиям.

Стандарт V.22 задает полностью дуплексный режим и скорость передачи данных 1200 бит/с. На его примере хорошо виден смысл единицы *bod* — числа модуляций в секунду. Модемы стандарта V.22 для передачи данных используют звуковые сигналы четырех частот, каждая из которых соответствует одной из четырех возможных комбинаций пары битов; одной частотой кодируется пара 00, второй — 01, третьей — 10, четвертой — 11. Таким образом, передавая в секунду 600 звуковых сигналов, модем V.22 передает 1200 бит/с и работает со скоростью 600 бод, т. е. 600 модуляций (смен частот) в секунду.

В том же режиме, но со скоростью 2400 бит/с работают модемы стандарта V.22bis. Они обеспечивают достаточно правильную передачу данных по хорошим телефонным линиям, если же линия подвержена помехам, требуют использования средств исправления ошибок, которые пока еще дороги.

## Блокнот для заметок

Эволюция персональных компьютеров заставляет придумывать для них все новые и новые названия, которые отражали бы не только размеры новинки, но и ее функциональные особенности. Японская фирма Seiko Epson, например, предлагает своим покупателям персональный компьютер под названием «блокнот для заметок» (PC 286 NOTE.F). Од-

# ЧТО? МОЖЕТ ЭВМ

нако этот 32-разрядный компьютер ничего общего, кроме размеров, с блокнотом не имеет. Судите сами: 1,25-мегабайтная память на интегральной схеме, жидкокристаллический дисплей, 3,5-дюймовый дисконд и даже цена — 1320 долларов — не позволяют называть этот довольно мощный компьютер пренебрежительным словом «блокнот».

## Мир на коленях

Бум, связанный с появлением миниатюрных магнитофонов, получивших название «прогуливающийся мужчина» (Walkman), кажется, поутих. Одновременно уменьшилось и число несчастных случаев, связанных с тем, что наушники плеера изолировали гуляющего не только от городского шума, но и от предупреждающих сигналов автомобилей.

Теперь мужчины могут прогуливаться с видеоплеером — японские инженеры сконструировали Video Walkman — портативный видеоманитофон, который удобно смотреть, положив его на колени (lap-top). Взяв с собой в поездку видеоплеер и несколько кассет, вы можете прямо в дороге записать интересующую вас телевизионную программу, просмотреть старые записи или фильм. Невыясненным пока остается, как новый прибор скажется на безопасности владельцев.

## Холодно, теплее, горячо!

Эти слова вполне мог бы произнести микропроцессор, встроенный в прибор американской фирмы Molelectron Detector Inc. Однако он не предназначен для детской игры, поэтому разработчики решили не усложнять компактное устройство и ограничились жидкокристаллическим дисплеем. Помимо отображения цифровых результатов измерения

прибор может представить их и в виде графиков.

Новый измеритель сконструирован для контроля распределенной в пространстве энергии и удобен для настройки импульсных лазеров. С его помощью можно с высокой точностью измерять энергии от одного микроджоуля до джоуля, энергию единичных импульсов, следующих с частотой до 500 в секунду, среднюю энергию десяти импульсов и общее количество энергии, падающей на поверхность.

## И на дуде игрец

Японская фирма «Канон» выпустила в продажу портативную персональную станцию NAVIGT. От персонального компьютера, даже самого хорошего, рабочая станция отличается более профессиональной и разнообразной периферией, позволяющей максимально использовать возможности компьютера. Например, станция NAVIGT стоимостью менее 2 тыс. долларов комплектуется сканером GT/S B4 стоимостью более 2,6 тыс. долларов, плоским дисплеем с разрешающей способностью 640×400 точек, высококачественным принтером, мышью, модемом и даже телефоном.

Новая станция может выполнять функции по крайней мере шести устройств: персонального компьютера, программируемого телефона, телефакса, языкового процессора, ориентированного на японский язык, электронного калькулятора и часов. И можно быть уверенным, что всю эту работу станция проделает быстро и с высоким качеством. Еще бы: она оборудована центральным процессором 80286, работающим на частоте 10 МГц, имеет оперативную память объемом 1,25 и постоянную объемом 4М байт. Долговременная информация может храниться на 3,5-дюймовых дисках.

Кроме функций, предусмотренных разработчиками, владелец станции имеет возможность ориентировать ее на выполнение иных. Для этого на станции предусмотрены стандартный интерфейс RS-232C, параллельные и модульные разъемы и дополнительные гнезда для плат расширительных модулей и шины.

И. ПЕРЕХОД, В. КАСАТКИН

## Комбинаторные задачи

### Общий метод генерации комбинаторных объектов

Комбинаторика как раздел математики рассматривает задачу подсчета количества элементов в различных конечных множествах.

Несколько примеров. Пусть дано конечное множество  $A = \{a_1, a_2, \dots, a_n\}$  и  $n$  мест, на которых можно переставлять элементы данного множества, размещая их по одному в каждой позиции. Спрашивается: сколько различных перестановок можно осуществить?

Пусть из данного множества  $A$  выбираются группы, каждая по  $K$  элементов. Сколько таких сочетаний из  $n$  элементов по  $K$  можно образовать?

Пусть из данного множества  $A$  выбираются группы по  $K$  элементов, затем в каждой такой группе осуществляются все возможные размещения элементов по  $K$  местам. Требуется вычислить, какое число таких размещений возможно.

Далее каждое соединение (подмножество), полученное из множества  $A$ , будем называть комбинаторным объектом.

В классической комбинаторике рассматриваются и другие комбинаторные объекты, подсчитывается и их количество. О них речь пойдет в дальнейшем.

В комбинаторике разработаны методы вычислений. Приведем формулы, не обращаясь к их выводу, для вычисления количества перестановок из  $n$  элементов ( $P_n$ ), числа сочетаний из  $n$  элементов по  $m$  ( $C_n^m$ ) и количества размещений из  $n$  элементов по  $m$  ( $A_n^m$ ).

$$P_n = n!$$

$$C_n^m = \frac{n!}{m!(n-m)!}$$

$$A_n^m = \frac{n!}{(n-m)!}$$

Классическая комбинаторика, уделяя главное внимание подсчету количества комбинаторных объектов, оставляла в тени задачу их формирования. Дело в том, что даже при небольших  $n$  число перестановок, сочетаний, размещений и иных комбинаторных элементов велико и сформировать, записать, каждый из комбинаторных объектов — задача непростая.

Например, уже при  $n=7$  число перестановок элементов составляет 5040, записать все их вручную невозможно. В то же время потребность в получении конкретных комбинаторных объектов вытекает из практики.

Ниже рассматривается единый подход к формированию комбинаторных объектов, основанный на использовании так называемых цифровых систем.

Понятие *цифровая система* лежит в основе многих теоретических выводов и прикладных алгоритмов. Введем его, сопоставляя с понятием *позиционная система счисления*.

Напомним формулу целого неотрицательного числа в позиционной системе степенного типа с основанием  $q$ :

$$(a_n a_{n-1} \dots a_1 a_0)_q = a_n q^n + a_{n-1} q^{n-1} + \dots + a_1 q + a_0. \quad (1)$$

Здесь  $a_i$  есть цифра  $i$ -го разряда, для которой выполняется условие  $0 \leq a_i \leq q-1$ . (2)

Диапазон изменения цифр задан верхней и нижней границами, границы каждого разряда одинаковы. Верхняя граница  $\sup(i)$ , нижняя —  $\inf(i)$ ,  $0 \leq i \leq n$ .

Известны позиционные смешанные системы счисления, отличающиеся от систем степенного типа тем, что их базис не является геометрической последовательностью.

Примером может быть так называемая факториальная система с базисом 1, 2, 6, 24, 120, 720, ..., или 1!, 2!, 3!, 4!, 5!, 6!, ...,  $n!$  ( $32211_\Phi = 3 \cdot 5! + 2 \cdot 4! + 2 \cdot 3! + 1 \cdot 2! + 1 \cdot 1!$ ).

Нетрудно понять, что диапазон изменения

цифр в каждом разряде факториальной системы неодинаков. Нижние границы всех разрядов постоянны и равны нулю:  $\inf(i) = 0$ . Верхние границы также постоянны, но их значения есть функция от номера разряда:  $\sup(i) = i!$ .

Существуют смешанные позиционные системы с различными базисами и правилами вычисления нижних и верхних границ. В частности, нижние границы могут отличаться от нуля. Вообще, границы могут быть константами, переменными, зависеть от номера разряда  $i$ , количества разрядов  $n$  и значений  $a_i$ . В последнем случае зависимость может быть рекурсивной.

Опираясь на сказанное выше, дадим конструктивное определение цифровой системы. Цифровая система задается:

- 1) разрядностью  $n$ ;
- 2) формулами нижних и верхних границ каждого разряда  $i$   $\sup(i)$  и  $\inf(i)$ ;
- 3) операцией следования, позволяющей получить очередной цифровой объект системы по предыдущему.

Например, для  $n=3$ ,  $0 \leq i \leq n$ ,  $\inf(i) = 0$ ,  $\sup(i) = i$  можно построить записи 3110, 2201, 1210 и др. Полученные записи следует считать в рамках цифровой системы кортежами (наборами, составленными из цифр), а не числами, записанными в некоторой системе счисления.

Данное выше определение цифровой си-

стемы позволяет получать объекты, построенные из символов произвольного упорядоченного алфавита. Например,  $n=4$ , алфавит  $A = \{a, b, c, d, e, f\}$ ,  $\inf(i) = \langle a \rangle$ ,  $\sup(0) = \langle b \rangle$ , ...,  $\sup(4) = \langle f \rangle$ . Объектами такой системы будут  $\langle ecbba \rangle$ ,  $\langle fedcb \rangle$ ,  $\langle aaaaa \rangle$ .

Итак, в рамках конкретной цифровой системы можно конструировать специфические объекты в форме  $n$ -значных кортежей. Элементы, образующие кортеж, — это элементы множества, на котором задан порядок следования. В дальнейшем будем использовать традиционные способы упорядочения, принятые для чисел в математике, и лексикографический порядок, свойственный алфавитам языков человеческого общения и языков программирования.

Основным преобразованием объекта цифровой системы является операция следования, которая для каждого объекта позволяет построить следующий за ним в лексикографическом порядке (порядок задается над объектом в целом).

Выполняя операцию следования в цикле, получаем счетчик (генератор) объектов цифровой системы.

Далее будут приведены и обсуждены понятие и алгоритм обобщенного счетчика. Объекты конкретных систем генерируются с его помощью посредством задания нижних и верхних границ и разрядности  $n$ .

О. БЕЛОНОЖКО, А. ЮДИЛЕВИЧ  
г. Орел

## Психологическая диагностика детей и компьютер

Один из перспективных путей применения ЭВМ в дефектологии — разработка компьютерных методов диагностики отклонений в психическом развитии ребенка, его обучаемости. Разработанная в результате совместной работы программистов, педагогов-дефектологов и психологов система компьютерной психолого-педагогической диагностики позволяет, во-первых, максимально унифицировать процедуру и содержание обследования для получения сравнимых корректных результатов; во-вторых, экономит время и, в-третьих, делает возможной оперативную количественную обработку результатов обследования в сравнении с данными, полученными при контрольном компьютерном

тестировании репрезентативной выборки нормально развивающихся детей данного возраста, хранящимися в специальном файле на ГМД.

Одним из важнейших принципов психолого-педагогической диагностической работы, проводящейся специальными медико-педагогическими комиссиями (далее — МПК), является принцип объективности и точности оценки всех сторон психической деятельности ребенка и на этой основе адекватного прогнозирования его дальнейшего развития. Результат обследования должен быть максимально объективным, т. е. никак не должен зависеть от степени внимания, утомления экспериментатора, его эмоцио-

нального настроения, симпатий и антипатий. Все это также обеспечивается использованием компьютеров в работе МПК.

В целях оптимизации и гуманизации процесса обследования детей и внедрения количественной оценки их психической деятельности Орловской областной медико-педагогической консультацией совместно с творческой лабораторией по созданию педагогических программных средств "FEEDER" Орловского областного института усовершенствования учителей разработаны и апробированы две экспериментальные программы: система компьютерной психологической диагностики детей младшего школьного возраста и обучающая программа по методике компьютерного психолого-педагогического обследования детей младшего школьного возраста для студентов дефектологических факультетов педагогических институтов страны.

Программа предназначена не только для обучения методам компьютерной диагностики студентов, но и для начального психолого-педагогического обследования младших школьников. В связи с функцией обучения в программу вложены как методики исследования чистого интеллекта, так и методики обследования школьных навыков детей. В результате обследования выдается специальный графический профиль, а также коэффициент интеллекта в процентах к среднему уровню, принятому за 100 % в данном регионе.

Система компьютерной психодиагностики (далее — КПД) предназначена для психологического обследования интеллекта детей младшего школьного возраста в качестве специализированного инструментального средства.

Программы являются экспертно-диалоговыми, т. е. они реализуют тройственное взаимодействие:



Испытуемому предлагается ряд тестов, позволяющих исследовать различные стороны его психической деятельности, оператор-экспериментатор уточняет задание, выясняет, насколько оно понятно испытуемому, наблюдает за действиями, оценивает словесные или иные реакции ребенка и в соответствии с рекомендациями, предлагаемыми ему программой, вводит оценку его деятельности.

В некоторых методиках, реализованных в программе, предусмотрен диалоговый режим «компьютер — испытуемый», причем

последний пользуется специальным пультом, подсоединяемым к разъему джойстика. Оценка способа или результата деятельности испытуемого производится оператором-экспериментатором одинаковым во всех программах лаборатории "FEEDER" способом — нажатием 3 клавиш: "ESC", "BS" и пробела, что означает «выполнено», «не выполнено», «выполнено с помощью» соответственно. Выбор этих клавиш обусловлен тем, что они расположены по краям клавиатуры, а следовательно, ими проще научиться управлять и меньше накапливается усталость оператора-экспериментатора при работе. Кроме того, этот принцип выбора ответов обкатан на множестве обучающих программ для средней школы, разработанных лабораторией.

В зависимости от введенной оценки программы заносят в память определенное число, пропорциональное сложности задания, правильности результата и времени выполнения. Оценки затем складываются в контрольную сумму для вычисления коэффициента интеллекта.

Результатом обследования системы КПД являются как числовые данные к каждой методике, так и специальная диаграмма, показывающая объективно установленный уровень развития у ребенка различных показателей его психической деятельности относительно нормы для данного социокультурного региона.

Количество детей, которые могут быть обследованы системой КПД, неограниченно. Все результаты обследования заносятся в специальные файлы на ГМД, образующие базу данных.

В программах использован проверенный и многократно описанный в психолого-педагогической литературе набор экспериментально-психологических методик и заданий для проверки и оценки внимания, восприятия, памяти, мышления, работоспособности, обучаемости, уровня развития младших школьников и старших дошкольников, которые усовершенствованы и адаптированы к методике компьютерного изложения.

Программа КПД написана на Бейсике. Этот язык выбран потому, что система психодиагностики не требует большой скорости работы программы, а средства, обеспечиваемые этим языком на компьютере «Ямаха», предлагают довольно широкие возможности для создания красочного и узнаваемого оформления иллюстраций к методикам для того, чтобы заинтересовать обследуемого ребенка работой с компьютером. При разработке иллюстраций применялся специальный драйвер. "PAINTER. DRW" перевода графических изображений, нарисованных в штат-

ном редакторе "PAINTER", в команду "DRAW" Бейсика и последующей оптимизации этих данных, что способствовало уменьшению занимаемого объема иллюстрации в 5—7,5 раз в зависимости от сложности изображения.

Система КПД выполнена в виде трехблочного комплекса, из которых два блока — первоначальная загрузка четырех экранных страниц изображениями (около 37К байт, для сравнения: эти же страницы, записанные с помощью команды "COPY", занимают на диске более 110К байт), а третий — управление методиками и обработка результатов. Все три файла занимают на ГМД 53К байта. Программа снабжена "fool-proof"-системой. Введены ограничения на возраст обследуемых детей: не выше 9, не ниже 6,5 лет (верхнее ограничение снимается вводом информации о том, что обследуется аномальный ребенок). К программе разработан специальный альбом методик на случай отсутствия цветного монитора или для ребенка, который не может работать с дисплеем. Кроме того, в программе реализован принцип свободы выбора для оператора-экспериментатора: возможна работа как по запрограммированному порядку следования методик, так и по составленному экспериментатором, который может включать и одну методику, и несколько внепрограммных дополнительных. В системе КПД предусмотрен также режим «ОТКАТ», т. е. при неудачном прохождении методик можно вернуться назад по используемому порядку следования. Среднее время, уходящее на обследование и обработку результатов с выдачей стандартного протокола на одного ребенка с помощью данной программы, — около 40 мин. Для ра-

боты с программой необходима небольшая подготовка экспериментатора.

В процессе апробации программы стало видно, что дети полностью раскрываются, работая с компьютером. В некоторых случаях удавалось обследовать детей, страдающих нарушениями коммуникации, выражающимися в полном отказе от общения с незнакомыми взрослыми людьми. Такие дети могли полноценно общаться с компьютером, а также в процессе работы начинали незаметно для себя отвечать на вопросы и выполнять задания экспериментатора.

Не отрицая возможностей и необходимости обследования детей традиционными методами на МПК, мы рассматриваем данную систему компьютерной психодиагностики не только как необходимый дополнительный источник информации для правильного решения вопроса о дальнейшей судьбе ребенка, но и как достаточно гуманное средство, позволяющее интенсифицировать работу МПК путем более рационального использования рабочего времени при повышающемся качестве обследования. Более того, использование компьютера позволит перейти к массовым обследованиям детей в школьных и дошкольных учреждениях, решить проблему ранней диагностики аномального детства и правильного формирования массовых классов и классов выравнивания в средних школах.

*Адрес для переписки:* 302016, г. Орел, ул. Комсомольская, д. 120, кв. 57, Белоножко Олегу Павловичу.

Гонорар за опубликованную статью авторы просят перечислить в фонд помощи детям, пострадавшим от аварии на Чернобыльской АЭС.

93

---

## Пользователи УКНЦ!

Если вы хотите работать на своих компьютерах без использования дисководов, с одним только магнитофоном, то вам нужны программы:

**Бейсик (130 руб);**

**экранный редактор текстов (120 руб);**

**современный язык Паскаль (300 руб).**

Это и многое другое высылается по почте в виде кассет наложенным платежом.

Заявки направляйте по адресу:

**248010, Калуга-10, а/я 481, В. В. Маслову.**

---

# НИИ В ДОШКОЛЬНОМ ОБРАЗОВАНИИ

**Е. ЗВОРЫГИНА,**

НИИ дошкольного воспитания АПН СССР

## Педагогические подходы к компьютерным играм для дошкольников

В настоящее время в НИИ дошкольного воспитания АПН СССР в целях амплификации (обогащения) детского развития разрабатываются теоретические основы применения новых информационных технологий (компьютерных игр) в системе дидактики детского сада. Создано несколько серий программ для дошкольников, которые условно, в зависимости от педагогической направленности, можно разделить на три большие группы: обучающие, развивающие и диагностические.

К обучающим программам относятся в основном программы «предметного» характера — элементарные программы, обучающие отдельным «учебным» дисциплинам: математике, родному языку, музыке и др. Они знакомят детей с окружающим миром: миром предметов, людей, природы и т. д., в этих программах содержание, ход игры, как правило, четко очерчены.

Развивающие программы побуждают детей к творческим самостоятельным играм или к общению со сверстниками. Дети сами ищут способы решения игровых задач, свободны в выборе сюжетов и т. д. (программа «Конструктор», «Море» и др.).

Диагностические программы позволяют выявить уровень развития тех или иных умений, способностей, интересов ребенка.

Анализ теоретических материалов и результатов экспериментальной апроба-

ции игровых компьютерных программ в детских садах, оборудованных ЭВМ, показывает, что компьютерные программы имеют педагогическую значимость, если они направлены на поступательное, прогрессивное развитие творческой личности ребенка, на всестороннее развитие его способностей.

Для определения развивающей функции компьютерных игр для детей в разные возрастные периоды дошкольного детства особую ценность, на наш взгляд, представляют положения, выдвинутые советским психологом А. В. Запорожцем, об иерархии уровней психофизиологического развития ребенка, выделяемые в зависимости от уровня доминирующего типа действий.

А. В. Запорожец представляет развитие в виде трех взаимообусловленных уровней (этажей). Каждый этаж развития имеет свою самостоятельную ценность, вместе с тем он является базовым для становления следующего. Первый уровень развития как фундаментальный возникает в условиях воспитания детей, когда им представляется возможность активно познавать мир предметов, разнообразно действуя с ними (ранний возраст). Без богатого чувственного опыта, без опыта действия при непосредственном восприятии ситуации в плане наглядно-действенного мышления невозможен переход на следующий уровень. Для второго уровня развития (дошкольный возраст) характерны действия



воображения, осуществляемые в плане наглядно-образного мышления. Овладение новыми действиями в воображаемой ситуации и их качественные преобразования подводят детей к переходу на третий уровень развития, к оперированию понятийным мышлением с помощью знаковых систем (школьный возраст).

Указанная последовательность смены определенного типа действия непосредственно связана с переходом от наглядно-действенного мышления к более отвлеченному образному и речевому понятийному мышлению.

Итак, если у ребенка есть прочная база практического реального, полученного через разные органы чувств (зрение, слух, обоняние, осязание и др.) опыта (первый этап развития), малыша можно постепенно переводить к действиям без постоянной опоры на конкретный предмет (звук, запах и др.), к действиям с символом и знаком, его замещающим. Пример: ребенок уже в раннем возрасте начинает выделять количественную характеристику предметов, активно действуя с ними (безошибочно приносит одну или «много» игрушек, конфет), затем считает предметы (ступеньки, по которым идет, свои пальцы, звуки, слова, которые слышит), позднее конкретные объекты счета могут быть легко заменены абстрактными формами, их изображениями (палочками, кружочками). К концу дошкольного детства дети при счете могут оперировать новыми, совсем отвлеченными знаками — цифрами. При этом изменяются и сами действия, они тоже становятся более обобщенными, обозначающими, заменяются указательным жестом и словом, переходят из внешнего плана во внутренний, представляемый воображаемый план. Так, ребенок, считая предметы, сначала их перекладывает, затем лишь слегка прикасается к ним, позже, показывая на предметы, но не касаясь их, ребенок шепотом перечисляет порядковый номер и потом с ходу производит арифметические действия (считает в уме). Только ребенок, умеющий реально действовать с реальными предметами легко и свободно ориентируется во всем многообразии окружающего предметного мира, способен к переходу к действиям в умст-



венном плане, может мыслить о предметах и действиях с ними с опорой на речь и другие символы и знаки.

Тот или иной тип действия, переход к более абстрактному мышлению происходит благодаря общению ребенка со взрослым при своевременном формировании разных видов его деятельности: предметной (прежде всего при овладении предметами быта), конструктивной, двигательной, изобразительной, музыкальной и др.

Однако, будет ли малыш владеть самостоятельно и творчески соответствующим типом действия, прежде всего зависит от наиболее доступной ему деятельности — игры. Не случайно основоположники отечественной психологической школы деятельности Л. С. Выготский, А. С. Леонтьев, Д. Б. Эльконин и другие называют игру ведущей, т. е. определяющей психическое развитие детей до школы, деятельностью. Игра позволяет наиболее естественным путем приобщать малыша к тому или иному типу действия, без спешки и принуждения, применять его в разных условиях, оттачивать и обогащать его.

В игре (сначала с предметами, затем в воображаемой ситуации с опорой на предмет или символ, его изображающий) наиболее ярко прослеживается качественное преобразование игровых

действий, их свертывание, обобщение, замещение знаком (жестом, обозначающим словом).

В зависимости от преобразования игрового действия и обобщения опыта деятельности С. Л. Новоселова выделяет ряд этапов развития игры в раннем и дошкольном детстве (игра-манипулирование, отобразительная, сюжетно-отобразительная, сюжетно-ролевая игра). Поэтапно развивающая игра как деятельность обеспечивает и становление соответствующего возрасту типа действия и развитие символической замещающей функции мышления, что, в свою очередь, ведет к переходу мышления от наглядно-действенного к наглядно-образному и мышлению речевого понятийного типа.

Каков конкретный механизм усвоения того или иного типа действий в играх, каков конкретный путь овладения средствами замещения, путь становления творческой личности в игре? Мы полагаем, что ответ на этот вопрос даст представление об игре как деятельности, в которой дети последовательно решают игровые задачи. Игровую задачу (игровые действия, по А. Н. Леонтьеву) как структурную составляющую можно выделить в разных по содержанию, форме и организации видах игр.

Играя, дети не просто механически подражая или по образцу выполняют отдельные операции, они учатся ставить игровую цель и находить способы ее реализации в различных условиях.

Итак, игровая задача — система условий, в которой задается близкая и понятная воображаемая (представляемая) цель, направленная на воспроизведение, отображение жизненного опыта ребенка, который ему эмоционально дорог, который имеет для него особый личностный смысл. Поставленную цель ребенок достигает игровыми способами и средствами. Именно способы решения игровых задач являются носителями того или иного типа действия — это развернутые действия с игрушками и более-обобщенные, обозначающие действия, постепенно заменяемые знаком (жестом, словом), действия, переходящие в умственный план. Обобщенность игрового действия зависит от средств, которые дети

используют в играх. Игровые средства — это разные по образному решению игрушки, предметы-заместители, конкретные и абстрактные предметы, изображенные на плоскости, и т. д.

Выделяются две большие группы способов решения игровых задач в зависимости от сущности той стороны действительности, которую дети воспроизводят в игре: предметные и ролевые способы. Предметные способы решения игровых задач отображают преобразование предметов, действия людей в предметном мире. Например, в компьютерной игре «Построй дом» детям надо подобрать соответствующие половинки расколотого кирпича, в игре «Радуга» — найти среди хаотично разбросанных полосок, разных по длине, необходимые и приложить их к оставшимся на экране частями радуги, а затем «раскрасить» с последовательностью их расположения в спектре.

Содержание ролевых способов решения игровых задач — воспроизведение социальной сферы действительности (взаимодействие, отношение и общение людей). Например, в программе «Войти в замок» представлены два взаимодействующих персонажа, из которых каждый выполняет свою роль: стражник разрешает войти путнику в замок лишь в том случае, если он правильно найдет ответ на вопрос (вычислит одно из загаданных).

Предметные игровые способы решения задач — генетически раннее приобретение ребенка, поскольку опыт непосредственной практической результативной деятельности с предметами у ребенка складывается начиная с первого года жизни. Игра возникает в недрах предметной деятельности. К 3—3,5 годам при благоприятных условиях воспитания ребенок свободно использует в игре развернутые игровые действия с игрушками и начинает легко переходить к действиям более обобщенным, обозначающим (жесту и слову, заменяющему действие). Возникнув в раннем возрасте, эта группа предметных способов совершенствуется в дошкольном детстве, переносится на более широкий круг знаний.

Ролевые способы менее «уловимы» в жизни, они связаны с осознанием себя,

своего отношения к другим людям и др. Они начинают складываться лишь к 3 годам, когда появляется знаменитое «Я сам», когда ребенок имеет опыт решения игровых задач предметными способами и может осознать воображаемую ситуацию и условность игры, условность ролевого поведения («Я — как будто мама, он будет шофером» и т. п.).

Ролевые действия изначально являются более обобщенными по сравнению с предметными: чтобы назвать себя и другого мамой, врачом, стражником, необходимо выделить и обобщить множество характерных для этих людей действий, свойств личности, поведения и т. д. И ролевые способы, постепенно преобразовываясь, становятся все более обозначающими: последовательность их становления такова: эмоционально-выразительные ролевые действия и принятие роли, ролевые высказывания, ролевой диалог.

Освоение ролевого способа, как и предметного, начинается с наглядной ситуации (в плане наглядно-действенного мышления): например, детям важно увидеть и передать конкретное взаимодействие покупателя и продавца, для них значимо внешнее проявление эмоций, ярко выражающееся в мимике, позе, интонации и др. По мере перехода в план воображаемый большее значение приобретают разнообразные речевые характеристики состояний героев и ответные реакции также выражаются чаще словом.

Компьютерные программы могут иметь преимущество по сравнению с другими формами игр: в них ролевые характеристики персонажей, результаты и действия могут быть представлены более развернуто, а иногда и несколько утрированно, что позволит детям увидеть то, что в жизни скоротечно, едва уловимо. Дети в динамике могут наблюдать, как малейшее изменение совместных усилий изменяет результат. Например, сказка «Репка» в компьютерном варианте может наглядно показать, как важна помощь каждого героя. Дети могут увидеть более развернуто и эмоциональные реакции людей, способы взаимодействия и общений, научиться соотносить их с ситуациями, которые их вызывают, в которых они проявляются, и т. д.

В настоящее время таких программ для дошкольников крайне недостаточно.

Наши исследования убеждают, что компьютерные программы интересны дошкольникам и обеспечивают развивающий эффект, если они представлены в игровой форме, созданы по принципу решения игровых задач.

Детям интересны те игровые задачи, в которых отражаются события, жизненные ситуации, пусть самые необычные, фантастические, но обязательно отражающиеся на его впечатления, знания, полученные вне игры (в реальной жизни, при чтении книг и т. д.).

В компьютерной игре может быть воспроизведен один жизненный эпизод (одна игровая задача). Например, в математической игре «Лестница» детям необходимо последовательно перевести девочку со ступеньки на ступеньку (нажимая клавишу с соответствующей цифрой). В этой игре одна и та же игровая задача повторяется несколько раз (дети считают в пределах первого десятка, затем второго). Аналогично они считают и в обратном порядке, когда девочка «спускается» по ступенькам лесенки (вторая часть игры с одной повторяющейся игровой задачей).

В других компьютерных программах могут быть представлены несколько эпизодов, выраженных в игровых задачах, разных по иерархии. Сначала предлагается общая (главная цель, замысел), а затем более частные игровые задачи, последовательно ведущие к выполнению задуманного. Например, в игре по сказке «Гуси-лебеди» необходимо, спасая Аленушку с братцем, помочь им сосчитать яблоки на яблоне, составить мостик, чтобы они перешли через речку, найти заслонку для печки и т. д.

Отдельные жизненные эпизоды (отдельные игровые задачи), объединенные общим смыслом, всегда взаимосвязаны, иногда сочетание их необычно, причудливо, они имеют свою логику развития, влияют на эмоциональную насыщенность игры. Сюжетная линия игры, событийное содержание игровых задач могут быть связаны с любой стороной действительности (мир животных, космос, отношения людей к предметам, друг к другу, изменение предметов и т. д.).

97

Что из этого многообразия мира интересно ребенку, какие эпизоды и жизненные ситуации не оставляют его равнодушным? Ответ на этот вопрос не прост, у каждого ребенка свой опыт, свои потребности и мотивы деятельности. Однако нам представляется, что общие тенденции для определения сюжетного содержания игровых задач могут исходить из двух концептуальных положений.

Во-первых, по Л. С. Выготскому, ребенок играет в то, что он хочет, но не может осуществить в реальной жизни в силу своего возраста (поехать по-настоящему на лошади, машине и др.). Замечено, что городские дети предпочитают самые разнообразные игровые задачи с игрушечными животными, а сельские дети, принимающие участие в уходе за животными, придумывают необычные игры с машинами, которых у них нет и о которых они мечтают.

Во-вторых, содержание знаний, предлагаемых в игровых задачах, может быть разным для ребенка по степени ясности и четкости. Согласно положению Н. Н. Поддьякова, познавательная сфера ребенка состоит из системы ясных и неясных знаний. Для малыша не могут быть интересными игровые задачи с теми знаниями, которые он прочно четко усвоил или которые для него абсолютно новы, не связаны ни с предыдущими знаниями, ни с его эмоциональным опытом. Но знания, которые едва намечались, едва обозначены в перспективе, привлекают ребенка своей загадочностью, неопределенностью, возможностью их самому открыть, понять и др. Например, мальчик 4 лет впервые внимательно выслушал стихотворение С. Маршака «Пожар», в котором этично, мягко описывается о небольшой, очень понятной оплошности девочки приведшей к несчастью. Конкретный жизненный факт всколыхнул весь жизненный опыт ребенка, он очень осторожно и внимательно рассматривая газовую плиту, вспомнил все эпизоды «Кошкиного дома» (как можно тушить пожар), он отыскал пожарную машину, долго играл с ней, ездил на пожар, тушил, спасал и т. д., задавал бесконечные вопросы взрослым, рассуждал по этому поводу и т. д. Для него доступная игра с этим

сюжетным содержанием («Про пожар») была бы своевременна и интересна. При отсутствии подобного опыта вряд ли у ребенка появится интерес к игре на эту тему.

Итак, событийная сюжетная линия игры, исходящая из игровых целей и замыслов (продукт воображения автора), лежит на поверхности и прежде всего привлекает ребенка.

Способы же решения игровых задач, будучи органически связанными с игровыми целями, выражают более глубинную деятельностную сущность игры, они обычно не осознаются ребенком, хотя, безусловно, в значительной степени влияют на эмоциональную сферу играющего. Как показывает практика игры в детском саду, способы игрового воспроизведения действительности лежат, как правило, и вне поля зрения педагога. Вместе с тем без четкого выделения игровых способов и средств нельзя поэтапно усложнять игру, а следовательно, косвенно развивать общие интеллектуальные способности дошкольников. Важно научиться в каждой игре не только выделять сюжетную канву, обеспечивающую конкретный обучающий и развивающий эффект (учить элементам математики, нравственно совершенствовать), но и суметь вычленив игровые задачи, способы и средства их решения.

Как определяются способы решения игровых задач в той или иной программе? Зная последовательность смены типа действий, этапы развития игры, можно создавать компьютерные игры на закрепление тех способов, которыми ребенок владеет, и, учитывая зону ближайшего развития (по Л. С. Выготскому), последовательно усложнять их, предлагая каждый раз что-либо новое. Например, в пределах предметных способов постепенно увеличивать абстрактность изображения. Так, в одной игре доминирующими могут быть действия с конкретными изображениями предметов, в другой, например в игре «Найди пару», дети оперируют с конкретными изображениями (рыбка, цыпленок и т. д.) и отвлеченными формами (круглые, овальные, квадратные); в третьей игре дети оперируют только с абстрактными формами («Конструктор», «Мозаика»).

Хорошо играющий ребенок принимает игровую задачу от взрослого или ставит ее сам. Он пытается выполнить ее тоже сам или с небольшой, чаще косвенной помощью взрослого. При этом он не может остаться равнодушным ни к ее сюжетному содержанию, ни к тем способам, средствам, которыми достигается игровая цель. Так, обеспечивается единство эффекта и интеллекта чрезвычайно необходимое для развития мотивов и потребностей детей, для становления творческой личности в дошкольном возрасте.

Как подвести детей к самостоятельным играм? Мы полагаем, что предъявление детям системы постепенно усложняющихся по содержанию знаний и способам решения игровых задач может обеспечить развитие подлинной творческой самостоятельной игры, в том числе и игры на компьютере.

Побуждая детей к принятию или постановке игровых задач, к поиску своих вариантов ее выполнения, педагог развивает у них инициативу, творческий потенциал. Ребенок в игре всегда предстает не как послушный исполнитель, а как творец своей деятельности.

В зависимости от степени свободы детей в постановке и решении игровых задач выделяются разные группы игр. В так называемых творческих играх (сюжетных, отчасти театрализованных) дети, отображая свои впечатления о мире, сами ставят игровые задачи и ищут пути их реализации.

В других содержание игровых задач, способы, средства и правила игры определены взрослыми. Эти игры традиционно передавались в народной педагогике для умственного (большая группа дидактических игр), физического (подвижные игры) воспитания, для того чтобы позабавить, порадовать, удивить ребенка (игры-забавы). Много разных игр всех видов созданы официальной дошкольной педагогикой.

Все компьютерные программы для дошкольников (обучающие, развивающие, диагностические) могут быть представлены как игры разных видов. Игровые компьютерные программы могут носить характер дидактических, сюжетно-режиссерских, театрализованных и даже

подвижных игр, игр-забав, игр-экспериментов.

Когда можно приобщать детей к играм на компьютере?

Компьютерные игровые программы не призваны заменить традиционные успешно применяемые методы и формы воспитания и обучения дошкольников. К игре на компьютере ребенок может приобщиться лишь при условии своевременного развития разных видов деятельности детей: предметно-продуктивной, игровой, музыкальной, конструктивной, изобразительной и др. Они являются базовыми для формирования того или иного типа действий. В компьютерных играх тот или иной тип действия, которым дети начали овладевать, например, при практическом конструировании, может осваиваться глубже, могут последовательно «оттачиваться» операции его составляющие.

99

Особую роль в подготовке детей к игре на компьютере выполняют игра, ее разные виды, широко распространенные в практике дошкольного воспитания (сюжетные, дидактические, подвижные и др.). В этих играх дети овладевают двумя важнейшими позициями, без которых невозможна игра на компьютере. Во-первых, они учатся достигать цель с помощью предмета-орудия. Осознание опосредованности результата действия поможет ребенку быстро установить связь между клавиатурой компьютера и изображением на экране. Уже в 3 года



дети могут улавливать простейшие способы управления на компьютере. В таком случае компьютер для них выступает сразу не как партнер, диктующий свои условия, а как новое средство, новое технологически сложное, но доступное орудие для достижения игровой цели. Во-вторых, в традиционных играх с раннего возраста обеспечивается плавный переход от предметных результативных действий к действиям в воображаемой ситуации, к действиям замещения с опорой на предмет-игрушку, на предмет-заместитель и другие более обобщенные знаки: жест и слова. Например, малыш, умывая куклу, вместо мыла берет кубик, затем он может лишь жестом показать, что он взял «мыло» с иллюстрации в книге «Мойдодыр» К. Чуковского, позднее он не берет ничего, а потирая руки куклы, говорит, что он моет их с мылом. Способность к замещению реального предмета игровым с переносом на этот предмет реального значения, реального действия игровым является необходимым условием осмысленного оперирования символами и знаками на экране компьютера, а также позволяет легко ориентироваться в условном обозначении клавиш компьютера.

Если ценное в воспитательном и образовательном отношении знание, конкретная учебная задача естественным образом смыкаются с содержанием игровой задачи, ребенок усваивает материал незаметно для себя, с увлечением играя, мобилизует при этом весь свой жизненный опыт, максимум внимания, произвольности и проявляет инициативу и творчество.

Включая компьютерные игры в общую систему игр, в педагогический процесс детского сада, необходимо подчеркнуть особую важность самостоятельных игр детей, когда они свободны в выборе вида игр, содержания и способов решения игровых задач. В таких играх, особенно сюжетного характера, в которых воспроизводятся жизненные впечатления, дети используют весь интеллектуальный и эмоциональный багаж. Они сами регулируют сложные и простые способы решения игровых задач и т. д.

Благодаря этому создается необходимое естественное равновесие между на-

глядно-действенным, наглядно-образным, речевым, понятийным мышлением.

Поэтому приобщение к играм на компьютерах никоим образом не должно вести к вытеснению сюжетных игр из повседневного обихода детей. Более того, необходимо продумать, как компьютерные игры, особенно развивающие (сюжетно-режиссерские) игры сочетать с содержанием сюжетно-ролевых игр и других видов деятельности (например, рисованием, конструированием). Они взаимно обогатят друг друга.

Итак, компьютерным играм для дошкольников свойственны общие закономерности, присущие игровой деятельности, о которых писалось выше. Вместе с тем они имеют свои правила, свою специфику, обусловленную технологическими особенностями этой новой «игрушки», а также особенностями психофизиологического воздействия компьютерных игр на детей.

Особые требования предъявляются к содержанию игровых задач, к той конкретной информации, которую несут дошкольникам компьютерные программы.

Создать интересные, даже уникальные компьютерные программы для дошкольников может каждый, кто знает ребенка, интуитивно или сознательно ориентируется в специфике его восприятия мира.

Однако систему развивающих компьютерных программ для дошкольников могут разрабатывать лишь детские психологи и педагоги с привлечением специалистов самых разных областей знаний. (Так, математические программы не могут быть созданы без математиков.)

Определить возможность и необходимость использования компьютера в музыкальном воспитании дошкольников может только профессионал-музыкант, знающий специфику художественного воспитания детей. Только он может дать теоретическое обоснование необходимости музыкальных компьютерных игр, найти их место и роль в общей системе музыкального воспитания, определить основное содержание музыкальных компьютерных программ. Он может написать и сценарий, обеспечивающий или ознакомление с новым материалом, или его закрепление, и т. д.

Вместе с тем необходимо рационально использовать и уникальные технологические возможности компьютерных игр. Например, динамичный характер компьютерного представления учебного содержания позволяет показать многие явления в виде непрерывных взаимосвязанных преобразований. Поэтому в компьютерных программах закладываются те знания, которые не могут быть усвоены в обычных условиях во всем многообразии и единстве (космический мир), которые надо выделить, представить более выпукло, в динамике; показать то, что ребенок не может или не успевает заметить в естественных условиях (например, как раскрывается бутон цветка, как растут корни дерева и т. д.).

Компьютерные игры создают благоприятные условия для экологического воспитания детей, для становления целостного представления о мире. В них можно отражать естественное единство живой и неживой природы, показать, как отдельные действия, поступки людей изменяют природу (лес, реки), и т. д. Компьютерные программы могут иметь преимущество по сравнению с другими формами игр в плане возможности наглядно показать детям ролевые способы решения игровых задач, например в динамике продемонстрировать результаты совместных действий и общения персонажей, их эмоциональные реакции при успехе, неудаче, что в жизни скоротечно, внешне едва уловимо (огорчение, сочувствие и т. д.).

Компьютерных программ для дошкольников с ролевыми способами решения игровых задач пока необоснованно мало. Между тем именно программы с ролевым поведением персонажей могут привлечь внимание детей к внутреннему миру другого, побуждают поставить себя на его место, помочь персонажу преодолеть препятствия, порадоваться за его успехи. Образцом сюжета таких игр могут служить народные сказки и другие произведения фольклора.

В таких играх дети приобретают опыт нравственного поведения в самых разнообразных условиях жизни, что, формируя эмоционально-нравственную ориентированность, помогает избежать штампов и стандартов в оценке поведения

разных персонажей в разных ситуациях. Дети, решая игровые задачи с ролевым поведением персонажей, практически усваивают средства коммуникации, способы общения и выражения эмоций.

Надо заметить, что компьютерные игры могут способствовать деловому сотрудничеству и речевому общению детей, если они организуются не только как индивидуальные, но групповые игры (для 2—3 детей).

Все компьютерные программы для дошкольников должны иметь положительную нравственную направленность, исключается агрессивность, жестокость, насилие. Вызывают интерес программы с элементами новизны, сюрпризности, необычности.

Однако не стоит перенасыщать их обилием «сюрпризов». Это может привести к перенапряжению и утомлению ребенка.

Компьютерные программы, как правило, строятся по принципу самоконтроля, сам сюжет программы подсказывает детям, верный или неверный ход решения они приняли. Например, в программе «Лестница» при ошибке в порядковом счете девочка, изображенная на экране, не переходит на следующую ступеньку. Вместе с тем в дошкольном возрасте широко применяются приемы и внешне-го поощрения, при правильном решении игровых задач ребенок слышит веселую мелодию, видит на экране улыбающееся лицо и, наоборот, грустную мелодию или печальное лицо, если задача решена не верно.

Эксперимент показал, что дети ждут оценку, эмоционально реагируют на ее характер.

У всех детей отмечается яркое, эмоциональное, положительное отношение к игре на компьютере. Неприятие компьютера, психологического барьера почти не наблюдается.

В компьютерных играх особые, более опосредованные, более сложные правила, они представлены в двух планах: во внешнем плане (управление клавишами) и внутреннем (правила по содержанию игры). Например, в дидактической компьютерной игре «Лето» необходимо из четырех частей составить картину (внутренние правила игры), но для этого надо

сначала научиться управлять курсором (на поле, в котором складываются картинка) и стрелкой (показывать, какую картинку надо выбрать).

Ознакомление детей с клавиатурой, с внешними правилами может быть выделено как особая задача, и для этого уже созданы специальные усложняющиеся по принципу управления изображениями на экране компьютера игровые программы. Одна из первых программ, знакомящих детей со спецификой компьютерных игр, — «Веселые картинки» — доступна детям 2,5—3-летнего возраста.

Таким образом, мы полагаем, что компьютерные игры не заменяют, а дополняют все традиционные формы игр и занятий, являясь естественным путем приобщения дошкольников к новым информационным технологиям, к оперированию знаковыми формами мышления, если они органически включены в ведущую деятельность дошкольника — игру.

Благодаря взаимосвязи разных видов игр, в том числе и компьютерных, ребенок овладевает умением самостоятельно и инициативно решать игровые задачи, постепенно усложняющиеся по содержанию знаний и степени обобщенности действий, поднимаясь в своем развитии на более высокий уровень.

Результаты экспериментальной апробации компьютерно-игрового комплекса в детском саду позволяют сказать о следующем.

Компьютер может быть широко использован в целях обучения и развития дошкольников, он может быть объективным диагностическим средством, позво-

ляющим выявить уровень развития тех или иных умений, способностей, интересов ребенка.

Через компьютер можно преодолеть «социальную изоляцию» ребенка, поскольку он дает объективную оценку интеллектуальным возможностям детей (особенно робких, не проявивших себя в коллективных занятиях и играх).

Компьютер может быть использован как мощное и тонкое средство, раскрывающее способности детей, побуждающее взрослого искать новые, нетрадиционные формы и методы воспитания детей в условиях современной цивилизации. Это уникальная игрушка, имеющая возможность настраиваться на конкретного ребенка, обеспечивая ему деятельность в зоне ближайшего развития.

Компьютерные игры в дошкольном детстве имеют особую направленность. Они не только стимулируют индивидуальную деятельность детей, их творческий потенциал, но и являются замечательным средством, объединяющим детей в интересных коллективных играх, способствуют их неформальному общению.

Компьютерные игры, в отличие от других видов игр, позволяют увидеть продукт своего воображения и своей игры; в отличие от других видов продуктивной деятельности (рисования, конструирования и др.) они позволяют восстановить динамику своего творчества. Все это ведет уже в дошкольном возрасте к становлению способности объективно оценивать результаты и ход собственной деятельности.

Е. ГЛУШКОВА, Л. ЛЕОНОВА, З. САЗАНЮК, М. СТЕПАНОВА

## Гигиенические требования к занятиям для дошкольников

Компьютеризация школьного образования в нашей стране имеет свою, хоть и недолгую, историю. А вот использование компьютеров для занятий детей в детских садах еще только начинается. Это новшество требует самого тщательного изучения со стороны не

только педагогов и психологов, но и медиков. Основания для беспокойства самые веские: гигиенические исследования, проведенные в школах, показали, что занятия с компьютером могут приводить к жалобам на зрительное и общее утомление. Выраженность измене-



ний зависит от организации и длительности работы с экраном ПЭВМ, качества изображения, состояния микроклимата, методики обучения, содержания занятий и, конечно, возраста ребенка. Особую тревогу вызывает тот факт, что детские, в том числе и дошкольные, учреждения оснащаются компьютерной техникой низкого качества, не отвечающей гигиеническим требованиям детского организма. Экраном нередко служит телевизор, на котором принципиально невозможно достичь необходимой четкости, контрастности и других параметров, обеспечивающих оптимальные условия для зрительной работы.

Дошкольники более чувствительны к воздействию различных факторов среды, поскольку их организм находится в состоянии интенсивного психофизиологического и морфофункционального развития. Ведь именно в возрасте 5—6 лет формируется нормальная рефракция глаза, происходит переход физиологической дальнозоркой рефракции в нормальную или близорукую, если к этому имеются генетические предпосылки или условия зрительной работы не соответствуют гигиеническим требованиям. Интенсивно развиваются костно-мышечная система, произвольное внимание и многие другие функции, определяющие общее развитие ребенка.

Поэтому очень важно, чтобы занятия дошкольников с компьютером не оказали неблагоприятного воздействия на их здоровье.

Среди множества проблем, связанных с компьютеризацией дошкольного образования, одной из основных является разработка гигиенически рациональных режимов занятий детей на ЭВМ. Совместные исследования в этом направлении начаты сотрудниками НИИ гигиены и профилактики заболеваний среди детей и подростков Минздрава СССР и НИИ физиологии детей и подростков АПН СССР.

Изучение влияния игровых занятий с компьютером на функциональное состояние организма детей дошкольного возраста проводилось в детском саду № 1623 Москвы, в учебно-воспитательном процессе которого используется

компьютерно-игровой комплекс (КИК) «Электроник». В состав КИК входит ЭВМ БК-0010 с видеотерминалом цветного изображения. Занятия проходили в специально оборудованном компьютерном зале.

Содержание программного обеспечения занятий разработано в ЦНИИ «Электроника» и НИИ дошкольного воспитания АПН СССР. Анализ используемых игровых программ показал, что все они соответствуют возрастным возможностям детей.

Под наблюдением находилось 35 детей 6-летнего возраста. Все они дважды (осенью и весной) прошли тщательный медицинский осмотр.

Для оценки влияния игровых занятий с использованием КИК до и после этих занятий (в течение года) определялись функциональное состояние зрительного анализатора (глазная эргометрия), центральной нервной системы (ЦНС) (хронорефлексометрия) и умственная работоспособность (корректирующая проба)\*.

В процессе игровой деятельности проводились хронометражные наблюдения за поведением детей, в том числе и за характером изменения позы. Дополнительным контролем за функциональным состоянием организма детей, степенью их утомления служили исследования умственной работоспособности в течение всего дня пребывания в детском саду.

Результаты исследования показали, что доля непосредственного общения с экраном дисплея во время занятий дошкольников высока и составляет 78—82 % от общей длительности занятий. Для определения допустимой одноразовой длительности занятий с компьютером детей 6 лет исследования проводились спустя 5, 10 и 15 минут после начала занятия. 5-минутные игровые занятия детей не вызывали неблагоприятных изменений в организме — функциональное состояние ЦНС и зрительного анализатора практически не изменялось. При 10-минутной игре

\* В проведении экспериментальных исследований принимали участие сотрудники НИИ физиологии детей и подростков АПН СССР А. Бирюкович, Л. Макарова, Т. Изотова.

внешние признаки утомления (двигательное беспокойство, отвлечения) отмечались лишь в отдельных случаях и в самом конце игры — на 10-й минуте. Вместе с тем физиологические исследования выявили признаки снижения функционального состояния ЦНС (в 5 % случаев) и ухудшение показателей состояния аккомодационного аппарата глаза (в 20 % случаев). Таким образом, десятиминутная продолжительность игровых занятий с компьютером у части детей приводит к появлению признаков утомления. В первую очередь страдает аккомодационная система глаза, обеспечивающая четкое рассматривание предметов на близком расстоянии.

При 15-минутной продолжительности занятия у всех детей уже на 11—12-й минуте появлялось и усиливалось двигательное беспокойство, нарастало число отвлечений и ошибок при игре, снижался интерес к ней. Наблюдалось снижение функциональных возможностей зрительного анализатора и ЦНС. Это выражалось в том, что у большинства детей достоверно увеличивалось число срывов дифференцированной реакции, уменьшался объем аккомодации (ухудшение работоспособности цилиарной мышцы).

Полученные данные свидетельствуют, что предельно допустимая длительность игровых занятий за ЭВМ такого типа для детей 6 лет не должна превышать 10 мин.

Контроль за функциональным состоянием организма детей при ежедневных занятиях с компьютером в течение недели показал, что по большинству показателей, отражающих состояние ЦНС, умственной работоспособности и зрительного анализатора, неблагоприятные изменения состояния отмечались преимущественно в понедельник и пятницу.

В конце недели отмечалось наибольшее снижение уровня умственной работоспособности детей: более половины заканчивали компьютерные занятия с выраженным утомлением. У всех детей выявлено значительное ухудшение функциональных возможностей зрительного

анализатора. Все показатели, отражающие состояние аккомодационного аппарата глаза, ухудшались: уменьшались объем и устойчивость аккомодации. Примерно такая же картина отмечалась нами и в понедельник.

Таким образом, наиболее благоприятными днями для проведения занятий с компьютером такого типа у детей 6 лет являются вторник, среда и четверг. Однако индивидуальный анализ данных показал, что без ущерба для состояния здоровья дети могут заниматься на ЭВМ лишь 1—2 раза в неделю. При большей кратности занятий (3 или более раз в неделю) большинство детей значительно утомляются.

Выраженное утомление после занятий на компьютерах диагностировалось у половины детей и в том случае, когда они проводились после 3 обязательных (учебных) занятий или вместо прогулки. Более благоприятная реакция отмечалась в тех случаях, когда компьютеры использовались на 3-м часу обязательных учебных занятий.

Исследования показали, что большое значение при занятиях на ЭВМ имеет правильная организация рабочего места. В группе детей, 10-минутные занятия которых проходили в условиях, не соответствующих гигиеническим требованиям (мебель не подбиралась по росту ребенка, отсутствовала спинка у стула, не было систематического контроля за осанкой детей), в значительном числе случаев (32 %) отмечалось ухудшение осанки от начала к концу года.

Учитывая, что около трети 6-летних детей в начале года уже имели те или иные функциональные нарушения осанки, эти результаты вызывают серьезные опасения.

Признавая, что компьютер — это новое мощное средство для интеллектуального развития детей, необходимо помнить, что его использование в учебно-воспитательных целях в дошкольных учреждениях требует тщательной организации как самих занятий, так и всего режима дня в целом.

## О программах для «Корвета»

О надежности первых партий «Корветов» написано достаточно, поэтому о процедуре установки этого компьютерного комплекта в КВТ мы умолчим, тем более что в процессе эксплуатации выявились новые интересные вещи. Например, генератор псевдослучайных чисел RND выдает повторяющиеся последовательности чисел длиной всего-навсего 24! Более того, каждое третье число в этой последовательности равно 0,618979. Оператор же RANDOMIZE, который мог бы помочь в этом случае, работает лишь на учительском компьютере ПК 8020, а на ученическом ПК 8010 его нет. Сколько минусов сулит отсутствие добротного RND, объяснить не нужно. К счастью, нам удалось найти способ, позволяющий ликвидировать этот недостаток.

Для этого надо РМП по локальной сети загрузить на все РМУ программу, состоящую всего из одной строки:

```
10 PRINT:END
```

Она загружается в ОЗУ с адреса 9000 и «засевает» дисплей девятками. После сброса их кнопок «Сброс» RND на РМУ работает нормально. Если в программе, загружаемой на РМУ, необходим оператор RANDOMIZE, то заменить его можно путем введения в основную программу двух дополнительных строчек:

```
6 PRINT "Нажмите клавишу СТРН"  
7 AA=RND(1):IF INKEY$<>CHR$(31) THEN 7
```

Программное обеспечение с КУВТ «Корвет» не поставлялось, даже в областном магазине «Учколлектор» его сначала не было. Но в сентябре, к нашему большому облегчению, туда поступили пакеты 1, 2 и 3 «Диалог», выпущенные Казанским НПО ВТИ и поначалу внушившие симпатии своими низкими ценами: шесть дискет — менее 100 рублей.

Радость наша была омрачена довольно

быстро. При работе с «Диалогом» были выявлены как новые отрицательные стороны у «Корвета», так и недостатки пакета программ.

Оказалось, что операционная система, записанная на дискете «Диалога», не обеспечивает нормальной работы по некоторым параметрам. При попытке определить размер файла программой HDIR.COM компьютер выдает нереальные числа. В то же время при загрузке операционной системы с других дисков эта программа работает хорошо. Есть у этой ОС и другие слабые места.

Программы на Бейсике объемом свыше 12К байт, загружаемые с РМП по локальной сети в РМУ, не работают. «Поломка» в программе происходит или во время загрузки, или в процессе работы, причем всегда в одном и том же месте программы.

Например, в программе КОД.BAS (пакет № 3) «ломается» строка 4510. Чтобы программа работала, перед этой строкой вводим строку 4505 достаточной длины и не влияющую на работу программы, например

```
4505 REM Эта строка введена в программу,  
чтобы предотвратить поломку,  
разрушающую строку 4510
```

«Поломка» теперь будет в этой строке. Таким способом были исправлены и другие программы.

Учебные программы «Диалога» не позволяют осуществить эффективную проверку знаний в связи с тем, что вариант контрольного задания не задается случайным образом, а его выбирает сам учащийся (программы КОД.BAS, ВРР.BAS, LOGI-KA.BAS из пакета № 3).

Некоторые программы «Диалога» не работают вообще вследствие грубых ошибок. Похоже, в Казанском НПО ВТИ даже не удосужились проверить эти программы на ПЭВМ, перед тем как приступить к их тиражированию. Так, в программе DEMON1.BAS неправильно записан оператор

установки цвета LUT. Он встречается в программе раз 40, а сама программа имеет объем 15К байт. Чтобы исправить только этот оператор, понадобилось два часа, а чтобы добиться работоспособности всей программы — около шести часов. Неужели каждый преподаватель ОИВТ должен заниматься этим самостоятельно?

Даже в игровой программе PAUT.BAS (пакет № 1), состоящей всего из четырех десятков строк, была ошибка.

Возможно, мы не стали бы так подробно останавливаться на плюсах и минусах первых ППС для «Корвета», выпущенных Казанским НПО ВТИ, если бы не столь широкий тираж этих программных средств и если бы не еще одно обстоятельство: недавно мы получили новый вариант пакета № 3 «Диалог». В нем добавлена лишь одна программа, а вот в старых ошибки так и не исправлены. Что же, так и оставим ошибки на конвейере?

В ноябре нам был прислан пакет программных средств «Парус» Черновицкого клуба «Компьютер». «Парус», как и «Диалог», был разослан во все школы Тамбовской области, имеющие КУВТ «Корвет», магазином «Учколлектор». Далекое не идеальным оказалось и это программное средство. При довольно солидной цене (более 400

рублей) оно записано на одной дискете, причем скопировать его на рабочие дискеты средствами, поставленными в комплекте с «Корветом», невозможно. А ведь если эту дискету использовать в качестве рабочей, а не в качестве контрольного образца, то долго она не прослужит!

Некоторые программы «Паруса» оказались неработающими (KLAW.BAS, BAZA.BAS и др.) Все обучающие программы по Бейсику (UROK 1.BAS — UROK 10.BAS) не позволяют контролировать знания учащихся, так как контроль производится на всех РМУ с помощью одного варианта проверки, и, только если учащийся выполняет программу второй раз, он получает другой вариант. Часть программ выполнена на примитивном уровне и тиражирования, на наш взгляд, никак не заслуживает.

В заключение мы предлагаем всем желающим поделиться опытом работы с КУВТ «Корвет», обменяться программами, рассказать о новых методиках преподавания информатики. Пишите нам по адресу: 393000 Тамбовская обл., пгт Дмитриевка, Никифоровский р-н, поселок сахарного завода, школа № 2.

Ю. ЛАПИН,  
В. РЕДЧЕНКО

## Чтобы не бегать по классу

При работе с Бейсик-программами на УКНЦ учителю не обязательно бегать по классу, чтобы найти и исправить ошибку, дать совет и т. д. Можно получить на экране монитора РМП текст Бейсик-программы, с которой работает ученик на любом РМУ.

Для этого:

1) устанавливается программа связи с классом NETUK.SAV в режиме записи;

2) ученик на РМУ набирает команду SAVE «ТТ» и нажимает ВВОД.

После этого на экране монитора РМП начинает выводиться текст программы, с которой работает ученик. Используя комбинацию клавиш УПР/Ч (приостановить вывод информации) и УПР/О (продолжить вывод

информации), преподаватель просматривает программу. Замечу, что команды приостановки и возобновления выдачи информации на экран в различных системах могут быть разными.

Во время просмотра NETUK продолжает функционировать.

Буду рад, если мой совет кому-нибудь пригодится: мне этот прием иногда очень помогает.

Со своей стороны был бы очень рад опубликованию какого-либо материала по работе на УКНЦ с принтером Д100М (в частности, как с его помощью можно распечатать «картинки»?).

А. РОЗЕНФЕЛЬД, Алтайский край

## Ищем спонсоров

От редакции. С проблемой, описанной автором письма, столкнулись многие учебно-производственные комбинаты. Об этом нам сообщили в Гособразовании СССР, где уже подготовлен документ, расширяющий финансовые права и экономическую самостоятельность УПК. Мы предполагаем опубликовать его в следующем номере.

УПК № 3 Черемушкинского района Москвы готовит лаборантов-программистов, нужную и перспективную специальность. Мы имеем большой парк разнообразной вычислительной техники, собственные программные и методические разработки. Накоплен немалый опыт профессионального обучения. Каждый год в УПК приходят около тысячи детей. Мы стараемся принять всех, понимая необходимость компьютерной подготовки для поколения, которому предстоит жить и работать в информационном обществе. Сложился сильный коллектив преподавателей — высококвалифицированные специалисты, многие с ученой степенью. Создание работоспособного, «заряженного на дело» коллектива было

делом не простым. Однако все наши усилия могут оказаться напрасными.

В этом учебном году мы почувствовали, что никому, кроме учащихся, мы не нужны. Дело в том, что оплата преподавателям за теоретические занятия производится из фондов районного отдела народного образования, а оплата практических занятий — из фонда предприятий. В нынешней экономической ситуации предприятия, для которых и мы готовим квалифицированные кадры, отказываются заключать с нами договора.

Обращаемся ко всем заинтересованным организациям, ассоциациям, кооперативам с просьбой оказать финансовую поддержку. Со своей стороны мы могли бы предоставить методические материалы и программные разработки, давать консультации, поделиться опытом организации учебного процесса.

Наш расчетный счет 142024 ПСБ МФО 201490 Черемушкинского района г. Москвы.

Адрес для справок: 117485, г. Москва, ул. Бутлерова, д. 5, УПК-3. Тел.: 330—20—83 — Краснова Рита Иосифовна.

**Р. КРАСНОВА**  
директор УПК № 3

107

## Письмо и комментарий

*Работаю в сельской средней школе более 15 лет. Сейчас веду уроки физики, трудовой подготовки в старших классах и ОИВТ с первого года. Работал и без машин, потом появился выход на ЭВМ, а два года назад получили УКНЦ.*

*Первоначально «ИНФО» мне очень помог (я брал из него до 80 % материала), а сейчас (последние два года) — только до 25 %.*

*Почему?*

*Очевидно, накопился кое-какой свой опыт, но и нужного материала стало мало (так думают и мои коллеги).*

*В преподавании информатики методика еще не сложилась. Кто должен заниматься этим? Наверное, Академия педнаук. Но и учителя тоже могут внести свою лепту, так как накопили практический материал.*

*Почему вам мало пишут учителя-практики (или вы мало их публикуете)? Очевидно, причиной является ваше ограничение: требования к рукописям!!! Учителю очень трудно выполнить эти требования (отпечатать на*

*машинке и т. д.), поэтому у вас публикуются статьи руководителей, кто имеет своих секретарей, штат.*

*Я не против нужных (но не заумных) статей (например, А. Архангельский «Мир ЭВМ» — где можно еще взять описание ОС?).*

*Предложение 1. Разрешить учителям присылать рукописи (до 2—3 листов), не соблюдая требований, — я уверен, в таких статьях можно найти много «изюминок».*

*Что такое компьютер без ППС? Язык с клавишами, и только! Где взять программы? Ваш журнал давал объявления двух кооперативов — «Полином» и «Электрон». У первого мало программ, а второй заломил высокие цены (посылаю вам каталоги — можете убедиться сами).*

*Предложение 2. Напечатайте мое объявление.*

*«Александров Борис Федорович предлагает всем владельцам УКНЦ обмен имеющимся*

программами. Условие: дискеты присылать отформатированными и инициализированными; если у вас нет программ для обмена, копирую свои (одну дискету копирую, одну беру за работу). Помогите советом в освоении компьютерного класса. Адрес: 142636, Московская обл., Орехово-Зуевский р-н, п/о Яковлево, дер. Савино, 8, кв. 48.»

*И последнее. КУВТ УКНЦ — это будущее для школ, которое уже сейчас работает.*

*Предложение 3. Расширьте страничку клуба пользователей БК и отведите там место и для УКНЦ.*

*Я уверен: по крупицам можно собрать многое. В школах, на местах, на периферии есть что сказать. Надо это собирать!*

В этом письме затронуты много важных тем. Например, «нужный материал». Что нужно учителю? Сценарий урока, способы ремонта ВТ, адреса продавцов программ, курсы программирования, идеология построения ППС? Стратегия информатизация школы и общества? К сожалению, ответов на анкету ИНФО в 1989 г. было все же не настолько много, чтобы достаточно ясно представить спектр потребностей читателей. Мы повторим анкету в 1991 г. и надеемся, что на этот раз ее результаты позволят нам лучше учитывать эти потребности.

Упрек в завышении требований к оформлению рукописи нам кажется несправедливым. В каждой школе есть пишущая машинка. А у автора письма есть и принтер, однако он написал письмо от руки. А ведь чем больше времени мы будем тратить на разбор почерков (это проблема не только техническая: чтобы, например, послать на рецензию, надо дать машинистке перепечатать, а у нее, во-первых, норма, предусматривающая перепечатку статей, идущих в номер, а во-вторых, перепечатка рукописного текста просто не входит в ее обязанности), тем меньше времени останется на содержательную часть работы. А ведь автор пишет нам раз в год, мы же получаем рукописи ежедневно.

Что же касается статей руководителей... Это просто неверно. Посмотрим, например, № 5: две статьи руководителей, 12 статей не руководящих работников педагогических институтов и институтов АПН СССР, 12 статей учителей и любителей программирования.

Теперь о главном — доступности ППС. Неформальная сеть распространения программ способна эффективно обеспечить педагогов уже имеющимися ППС, но как она повлияет на создание новых? Конечно, всегда найдутся энтузиасты, готовые раздавать свои программы бесплатно. Однако программы любителей, как правило, далеки от совершенства. Некоторые талантливые педагоги даже

в свободное от основной работы время смогут делать по-настоящему хорошие программы; но постепенно им станет в тягость ночами программировать, а днем зарабатывать на жизнь. Таланту свойственно стремиться заниматься именно своим делом. Тут-то и скажется порочность предлагаемой системы. Как только программа попадет во владение «неформальных распространителей», она размножается в необходимом количестве экземпляров; в предельном случае автор может продать ее только один раз. Следовательно, цена должна быть высокой, а кому она будет по карману? И окажется, что для профессионального педагога-программиста в этой системе места нет. А на госзаказ и централизованные закупки пока надежды мало: опыт показывает, что они отнюдь не гарантируют качества программ (и даже их работоспособности) и эффективности распространения.

В то же время при отсутствии информационного обмена между пользователями, при их разъединенности попавшие на рынок низкокачественные (а порой просто не работающие) программы имеют сбыт, практически не уступающий сбыту лучших образцов. Как с этим бороться? Традиционный путь — широкое неформальное объединение (некий «Информационный клуб пользователей УКНЦ»), собирающее и обнародующее оценки пользователей (своеобразный хит-парад программ).

Другой вариант — коммерческое объединение, с одной стороны, закупающее программы, с другой — распространяющее их. Членство в нем должно быть платным; на собранные деньги оно сможет покупать прошедшие первичное тестирование (на элементарную работоспособность) программы и распространять их среди своих членов (без дополнительной оплаты; все расходы объединения покрываются членскими взносами). Впоследствии, если отзывы пользователей хороши, объединение доплачивает автору программы, авансом оплачивает ему дальнейшую работу и т. п. Естественно, такое объединение должно быть не одно — во избежание монополизации и связанных с нею эксцессов.

Реализовать второй вариант малочисленной редакции журнала не под силу, а вот первый... Мы готовы публиковать хит-парад; ждем писем читателей с оценками программ.

Что же касается места для УКНЦ в «Клубе БК»... Этому компьютеру можно выделить место под собственный клуб. Только пишите! Ведь мы публикуем практически все материалы по УКНЦ, получаемые редакцией; жаль только, мало таких материалов...

## ДОСКА ОБЪЯВЛЕНИЙ

Всех обладателей «Синклер Спектрум», а также совместимых с ним компьютеров «КРОК», «Вариант-1», «Вариант-2», «Эльро-Юнион», желающих обмениваться прикладными и игровыми программами, приглашаем писать по адресу: 320106, Днепропетровск, просп. Героев, 21, кв. 4.

Г. и М. Мельники

Имею более 200 программ для «Синклера ZX». Хотел бы переписываться с пользователями этого компьютера.

620120, Свердловск,  
ул. Ленина, 24, кв. 40.  
А. В. Буденко

Имею множество игровых программ для «ZX Spectrum», готов ответить всем желающим.

352700, Майкоп,  
ул. Пушкина, 274, кв. 15.  
Т. Ш. Шу

Компьютерный центр при Новопсковской СШ № 2 ищет коллег — обладателей компьютеров «Атари» для обмена программами и текстовым материалом.

349670, Луганская обл.,  
Новопсков,  
ул. Карла Маркса, 307, СШ № 2.  
А. А. Васильев

Хотел бы переписываться с владельцами ПЭВМ ПК-01 «Львов» (Бейсик 2.0 MSX), ищу программы к этому компьютеру.

280025, Хмельницкий,  
ул. Олимпийская, 7, кв. 8.  
О. С. Пузырев

Всех желающих обменяться программами и схемотехническим расширением для «ПК-86» прошу писать по адресу: 252142, Киев, ул. Вернадского, 69а, кв. 33.  
Тел.: 444-17-59.

Геннадий Бондарь

Ищу программы для БПЭВМ «Вектор 06Ц». Сам имею 30 игровых программ (из них 12 в кодах).

400066, Волгоград,  
ул. Краснознаменная, 8, кв. 63.  
А. Л. Пономарев

Предлагаю владельцам ПК «Криста» и ПК «Микроша» организовать свой клуб. Идеи и предложения прошу присылать по адресу: 229070, Юрмала-16, ул. Талсу-шюссе, 31, корп. 3, кв. 38.

Сергей Григорьев

109

---

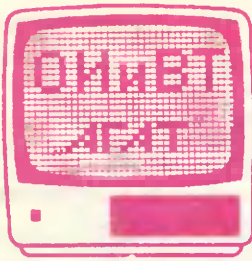
## Компьютерный салон «КРОНОС»

предлагает владельцам компьютеров  
«Атари» серии XL/XE

широкий выбор игровых и прикладных программ,  
литературу и другие услуги.

Заявки на каталог просим направлять по адресу:  
200032, Таллинн-32, а/я 2061.

---



## ПАКЕТ ПРОГРАММ

### «Основы информатики и вычислительной техники для ПЭВМ «Агат»

Пакет программ «Основы информатики и вычислительной техники» разработан научно-производственным объединением «Волгоградпрограммсистем».

На базе созданных программных средств и ПЭВМ «Агат» объединение поставляет в средние школы учебные классы — программно-технические комплексы (ПТК ОИ и ВТ—Агат-1).

Программы пакета предназначены для обучения учащихся основам информатики и вычислительной техники, факультативных занятий по информатике, проведения уроков по учебным предметам, а также решения отдельных практических задач на ПЭВМ «Агат».

Программные средства по назначению состоят из:

- программ поддержки курса информатики;
- обучающих программ;
- игровых программ;
- сервисных программ (утилит);
- общеобразовательных программ.

В программы поддержки курса информатики входят:

- практикум по алгоритмическому языку школьного учебника «Основы информатики и вычислительной техники» — «Алгоритмический практикум» (АП);
- практикум по графике, основанный на простейших графических командах, — «Начальный алгоритмический практикум графики»;
- демонстрационная модель работы процессора с интерпретатором команд процессора — «Алгоритм»;
- программа для обучения языку Бейсик — «Здравствуй, «Агат» (9 уроков);

Сервисные программы (10 утилит).

Общеобразовательные программы:

- система подготовки текстов (СПТ) «Агат—Автор»;
- система численного моделирования (СЧМ);
- модифицированная информационно-поисковая система на базе системы «АИСТ».

Пакет предназначен для ПЭВМ «Агат» 7-го исполнения с диском 830К байт и 140К байт.

Стоимость пакета — 950 рублей.

Стоимость оказания научно-технических услуг по освоению пакета — по прейскуранту.

- программа для изучения двух способов сортировки информации — «Сортировка»;

- тренажер работы с клавиатурой — «Клавиатурный тренажер» (КТ);

- исполнители в системе «Школьница» («Дежурик», «Машинист», «Муравей»).

Обучающие и контролирующие программы по математике, физике, химии, русскому языку, иностранным языкам, истории.

Наш адрес: 400016, Волгоград, ул. Прибалтийская, 1а, НПО «Волгоградпрограммсистем».

Телетайп: 266 «Поиск».

Телефоны для справок: 34-91-82, 79-06-92, 79-22-14.





## Ассоциация учителей информатики проводит

### Всесоюзный конкурс «Юный программист» (положение)

Ассоциация учителей информатики, Главное управление народного образования Красноярского крайисполкома, центр НТТМ «Спец» проводят конкурс «Юный программист» на лучшую программу, созданную школьниками для детей младшего возраста.

На конкурс принимаются клавиатурные тренажеры, демонстрационные, обучающие, контролирующие программы по предметам, исполнители, игровые программы и т. д., пригодные для использования в I—VI классах.

При разработке программ следует ориентироваться на школьные персональные ЭВМ: «Электроника БК-0010» в комплекте с

ДВК-2М (КУВТ-86), «Корвет», УКНЦ, «Ямаха», «Агат», «Правец» и др.

Победителей ждут призы: радиоприемники, фотоаппараты, часы, библиотечки литературы по информатике, персональные приглашения в летние школы по программированию.

Для участия в конкурсе до 1 февраля 1991 года необходимо прислать следующие материалы: заявку и подробный сценарий программы. Сценарий должен содержать основные идеи и покадровое описание программы.

Авторы отобранных по итогам конкурса сценариев будут приглашены в г. Красноярск в конце марта 1991 г. для участия в заключительном туре конкурса.

Материалы направлять по адресу: 660051, г. Красноярск, а/я 11533. Оргкомитет конкурса «Юный программист».

### ЗАЯВКА

Ф. И. О. \_\_\_\_\_  
Домашний адрес \_\_\_\_\_  
Номер и адрес школы (СПТУ) \_\_\_\_\_  
Класс \_\_\_\_\_  
Тип машины \_\_\_\_\_  
Язык программирования \_\_\_\_\_  
Название программы (кому и для чего предназначена) \_\_\_\_\_  
Объем программы \_\_\_\_\_  
Среднее время работы программы \_\_\_\_\_  
Ф. И. О. учителя информатики (руководителя кружка, преподавателя УПК), оказавшего помощь в работе над программой, обучении программированию \_\_\_\_\_  
\_\_\_\_\_

## Заочный конкурс педагогических программных средств (положение)

Задачи конкурса — активизация работы по созданию педагогических программных средств и использованию ЭВТ в обучении общеобразовательным предметам и профессиональной подготовке учащихся средних учебных заведений всех типов и привлечение учащихся к этой работе.

Организация. Соучредителями конкурса являются Ассоциация учителей информатики и Министерства народного образования ЧАССР. Для проведения конкурса создается экспертный совет, осуществляющий приемку и регистрацию конкурсных материалов и впоследствии их оценку.

Награждение победителей конкурса производится Ассоциацией учителей информатики и Министерством народного образования ЧАССР. Специальные призы могут быть вручены от имени и по поручению учредивших их организаций.

Тематика. На конкурс принимаются педагогические программы средств (ППС), созданные учащимися в индивидуальном порядке или коллективами учащихся для следующих ПЭВМ: «Ямаха», УКНЦ, «Корвет», «Агат», БК-0010, или для учебных комплексов (КУВТ, КВУ), состоящих из поименованных ПЭВМ.

На конкурс принимаются ППС по следующим тематическим направлениям:

преподавание общеобразовательных дисциплин в средних школах и ПТУ;

профессиональная подготовка в УПК и ПТУ;

профориентационная и внеклассная работа;

управление учреждениями народного образования;

Конкурсные материалы. ППС представляется на конкурс на магнитном носителе (дискете, кассете) и сопровождается документацией на бумаге, содержащей:

полное название программы (пакета) на

титальном листе;

состав авторского коллектива с указанием фамилий, имен, отчеств, места учебы, класса, почтового адреса для переписки;

справки с места учебы на всех членов авторского коллектива;

назначение и возможности программы (пакета);

условия применения (тип ЭВМ, необходимая периферия, используемая операционная система, язык программирования);

руководство по запуску и эксплуатации программы;

укрупненную блок-схему или блок-программу ППС;

ППС и сопроводительная документация в 2 экземплярах направляются до 1 марта 1991 г. в адрес оргкомитета конкурса.

Всем авторам и авторским коллективам, представившим ППС на конкурс, гарантируются их авторские права. Программы не тиражируются и не передаются на сторону, а по окончании конкурса возвращаются авторам вместе с магнитными носителями.

Итоги и награждение победителей. Победители конкурса определяются по типам ПЭВМ. Устанавливается соответственно 5 первых, 5 вторых и 5 третьих мест. Победители, занявшие первые места по своим типам ПЭВМ, награждаются 5 путевками в международный лагерь юных программистов летом 1991 г. Участники, занявшие второе и третье места, награждаются ценными призами МНО ЧАССР. По решению экспертного совета участники конкурса могут быть награждены поощрительными призами и дипломами.

По итогам конкурса Ассоциации учителей информатики, МНО ЧАССР и другие заинтересованные организации могут заключать с авторами договоры на приобретение ППС, на взаимной основе решая вопросы материального вознаграждения и авторского права.

Адрес оргкомитета: 428034, г. Чебоксары, а/я 99, Оргкомитет конкурса ППС.

# Международный семинар «Образование и информатика»

В июле 1990 г. в г. Минске в рамках II Международной олимпиады по информатике был проведен научно-методический семинар, в котором приняли участие члены жюри и руководители команд, а также специально приглашенные ученые и педагоги. Участники поделились опытом информатизации образования в своих странах, обменялись взглядами на содержание и характер подготовки школьников, студентов, педагогов.

Программу семинара начал В. А. Лабунув, выступивший с докладом «Проблемы информатизации в Белорусской ССР». Он критически оценил уровень информатизации как в БССР, так и в СССР в целом. По его мнению, в силу ряда причин информатика у нас, по сути дела, оказалась невостребованной — нет даже общепризнанной ее концепции. Сложившееся отставание от передовых в этой области стран весьма тревожно. Для его преодоления в докладе предлагалось сделать упор на региональные программы. Докладчик охарактеризовал программу «Информатика» в БССР и отметил, что подобные (свои для каждого региона) программы следовало бы иметь другим республикам, Москве и Ленинграду.

Представитель ЮНЕСКО Э. Джекобсон в своем выступлении говорил о необходимости разнообразного обмена опытом между странами в области информатизации образования. С его точки зрения, при обучении в массовой школе не нужен уклон в программирование, больше внимания следует уделять работе в лабораториях, изучению продвинутых программных средств (СУБД, машинная графика и т. д.), обмену обучающими программами.

А. Хартсвинер рассказал об информатизации образования в Нидерландах. Другой представитель этой страны Р. Кок представил в своем докладе один из вариантов LOGO, основанный на LISP. Касаясь школьного образования, он высказал мнение, что после небольшого вводного раздела информатика должна «уходить» в математику, физику, другие дисциплины и лишь в конце курса обучения завершаться рассмотрением обобщающих этот учебный предмет тем. Именно такой подход получает в Нидерландах государственную поддержку.

О компьютерном образовании в Таиланде, его сегодняшнем состоянии, проблемах и перспективах рассказал М. Канчит. К. А. Гъевров, охарактеризовав основные вехи компью-

теризации школьного образования в Болгарии, остановился на содержании и организации профессиональной подготовки школьников по специальности «Оператор-программист ЭВМ».

Информатизация образования в Великобритании — предмет выступления В. Джейма. Он отметил высокие темпы совершенствования компьютеров и программного обеспечения, эффективное использование которых требует от учителей все больших усилий. К сожалению, по оценке докладчика, количество компьютеров в школах намного превосходит число учителей, подготовленных к их использованию. Программные средства появляются с большим отставанием от потребностей учебного процесса, и техника успевает устареть еще до появления хороших программ.

Интересный исторический экскурс о сопоставлении нововведениям со стороны учителей сделал Т. Лингефьярд, выступивший с докладом «Информатика в гимназиях Швеции». От пересказа этого фрагмента трудно удержаться: еще в начале XVIII в. учителя были весьма обеспокоены потерей учащимися умений готовить древесную кору для вычислений в связи с введением в употребление грифельных досок. Позднее их крайнюю озабоченность вызвало широкое применение покупных, а не изготовленных лично самим студентом чернил, не так давно — применение шариковых ручек вместо перьевых. В настоящее время некоторые учителя полагают, что использование калькуляторов и персональных компьютеров будет препятствовать усвоению учащимися математики. На самом же деле, считает докладчик, учителям не следует пугаться вычислительной техники. Им необходимо изменить методы обучения той же математике, а в дальнейшем — пересмотреть и содержание учебных курсов. Будущих учителей обязательно надо учить с помощью компьютеров.

Представитель Марокко А. Хафави рассказал о преподавании информатики в средних учебных заведениях и подготовке педагогических кадров. Если раньше в Марокко преподавателей информатики готовили в основном из числа инженеров, то в настоящее время открыто много школ, где проходят переподготовку прошедшие отбор учителя математики и физики. Срок обучения — 2 года с полным отрывом от работы и сохранением содержания (сравните с принятой у

нас продолжительностью обучения в 2 недели!). По мнению докладчика, для последующего преподавания информатики именно учителя математики и физики подходят лучше всего.

И. Матиессен (Дания) говорил о подготовке специалистов по информатике в вузах. В условиях исторически сложившегося самоуправления в Копенгагенском университете программы обучения определяет совет, состоящий из равного числа преподавателей и студентов. Программы ежегодно меняются с учетом развития науки и техники и потребностей, возникающих на рынке интеллектуального труда. Принята в университете и своеобразная система экзаменов: своего рода аналог «бригадного метода». Любопытно, что все используемые учебники написаны на английском языке и на датский не переводятся!

Одним из наиболее активных участников семинара был В. А. Каймин. В его докладе и выступлениях речь шла о курсе информатики в средней школе, трактовке терминов «информатика» и «компьютерная грамотность», о подготовке учительских кадров. Проводя сравнительную характеристику действующих у нас в стране учебников и учебных пособий, докладчик выделил шесть разделов, которые, по его мнению, и составляют содержание каждой из этих книг. Отличия сводятся лишь к порядку следования и к объемам разделов. Вот их перечень: элементы программирования, элементы алгоритмизации, решение задач на ЭВМ, работа на ЭВМ, элементы информационных технологий, основы и применение ЭВМ. В. А. Каймин считает, что в учебном процессе могут применяться любая техника, любые языки программирования. Учитель должен иметь право на выбор учебника.

В. В. Прохоров свое выступление посвятил проблеме организации олимпиад по информатике. Он считает, что цели олимпиад не сводятся лишь к выявлению победителей. И школьники, и учителя рассматривают олимпиадные задачи как своего рода ориентир. Поэтому нельзя согласиться с устанавливающейся традицией сводить олимпиаду по информатике к соревнованию в решении задач «этудного программирования». Основные должны быть задачи с исходно неформальными постановками, моделирующие, возможные реальные ситуации. Школьники должны проявлять в первую очередь не виртуозность в работе с клавиатурой и не натасканность на типовые решения, а широту мышления. Отвечая на вопросы, он высказал мнение, что, возможно, настало время отдельно проводить олимпиады по различным направлениям информатики.

Н. А. Реуцкая, говоря о новых информа-

ционных технологиях в детском саду, отметила, что их введение требует нового содержания учебно-воспитательного процесса.

Г. С. Цейтин выступил с докладом «О профессионализме в программировании». Профессионализм в его понимании — это прежде всего умение учесть потребности пользователей, предусмотреть возможности модификации программного обеспечения, его переноса на другие типы ЭВМ и т. д. Докладчик выразил озабоченность ошибками, допускаемыми в подготовке программистов, слабой математической культурой многих из них.

А. А. Матюшкин-Герке говорил о том, что применение электронно-вычислительной техники в любой сфере человеческой деятельности может стать успешным лишь при условии, что сама эта деятельность будет кардинальным образом изменена как по содержанию, так и по методам решения возникающих в ней задач. В частности, компьютеризация образования требует полной перестройки учебного процесса и, прежде всего, содержания изучаемых дисциплин. Попытки же искусственного «внедрения» компьютеров в традиционные схемы школьных учебных предметов заведомо обречены на неуспех.

Важнейшей проблемой сегодняшнего дня докладчик считает подготовку учительских кадров. Практикуемая сейчас двухнедельная «курсовая учеба» совершенно недостаточна. Для более или менее удовлетворительной подготовки учителя необходимо 1-2 года с полным отрывом от работы! В таком обучении методологические (не путать с методическими!) аспекты должны выдвинуться на первый план. Без хорошо подготовленного учителя никакая, даже самая совершенная, техника не поможет делу. Великолепные учебники и программные комплексы в лучшем случае окажутся бесполезными.

Семинар проходил очень активно. И в докладах, и в дискуссиях практически отсутствовали банальности. Было высказано много нового, порой — неожиданного. В заключительном слове зам. председателя оргкомитета А. Н. Курбацкий отметил несомненную пользу семинара для работников народного образования. Различные точки зрения, споры, отсутствие единой концепции, естественные для такого мероприятия, дают здоровую пищу для размышлений.

Глубокой благодарности заслуживают организаторы семинара. Их усилия не пропали даром — семинар оказался успешным. Сожаление вызывает лишь отсутствие полных текстов докладов. Хотелось бы, чтобы впрямь такие тексты размножались если и не по новым, то хотя бы по старым информационным технологиям.

# Государственный реестр программ

Международный форум по проблемам интеллектуальной собственности в сфере программирования, проходивший в СССР в июне 1989 г., отметил в своей декларации, что ключевую роль в формировании рынка программ играет понятие интеллектуальной собственности на программы, означающие необходимость санкций собственника на копирование или использование программ.

В целях документирования и учета авторских прав разработчиков программ для ЭВМ, содействия защите этих прав, повышения качества и конкурентоспособности разрабатываемых программ, а также для содействия распространению наиболее эффективных и оригинальных из них Государственным комитетом СССР по вычислительной технике и информатике (ГКВТИ СССР) принято постановление о создании Государственного реестра СССР программ для ЭВМ.

Таким образом, появилась возможность государственной регистрации программ для ЭВМ, разработанных организациями, творческими коллективами, индивидуальными разработчиками.

Всемирная конвенция об авторском праве, к которой СССР присоединился в 1973 г., закрепила положение о том, что любая из стран-участниц вправе ввести соблюдение формальностей (таких, как регистрация, депонирование, уплата взносов и т. д.) в качестве условия охраны авторского права. Подавляющее большинство национальных законодательств требуют выполнения определенных формальностей для подтверждения авторских прав; в частности, в СССР государственная регистрация программ для ЭВМ

необходима для защиты прав на интеллектуальную собственность, а также для обеспечения гарантий равной охраны ее в любой из стран-участниц Всемирной конвенции об авторском праве.

Регистрацию, ведение реестра и выдачу регистрационных свидетельств осуществляет Центральная служба Государственного реестра СССР программ для ЭВМ (ЦСГ СССР программ для ЭВМ) при Специальном научно-производственном объединении «Алгоритм» (СНПО «Алгоритм») ГКВТИ СССР.

ЦСГ СССР программ для ЭВМ предлагает разработчикам информационные и консультационные услуги, годовое абонентское обслуживание.

ЦСГ СССР программ для ЭВМ обеспечивает всех заинтересованных лиц и организации нормативно-методическими документами Госреестра СССР в удобном виде (брошюры, дискеты); размещает заказы на доработку программной документации, проведение испытаний ПС; предлагает разнообразную информацию по направлениям деятельности Госреестра СССР. Ежеквартально ЦСГ СССР выпускает информационный бюллетень государственной регистрации программ для ЭВМ, содержащий краткие сведения о программах и их авторах.

ЦСГ СССР программ для ЭВМ принимает заявки от юридических лиц и отдельных граждан.

Наш телефон для справок: 203-79-97.

Наш адрес: 109068, Москва, ул. Велозаводская, 4, СНПО «Алгоритм», ЦСГ.

**Н. ПАЛЬМОВА**

# Первый раз в «пилотный» класс



1



2



3

Необычно проходила торжественная линейка-митинг (фото 1), посвященная началу учебного года и Дню знаний, в СШ № 1217 Москвы. В этот день в СССР открылась первая «пилотная» школа в рамках проекта Гособразования СССР и фирмы IBM (см.: «ИНФО», № 5, 1990). Присутствовали гости из США, преподаватели и ученые столицы, сотрудники КУДИЦ и УНМЦ НИТ, журналисты.

Открыла линейку директор СШ № 1217 Т. А. Понтяева: «Наш педагогический коллектив не стоит на месте. Мы экспериментируем, участвуем в различных проектах. И вот сегодня стали «пилотной» школой — в нашу жизнь пришел компьютер!» Значимость события подчеркнул Первый заместитель председателя Гособразования СССР Ф. И. Перегудов (фото 2): «Я твердо верю, что это лишь первый шаг, первая школа, за которой последуют тысячи. Наше сотрудничество будет развиваться на благо советского и американского народов и прежде всего детей. Надеюсь, в ближайшее время мы начнем обмениваться наиболее интересными программами по математике, физике, химии, биологии и другим предметам».

Представитель фирмы IBM Ж. Даванс (фото 3) пожелал успеха ученикам VIII и IX классов — они первыми войдут в дисплейный класс. Первоклассникам были вручены памятные сувениры.

После торжественной части присутствующих пригласили в дисплейный класс, оборудованный компьютерами фирмы IBM. Учителя «пилотных» школ, прошедшие летом подготовку для участия в проекте, провели открытый урок. Они продемонстрировали свои программные разработки по различным предметам, ответили на вопросы присутствующих (фото 4). Учитель физики «пилотной» школы (№ 1234) И. В. Топчий подчеркнула, что «участие в эксперименте открывает широкие горизонты для реализации собственных педагогических идей». Что может быть важнее для учителя!



## «ИНФО» В НОВОМ ГОДУ

В 1991 г. журнал будет продолжать знакомить учителей с практическими разработками для уроков информатики. В ближайшее время выйдут статьи о возможностях многостраничной графики, об организации компьютерного урока, о подходах к разработке педагогических сценариев.

Важной теме использования компьютеров в развитии и обучении младших школьников посвящены статьи Ю. А. Первина «Элементы музыкальной грамоты», В. А. Буцика «Методика формирования информационной культуры учащихся I—IV классов», Я. М. Марголиса, А. М. Иванова и Э. С. Баранкиной «Непрерывное обучение информатике в средней школе».

В рубрике «НИТ в дошкольном образовании» читатели увидят проект компьютерного детского сада, получают практические рекомендации по организации работы дошкольников за компьютером.

Сотрудники филиала ИАЭ им. И. В. Курчатова поделятся опытом организации и работы компьютерного детского клуба «Байтик», отмечающего в 1991 г. свой пятилетний юбилей.

Мы подробно расскажем о совместных проектах Гособразования СССР и фирмы IBM (США) «Пилотные школы» и «Дети-инвалиды».

В портфеле редакции есть интересные статьи для учителей физики, химии и математики.

Лицей, гимназия, вуз-школа-УПК... Эти многообещающие названия появились на некоторых московских школах. То, что за этим стоит,— предмет нашего журналистского исследования, с результатом которого мы по-

знакомим читателей.

В рубрике «КВТ» к публикации в первых номерах 1991 г. готовятся статьи В. С. Артамонова о дизассемблере для «Корвета», В. И. Литвинова о препроцессоре Бейсика для «Агата», продолжение цикла статей В. Г. Тищенко о сборке простейшего компьютера и работе с ним. Готовится также цикл статей о компьютере «Синклер Спектрум».

Если читатели проявят заинтересованность и активность, будет печататься «жит-парад» программ (рейтинг их качества и популярности). Для этого потребует большая организационная работа, но она должна окупиться: всевозрастающее количество программных продуктов появляется на рынке, и не всегда они работоспособны, не говоря уже об эффективности. Насущная задача — защита и консультирование пользователей.

В «Клубе БК» будут опубликованы описания электронного диска, расширения ОЗУ, энергонезависимого ОЗУ, программатора ППЗУ, методики защиты Бейсик-программ от просмотра и модификации. Продолжится публикация серии статей Ю. А. Зальцмана об ассемблере и архитектуре БК. В статье А. И. Ингоря приводится программа, выполняющая функции компилятора Бейсика: она переводит Бейсик-программу в последовательность шитых кодов. Ленинградец А. П. Гармашов предлагает аналог команды MERGE, позволяющий объединить программы, записанные в форме команды CSAVE.

Мы поделились с вами лишь частью наших замыслов; будут и сюрпризы! О них узнают те, кто стал нашим подписчиком. До встречи в 1991 г.!

## По страницам тематических планов издательств на 1991 г.

1. Антипов И. Н. Основы информатики и вычислительной техники. Методическое пособие для преподавателей техникумов. — М.: Высшая школа (вузы и техникумы). 1991 (I кв.). — 15 л. — 45 к., 25 000 экз.

В пособии рассматривается специфика формирования методики преподавания информатики в средних специальных учебных заведениях. Содержатся методические рекомендации и серия заданий для проведения лабораторных практикумов.

2. Басманов А. С. Персональные ЭВМ: бытовые, учебные, профессиональные: Справ. изд. — М.: Финансы и статистика, 1991 (III кв.). — 25 л. — 1 р. 60 к., 70 000 экз.

Для специалистов в области вычислительной техники, а также широкого круга работников всех отраслей народного хозяйства, использующих ПЭВМ в своей деятельности, студентов и аспирантов вузов.

3. Белоцерковский С. М. ЭВМ в науке, авиации, жизни. — М.: Машиностроение, 1991 (II кв.). — 16 л. — 1 р. 20 к., 40 000 экз.

Эта книга — увлекательный рассказ о становлении нового метода в науке — вычислительного эксперимента. Для широкого круга читателей.

4. Беляков М. И., Рабовер Ю. И., Фридман А. Л. Мобильная операционная система: Справочник. — М.: Радио и связь, 1991 (III кв.). — 15,5 л. — 80 к., 100 000 экз.

Материал книги максимально приближен к наиболее авторитетным проектам стандартов на интерфейсы UNIX, которые реализуются в последних версиях системы. Для прикладных и системных программистов.

5. Бурсиан Э. В. Задачи по физике для решения с помощью вычислительной техники: Учеб. пособие для студентов пед. ин-тов. — М.: Просвещение, 1991 (III кв.). — 15 л. — 70 к.

Автором подобраны задачи по всем разделам курса общей физики, которые невозможно или нерационально решать вручную. Приведены подобные решения для ПКМ на Бейсике, т. е. в книге рассматривается один из вариантов внедрения вычислительной техники в учебный процесс пединститутов. Для студентов, учителей физики старших классов средней школы.

6. Гольдштейн С. Л., Ковалев М. П., Рогович В. И. Знакомьтесь — компьютер. —

М.: Машиностроение, 1991 (I кв.). — 18 л. — 1 р. 30 к., 50 000 экз.

Рассмотрены устройство и вопросы эксплуатации персональных компьютеров, изложены основы алгоритмизации и программирования на Бейсике, Фокале, Фортране, приведены примеры задач, дан краткий терминологический словарь. Для широкого круга читателей.

7. Гранпьер Ж., Коттэ Р. Трехмерная графика на Турбо-Паскале: Пер. с фр. — М.: Машиностроение, 1991 (II кв.). — 20 л. — 1 р. 60 к., 50 000 экз.

Для специалистов, использующих САПР на базе ПЭВМ. Книга полезна в качестве учебного пособия по машинной графике, Паскалю и Турбо-Паскалю.

8. Грешилов А. А. Как принять наилучшее решение в реальных условиях. — М.: Радио и связь, 1991 (IV кв.). — 12,5 л. — 70 к., 40 000 экз.

В популярной форме рассматривается широкий круг задач принятия оптимальных решений в повседневной жизни. Излагаются методы решения этих задач, алгоритмы, ориентированные на применение персональных компьютеров. Для широкого круга читателей.

9. Грэхем Р., Кнут Д., Паташник О. Конкретная математика: Основания информатики: Пер. с англ. — М.: Мир, 1991 (II кв.). — 45 л. — 3 р. 50 к., 50 000 экз.

Пробором книги послужил раздел «Математическое введение» монографии Д. Кнута «Искусство программирования» (М.: Мир, 1976). В книге обучение общим методам оперирования формулами для дискретных объектов проводится на примерах решения конкретных задач. Приведены нестандартные упражнения с ответами. Для всех изучающих и применяющих дискретную математику и информатику.

10. Гузеев В. В. Обучение математике (с компьютерной поддержкой): Кн. для учителя. — М.: Просвещение, 1991 (IV кв.). — 7 л. — 20 к., 100 000 экз.

В книге рассматривается новая технология обучения, использующая возможности широкого применения компьютеров в режиме тренажера с целью реализации дифференцированного подхода к учащимся. Книга адресована учителю и может приме-



няться как при занятиях в компьютерном классе, так и в условиях безмашинного обучения.

11. Дейт К. Введение в системы баз данных. В 2 т.: Пер. с англ./Под ред. В. С. Минаева. — М.: Наука, 1991 (I кв.). — 40 л. — 3 р. 60 к.

Новый вариант известной книги, посвященной одному из стержневых направлений в информатике, отражает современное состояние бурно развивающейся области систем баз данных. Для научных работников в области информатики.

12. Джексон Г. Проектирование реляционных баз данных для использования с микроЭВМ: Пер. с англ. — М.: Мир, 1991 (IV кв.). — 11 л. — 65 к., 20 000 экз.

В книге содержится подробное описание возможностей СУБД dBASE III и R:BASE 5000 и демонстрируется их использование на примере одного и того же приложения. Для широкого круга пользователей.

13. Дидактические особенности построения учебных технических систем / Л. М. Терещенко, В. А. Трайнев, А. Л. Костин, В. В. Крынкин. — М.: Педагогика, 1991 (II кв.). — 15 л. — 1 р., 15 000 экз.

В книге дан критический анализ современного состояния оснащенности учебного процесса ТСО, рассмотрены методика внедрения компьютерной технологии в обучение, а также рациональные подходы к оптимальному построению технических учебных комплексов на базе персональных ЭВМ. Для специалистов в области педагогики.

14. Дрига И. И., Рах Г. И., Каплиев С. А. Технические средства обучения: Учеб. пособие для студентов пед. ин-тов и уч.-ся пед. уч.-щ. — 2-е изд., перераб. и доп. — М.: Просвещение, 1991 (III кв.). — 19 л. — 80 к.

Во второе издание книги включен новый раздел «Вычислительная техника и технические средства обработки информации и управления учебным процессом».

15. Дьяконов В. П. Язык программирования ЛОГО. — М.: Радио и связь, 1991 (I кв.). — 10 л. — 55 к., 30 000 экз.

Книга содержит более 250 процедур и функций для версий современных ПЭВМ. Особое внимание уделено применению ЛОГО для расчетов, построения графиков функций, фигур вращения, гистограмм. Для пользователей-непрофессионалов.

16. Зарецкий А. В., Труханов А. В. Энциклопедия профессора Фортрана: Кн. для учащихся. — М.: Просвещение, 1991 (IV кв.). — 12 л. — 1 р. 80 к., 100 000 экз.

Предлагаемая книга направлена на формирование компьютерной грамотности у младших школьников. Энциклопедия, являясь ярким, красочным изданием, привлечет к себе внимание детей, будет стимулировать их интерес к сложной технике.

17. Информатика и вычислительная техника: Учеб. пособие / В. В. Вьюхин, С. В. Кудымов, В. Г. Накрохин и др.; Под ред. В. В. Ларионова. — М.: Высшая школа (вузы и техникумы), 1991 (II кв.). — 10 л. — 35 к., 30 000 экз.

Пособие содержит основные сведения о компьютерах, технологии решения задач, типовом программном обеспечении ПЭВМ и ориентировано на использование комплекта учебной вычислительной техники «Корвет». Для студентов инженерно-педагогических специальностей вузов.

18. Калинин А. Г., Мацкевич И. В. Универсальные языки программирования. Семантический подход. — М.: Радио и связь, 1991 (III кв.). — 21 л. — 4 р. 50 к., 10 000 экз.

Обобщены и систематизированы требования к языкам программирования (ПЛ-1, Паскаль, Си, Ада и др.). Книга может служить методологической основой при разработке, оценке и освоении языков, при верификации программ аналитическими методами. Для научных работников, профессиональных программистов.

19. Карпиленко Е. В. Основы информатики и вычислительной техники в ПТУ: Метод. пособие. — М.: Высшая школа (профтехобразование), 1991 (I кв.). — 5 л. — 25 к., 50 000 экз.

Книга посвящена языку программирования Бейсик — Агат. Описаны основные функции дисковой операционной системы (ДОС). Лабораторные работы, иллюстрирующие книгу, стимулируют переход к практической работе на ЭВМ. Для преподавателей профтехучилищ. Может быть полезна школьникам, студентам и преподавателям вузов и техникумов.

20. Касаткин В. Н. Информатика, алгоритмы, ЭВМ: Кн. для учителя. — М.: Просвещение, 1991 (I кв.). — 14 л. — 60 к., 200 000 экз.

В основе методической дискуссии, посвященной проблемам педагогически эффективного преподавания информатики в школе, лежит более чем 25-летний опыт автора в преподавании основ информатики и вычислительной техники в крымской Малой академии наук «Искатель».

21. Кендал К., Кендаж Дж. Анализ и проектирование систем: Пер. с англ. — М.: Мир, 1991 (II кв.). — 56 л. — 4 р. 40 к., 20 000 экз.

В книге рассматриваются важнейшие аспекты системного анализа и проектирования систем компьютерной обработки информации, а также методы создания программного обеспечения с помощью ЭВМ. Для специалистов народного хозяйства и студентов вузов.

22. Котов Ю. В., Павлова А. А. Основы машинной графики: Учеб. пособие для студентов пед. ин-тов. — М.: Просвещение, 1991 (III кв.). — 15 л. — 75 к.

В книге нашли отражение как базовые темы машинной графики (технические средства программного обеспечения), так и специальные разделы. Для студентов педвузов, учителей информатики, черчения, изобразительного искусства и труда.

23. Кузнецов Е. Ю., Оксман В. М. Персональные компьютеры и программируемые микрокалькуляторы: Учеб. пособие. — М.: Высшая школа (профтехобразование), 1991 (I кв.). — 10 л. — 25 к., 100 000 экз.

В книге описаны основные типы отечественной техники и показаны возможности их применения в различных сферах учебной и трудовой деятельности. Для учащихся ПТУ и средних школ.

24. Мячев А. А., Степанов В. Н. Персональные ЭВМ и микроЭВМ. Основы организации: Справочник / Под ред. А. А. Мячева. — М.: Радио и связь, 1991 (I кв.). — 26 л. — 1 р. 60 к., 100 000 экз.

Содержатся общие сведения, классификация, терминология по техническим и программным средствам ПЭВМ и микроЭВМ, комплектам

учебной вычислительной техники. Для инженерно-технических работников.

25. Нортон П., Джорден Р. Работа с жестким диском IBM PC: Пер. с англ.— М.: Мир, 1991 (IV кв.).— 24 л.— 1 р. 50 к., 50 000 экз.

В книге освещаются практически все вопросы использования жесткого диска типа «Винчестер». Для широкого круга пользователей ПЭВМ типа IBM PC.

26. Основы информатики и вычислительной техники: Учеб. пособие / А. И. Громов, А. К. Зотов, В. И. Мансуров, М. Я. Софин.— М.: Изд-во УДН, 1991 (IV кв.).— 5 л.— 25 к., 25 000 экз.

В книге рассматриваются основные разделы информатики, принципы обработки информации, алгоритмизации вычислительных задач, составления программ на Бейсике, а также устройство ЭВМ. Изложение сопровождается простыми примерами и словарем на четырех языках с необходимой терминологией. Для советских и иностранных студентов, специалистов в области информатики и вычислительной техники.

27. Основы информатики и вычислительной техники: Учеб. пособие / Е. В. Алтухов, Л. А. Рыбалко, В. С. Савченко, А. И. Хохлов.— М.: Высшая школа (вузы и техникумы), 1991 (III кв.).— 15 л.— 50 к., 150 000 экз.

Для учащихся средних специальных учебных заведений, средних школ, широкого круга читателей.

28. Основы робототехники: Учеб. пособие для вузов / К. Д. Никитин, Н. В. Василенко, В. П. Пономарев, А. Ю. Смолин.— Томск: Радио и связь, 1991 (I кв.).— 25 л.— 1 р. 20 к., 20 000 экз.

Рассмотрены принципы построения и конструкция робототехнических систем. Для студентов вузов, изучающих общие вопросы робототехники.

29. Оттенсманн Дж. Табличный процессор QUATTRO: Справочное руководство для начинающих: Пер. с англ.— М.: Финансы и статистика, 1991 (II кв.).— 20 л.— 1 р. 70 к., 100 000 экз.

Описаны управление электронными таблицами, базой данных, возможности деловой графики, использование функций для прикладного статистического анализа и макрокоманд. Для широкого круга пользователей профессиональных ПК.

30. Персональный компьютер: диалог и программные средства: Учеб. пособие / Под ред. В. М. Матюшка.— М.: Изд-во УДН, 1991 (III кв.).— 20 л.— 2 р., 50 000 экз.

Излагаются принципы и приемы использования операционной системы MS DOS, текстовых, редакторов, табличных процессоров, системы баз данных РЕБУС, компиляторы языка dBASE-III—CLIPPE, средств деловой графики, интегрированных прикладных систем FRAMEWORK и SIMPHONY языков программирования Бейсик, Фортран-77, Турбо-Паскаль, а также программных средств для организации локальных сетей на базе ПЭВМ. Для студентов, преподавателей вузов и широкого круга пользователей ПЭВМ, желающих овладеть навыками общения с компьютером.

31. Первин Ю. А. Машинная графика на уроках информатики: Кн. для учителя.— М.: Просвещение, 1991 (IV кв.).— 11 л.— 90 к., 100 000 экз.

32. Персональный компьютер для всех. В 4 кн.: Практик. пособие / А. Я. Савельев, Б. А. Сазо-

нов, С. Э. Лукьянов; Под ред. А. Я. Савельева.— М.: Высшая школа (вузы и техникумы), 1991 (IV кв.).— 10 л.— 50 к., 150 000 экз.

В четырех книгах рассматриваются основные вопросы хранения и обработки информации на ЭВМ, подготовки и редактировании документов, создания и использования баз данных, описываются вычислительные и графические возможности ЭВМ. Для студентов вузов, учащихся школ и ПТУ, специалистов народного хозяйства, лиц, имеющих персональный компьютер в личном пользовании.

33. Поддубная Л. М., Шаньгин В. Ф. Мне нравится Паскаль.— М.: Радио и связь, 1990 (IV кв.).— 10,5 л.— 60 к., 100 000 экз.

Для школьников и учащихся ПТУ.

34. Пономарева К. В., Кузьмин Л. Г. Информационное обеспечение АСУ: Учеб. пособие — 2-е изд., перераб. и доп.— М.: Высшая школа (вузы и техникумы), 1991 (II кв.).— 15 л.— 45 к., 20 000 экз.

Во 2-ом издании (1-е — 1981 г.) излагаются основные направления развития АСУ и информационного обеспечения на современном этапе. Для учащихся средних специальных учебных заведений, обучающихся по специальности «Обработка информации в АСУ».

35. Практикум по методике преподавания математики: Учеб. пособие для студентов пед. ин-тов / Сост. В. И. Мишин.— М.: Просвещение, 1991 (IV кв.).— 15 л.— 75 к.

В книге даются рекомендации по использованию различных форм и методов обучения в соответствии с возрастными особенностями учащихся. Приводятся практические советы по использованию ЭВМ в процессе обучения математике. Для студентов пединститутов физико-математических факультетов.

36. Программирование микроЭВМ на языке Бейсик: Справочник / Е. С. Башмакова, И. М. Витенберг, А. Б. Либеров, А. Л. Пашков / Под ред. И. М. Витенберга.— М.: Радио и связь, 1991 (II кв.).— 18 л.— 1 р. 20 к., 150 000 экз.

Для инженеров, программистов и преподавателей.

37. Программное обеспечение микроЭВМ. В 2 кн.: Практик. пособие / Под ред. В. Ф. Шаньгина.— 2-е изд., перераб.— М.: Высшая школа (профтехобразование).— 150 000 экз., 1991 (IV кв.).

В этой серии из 2 книг, представляющих собой второе переработанное издание аналогичной серии в 1987—1988 гг., рассмотрены вопросы структуры и функционирования микроЭВМ, системного и прикладного программного обеспечения, описываются принципы программирования на языках Фокал, Бейсик, Фортран, Паскаль, ПЛ/М, рассматриваются вопросы применения ЭВМ в системах управления оборудованием, контроля, наладки и тестирования. Для системы профтехобразования. Могут использоваться студентами, учащимися школ и техникумов.

38. Романов Б. А., Кушниренко А. С. dBASEIV: руководство по изучению и использованию.— М.: Радио и связь, 1991 (IV кв.).— 17,5 л.— 1 р. 20 к., 30 000 экз.

Для программистов, создающих прикладные информационно-расчетные системы, а также поль-

зователей ПЭВМ, не имеющих навыков программирования.

39. САПР. В 4 кн.: Учеб. пособие / А. Н. Божко, Д. М. Жук, П. К. Кузьмин и др. / Под ред. И. П. Норенкова.— М.: Высшая школа (вузы и техникумы).— 50 000 экз. (I—II кв.).

В четырех книгах рассматриваются вопросы математического обеспечения САПР, создания и использования вычислительных и программно-методических комплексов, методы и средства интерактивного проектирования. В четвертой книге серии приводится словарь, содержащий около 800 терминов из области автоматизированного проектирования. Для студентов вузов.

40. Семенов В. А., Айдинян В. М., Липовой А. Д. Электронно-вычислительные машины: Учеб. пособие.— М.: Высшая школа (профтехобразование), 1991 (III кв.).— 18 л.— 75 к., 100 000 экз.

В книге приведены общие сведения, принципы построения и основы организации ЭВМ. Для учащихся ПТУ, средних школ.

41. Смирнов А. Д. Архитектура вычислительных систем: Учеб. пособие: Для вузов.— М.: Наука, 1991 (I кв.).— 20 л.— 90 к.

Рассматривается архитектура вычислительных машин и систем от супер-ЭВМ до мини- и персональных ЭВМ. Охватываются поколения машин от первого до пятого. Для студентов вузов.

42. Сова Дж. Структура концепций: Обработка информации: Пер. с англ.— М.: Наука, 1991 (IV кв.).— 30 л.— 4 р.

Книга посвящена вопросам понимания человеческого языка, накопления знаний и их извлечения. Излагаются проблемы моделирования на ЭВМ процессов, происходящих в мозгу человека. Для научных работников в области прикладной информатики и математики, особенно связанных с разработкой ЭВМ пятого поколения.

43. Тимофеев А. В. Информатика и компьютерный интеллект.— М.: Педагогика, 1991 (I кв.).— 5,6 л.— (Ученые — школьнику).— 35 к., 20 000 экз.

Из книги школьники узнают о развитии информатики, о новых возможностях диалога между человеком и компьютером, о компьютерах пятого поколения. Для старшеклассников.

44. Шах М. Имитационное моделирование: Методы и применение с помощью персональных компьютеров «Ай-би-эм»: Пер. с англ.— М.: Машиностроение, 1991 (III кв.).— 23 л.— 1 р. 80 к., 15 000 экз.

Для специалистов, занятых имитационным моделированием технических систем.

45. Шнейдеров В. С. Занимательная информатика.— М.: Машиностроение, 1991 (IV кв.).— 26 л.— 1 р. 80 к., 50 000 экз.

Веселая книжка для домашнего чтения предназначена для всех, кто интересуется информатикой.

46. Экономическая информатика и вычислительная техника в торговле: Учеб. для торг. вузов / С. И. Волков, А. А. Спиринов, А. Н. Вишняков, Б. А. Столяров.— М.: Экономика, 1991 (IV кв.).— 16 л.— 90 к., 30 000 экз.

Учебник ориентирован на приобретение практических навыков работы на ЭВМ, постановки задач, их алгоритмизации, использования готовых программных средств.

47. Электронно-вычислительные машины: В 8 кн. Практик. пособие / Под ред. А. Я. Савельева.— 2-е изд., перераб. и доп.— М.: Высшая школа (вузы и техникумы), 1991 (I—II кв.).— 150 000 экз.

В восьми книгах серии последовательно излагаются: введение в ЭВМ, основы информатики, технология подготовки задач для решения на ЭВМ, языки программирования Фортран-IV, Фортран-77, ПЛ/1 Паскаль, ПЛ/М, средства общения с ЭВМ. Одна из книг посвящена практикуму по программированию. Завершает серию книга по решению прикладных задач. Для студентов вузов. Может использоваться учащимися техникумов, преподавателями и практическими работниками.

Сост. В. Фрейман

## Новые журналы

Дефицит «компьютерной» литературы заметно уменьшается в последнее время благодаря усилиям различных СП. Они издают уже четыре журнала для пользователей ПЭВМ.

Самый старый из них — «Мир ПК». Он издается советско-американским предприятием «Информейшн Компьютер Энтерпрайз». Солидное, хорошо иллюстрированное издание, знакомящее с аппаратными средствами, программным обеспечением, дающее словарь новых терминов по вычислительной технике.

Стоимость журнала 5 рублей\*, выходит он 6 раз в год.

В газете «Компьютер Экспресс» (приложение к «Миру ПК», выходит 12 раз в год, стоимость одного экземпляра — 50 копеек) — информация для самого широкого круга читателей: программистов высокого класса, специалистов маркетинга, электрон-

\* Цены приведены по состоянию на 1990 г.

щиков, начинающих пользователей. Последним можно посоветовать ознакомиться со статьей Д. Роуэлла «Введение в файлы DOS», напечатанной в № 2, 3 за 1989 и 1, 2 за 1990 г. В доступной форме она знакомит с программами той операционной среды, в которой работают компьютеры фирмы IBM.

Более специальным является другое издание этого совместного предприятия — бюллетень «Сети» (6 раз в год, 2 рубля). Он дает информацию о разработке, производстве, внедрении и эксплуатации сетевых систем.

«Компьютерпресс» — издание советско-американского предприятия «Соваминко». Это ежемесячное обозрение зарубежной прессы, посвященное в первую очередь персональным компьютерам: их операционным системам, языкам программирования, интегрированным пакетам, текстовым редакторам и т. п., а также аппаратной части: процессорам, принтерам, мониторам и многому другому. Стоимость одного выпуска — 2 рубля 80 копеек.

«Интеркомпьютер» — информационный бюллетень совместного предприятия «Интерюнити» (цена одного экземпляра 3 рубля). Его авторы так формулируют свою стратегию: «Мы стараемся начинать «от печки», чтобы люди, впервые сталкивающиеся с рассматриваемой проблемой, но живо интересующиеся всем, что связано с компьютерами, могли расширить кругозор. В следующих номерах мы продолжаем начатую тему, углубляясь в детали и переходя на тот уровень проникновения в предмет, который соответствует требованиям специалистов. Авторы пытаются писать просто даже о самых сложных вещах».

Но где же можно это все купить? Записывайте адрес: 119034, Москва, Смоленский бульвар, 4, Московский городской центр информатики ГКВТИ СССР (тел. 246-96-59). Все эти издания, а также многие другие услуги в области информатики предоставят вам специалисты центра. Здесь можно приобрести обучающие, игровые, системные, прикладные программы для разных типов компьютеров, получить консультации, заказать разработку программного обеспечения, пройти курс «Основы информатики», арендовать машинное время, а также подписаться на оригинальное электронное издание «Информационный вестник МГЦИ», издаваемый авторским коллективом центра.

**Т. ТАТЕВЯН**

**От редакции.** К сожалению, 90 % материалов перечисленных изданий посвящены компьютерам, совместимым с IBM PC. Для владельцев более простых ПЭВМ единственным журналом остается «ИНФО».

А может быть, про IBM PC интересно почитать и владельцам БК? Если наши читатели пожелают, мы могли бы печатать рефераты наиболее интересных материалов. Пишите нам, сообщайте, какие темы вас интересуют.

А пока — несколько заметок «по мотивам» «Интеркомпьютера» № 1 за 1990 г.

## Цена программы

Что дешевле — двадцать программ на шести дискетах за сто рублей или десять на одной за двести? И то и другое может оказаться чересчур дорого. Но какова же «справедливая» цена программ?

Теоретически цена должна определяться как сумма затрат на разработку и тиражирование программы, а также некоторого дохода, деленная на число покупателей. Отсюда следует одно печальное следствие: поскольку в СССР в 100—1000 раз меньше персональных компьютеров, чем в США, а следовательно, во столько же раз меньше потенциальных покупателей программ для них, то, несмотря на вдесятеро меньшую зарплату советского программиста по сравнению с американским, цены на программы в СССР должны быть выше, чем в США (в 10—100 раз). Примерно такое соотношение и наблюдается на практике.

В ценообразовании на программный продукт есть еще множество интересных и почетных аспектов, и, если этот вопрос вас интересует, рекомендуем прочесть статью «Ценообразование на рынке программного обеспечения: теория и практика».

## Кошелек или информация

Сколько это будет стоить? А сколько будет стоить отсутствие этого?

Именно такова квалифицированная постановка вопроса защиты информации. Неквалифицированный же подход приводит либо к чрезмерным затратам, либо к информационным потерям, не искупаемым даже весьма крупными суммами. Яркий пример последнего типа — компьютерная кража летом этого года, когда вместе с ПЭВМ из медицинского института пропали дискеты, на которых были записаны результаты наблюдений за людьми, подвергшимися радиационному воздействию Чернобыля. Потерю такой научной информации деньгами не измеришь.

На данном примере ясно видно и то, что первая задача «защитников» информации — убедить пользователей в необходимости за-

щиты; вторая — синтезировать систему защиты. Для многих будет неожиданным, что в этой системе главную роль обычно играют не инженерные, а организационные аспекты.

Более полное раскрытие этой темы — в статье «Введение в защиту информации».

## Как это начиналось

1965 год, появление на рынке компьютера PDP-8... Первая ЭВМ ценой менее 20 тыс. долларов, предназначенная для управления процессами в реальном масштабе времени, стала родоначальницей нового класса вычислительных устройств — мини-ЭВМ. За PDP-8 последовали другие модели, в том числе и популярнейшая PDP-11, благодаря которой доходы производившей их фирмы DEC (США) превысили миллиард долларов.

Статья «Генеалогия семейства PDP» описывает историю развития этой линии ЭВМ.

## Переход количества в качество

Язык Си пользуется большой популярностью и имеет поэтому много диалектов. Но не верьте глазам своим! В статье «Язык Си ++ — потомок Си» особо подчеркивается, что «Си ++ не является диалектом языка Си или вариацией на его тему. Это по сути дела абсолютно новый язык программирования, основанный в значительной степени на языке Си и базирующийся на концепциях абстракции данных и объектно-ориентированного программирования. Поэтому язык Си ++ заслуживает особого внимания».

Если вы поверили — прочтите статью, ознакомьтесь с языком; если нет — все равно прочтите: а вдруг автор прав?

## Что такое AutoCAD?

Ну как вам сказать... Это AutoCAD.

В маленькой заметке трудно более полно охарактеризовать комплекс из нескольких десятков программ, способный работать на нескольких десятках моделей компьютеров и использовать несколько десятков периферийных устройств, поэтому мы сразу отсылаем интересующихся к статье «AutoCAD — стандарт de facto в конструкторском проектировании». Кстати, там приведены не только описания программ, но и их цены. Правда, преимущественно в фунтах стерлингов.

## Интимная жизнь

Речь здесь пойдет о вирусах, естественно, компьютерных. Их отличает от прочих программ способность к размножению. Впрочем, не все самопроизвольно размножающиеся программы являются вирусами: есть еще «червяки». Не исчерпываются вирусами и программы-«диверсанты», разрушающие данные: существуют еще «бомбы», «тройские кони»... Но не будем отвлекаться: сегодня в программе — «Анатомия и физиология вирусов». В этой статье подробно рассматривается функционирование вирусов, поражающих загрузочный сектор, а также вирусов других типов.

123

## Полезные советы

Вы включили свой IBM PC, а что с ним делать дальше, не знаете? Тогда вам поможет рубрика «Советы Бориса Александровича». Она предназначена для начинающих (во только сейчас!) пользователей.

Хорошая рубрика, но с одним недостатком: она занимает только страницу. Так что ее информацию вы используете за 10—15 минут, а затем будете месяц ждать следующей порции.

Не забудьте выключить компьютер!

## Оптическое считывание без оптики

Для обработки и передачи изображения с помощью компьютера первым делом его необходимо преобразовать в электрический сигнал. Для этого с помощью линз изображение фокусируется на светочувствительной полупроводниковой матрице, с которой закодированный сигнал поступает в процессор. Наиболее компактные оптические системы позволяют уменьшить расстояние между бумагой и фокусирующей линзой до 14 мм. Однако и это достижение конструкторам японской фирмы *Matsushita* показалось недостаточным. В результате на свет появилась принципиально новая считывающая головка для телефаксов, в которой нет ни одной линзы. Это позволило уменьшить расстояние между оптоэлектронным датчиком и бумагой всего до 100 микрон.

Уменьшение внутреннего пространства датчика до десятой доли миллиметра позволит дизайнерам делать некоторые модели телефаксов почти вдвое более тонкими.

## Компьютер для верующих

В современном деловом мире следить за временем нелегко всем, в том числе и верующим, особенно если часы молитвы приходится соблюдать с высокой точностью. Японская фирма *Excellent System Ltd.* решила помочь им, выпустив карманное устройство, в котором объединены электронные часы, радиоприемник и, разумеется, компьютер, который не только напомнит о времени очередной молитвы, но может и произнести ее. Если ко времени наступления молитвы включен радиоприемник, компьютер выключит его, а по окончании священнодействия вновь включит.

Прибор не подведет владельца ни дома, ни в путешествии: он

может работать в любой стране мира, достаточно установить местное время восхода и заката солнца, относительно которых и вычисляется время пяти молитв.

В предыдущей версии электронного служителя для воспроизведения молитвы использовался небольшой диктофон. В новом приборе голос священника оцифровывается и хранится в памяти компьютера, объем которой позволяет запомнить молитву продолжительностью до 6 минут 24 секунд. Это избавляет верующих от значительного числа проблем, связанных с непрочностью механических узлов диктофона и недолговечностью магнитной ленты.

## А или Б?

О широком распространении сканеров на Западе мы уже сообщали. А вот как они используются? Естественно, для ввода изображений; но зачастую это только первый шаг.

Если в ПЭВМ вводится изображение текста, встает задача его распознавания — ведь оно занимает много места в памяти (прикиньте, сколько, если сканируется машинописная страница с дискретностью пятьсот точек на дюйм, пусть даже с двумя градациями яркости), да и редактировать его очень трудно. Поэтому в ход идут программы распознавания текстов.

Они делятся на две основные группы: обучаемые программы и интеллектуальные. В обучаемых хранятся образцы начертаний символов, и распознавание очередного анализируемого символа осуществляется путем его сравнения с каждым из образцов до тех пор, пока не произойдет совпадение. Работают такие программы быстро и надежно, если качество введенного со сканера изображения достаточно высоко и если текст напечатан известным программой шрифтом. Их главный недостаток — трудоемкость ручного «обучения» распознаванию нового шрифта (а им даже известный

шрифт, введенный в новом масштабе — чуть мельче или чуть крупнее, — кажется новым).

«Интеллектуальные» программы при распознавании используют не начертания символов, а более абстрактные инварианты, называемые контурами. Для примера скажем, что контур буквы *O* характеризуется «чернотой» на периферии и «белизной» во внутренней области. Используются также известные статистические закономерности частоты появления букв, их сочетаемости, орфографические словари (для уточнения неуверенно распознаваемого символа, если остальные символы в слове опознаны) и т. п. К сожалению, шрифты с необычными начертаниями такие программы распознавать не могут. Кроме того, для их работы нужны очень высококачественные изображения текста (оцифровка с плотностью 10—30 точек на миллиметр), высокое быстродействие процессора и большие объемы памяти. Тем не менее они широко используются там, где приходится вводить в ЭВМ большие объемы текстов.

## Шесть процессоров на чипе

Известная американская фирма *Intel* обещает выпустить к концу века (а ждать этого уже недолго!) микросхему *I786*. Микропроцессором ее назвать будет уже нельзя — процессоров в ней будет шесть штук: четыре для обработки целых чисел (они будут работать параллельно) и два векторных для построения изображений, в том числе объемных. Суммарное их быстродействие будет 2000000000 команд в секунду, тактовая частота — 250 МГц.

Однако основную часть общей площади в 6,25 см<sup>2</sup> занимают не процессоры, а сверхоперативная (кэш) память емкостью в два мегабайта.

Итак, в 2000 г. — 100 млн. транзисторов на квадратном дойме... Пожелаем фирме успехов.

Информатику необходимо сохранить	5	<i>Каптелин В., Дроздова Т.</i> Опыт обучения работе с текстовым редактором	4
<b>Общие вопросы</b>		<i>Кузнецов В.</i> Как войти в информатику?	5
<i>Болтянский В., Рубцов В.</i> Игровые компьютерные среды учебного назначения	5	<i>Лебедев Г.</i> О новом учебнике информатики	5
<i>Брусилоский П.</i> Языки для обучения основам программирования	5	<i>Ляхович В.</i> Методика составления алгоритмов	1
<i>Бургин М., Степенко Г.</i> Информационный поиск и компьютерная грамотность	2	<i>Манзюра В., Шмелев Д.</i> Моделирование правил орфографии	3
<i>Когдов Н., Цевенков Ю.</i> Центры новых информационных технологий: цели и задачи	1	<i>Руденко В.</i> Имитационное моделирование учебных алгоритмов	5
Концепция информатизации образования	4	<i>Сапир М.</i> Курс ОИВТ — каким мы его видим	5
<i>Мошкова И., Семибратов А.</i> Компьютеризация производственного обучения	1	<i>Степанов М.</i> Система индивидуальных заданий для практических занятий по информатике	5
<i>Невуева Л., Сергеева Т.</i> О перспективных тенденциях разработки педагогических программных средств	2	<i>Сулима-Самойлов К.</i> Опорные конспекты в курсе ОИВТ	3
<i>Перегудов Ф.</i> Системная деятельность и образование	5	<i>Урнов В., Зубченко А.</i> Переходные среды в предмете «Информатика»	4
<i>Садовникова Н., Ратинский М.</i> «Информатика» — базовый предмет в школах РСФСР	1	<b>Кабинет ВТ</b>	
<i>Уваров А.</i> Информатика в школе: вчера, сегодня, завтра	4	<i>Аляев Ю., Овчинников В.</i> Защита НГМД	3
<b>Методика обучения</b>		<i>Артамонов В.</i> Музыка КУВТ «Корвет»	6
<i>Авербух А., Гисин В., Зайдельман Я.</i> Методические рекомендации к учебнику информатики	4	<i>Белова Л., Белов Ю.</i> Изображение поверхностей	2
<i>Акимова Г., Дега В.</i> Демонстраторы помогают учиться программированию	4	<i>Боксер О., Васильченко А., Прияткин А.</i> Реакциометрические обследования на учебных ПЭВМ	1
<i>Ватник Н., Куприянов С., Лобанова Л., Пахомов Ю.</i> Опыт разработки и использования УАОС	3, 4	<i>Бондаровская В., Пovyакель Н.</i> Психолого-эргономическое обеспечение программных средств	5
<i>Гейн А., Линецкий Е., Сапир М., Шолохович В.</i> Информатика: циклы и вспомогательные алгоритмы	6	<i>Борисов В., Поляков С.</i> Использование языков высокого уровня на КУВТ-86. Возвращаясь к теме...	1
<i>Гольцман М., Дуванов А., Зайдельман Я., Первин Ю.</i> Алгоритмические этюды	6	Временные санитарно-гигиенические нормы и правила	2
<i>Гольцман М., Дуванов А., Зайдельман Я., Первин Ю.</i> Арифметические исполнители	3, 4	<i>Галицкий М.</i> Коллектив исполнителей	3
<i>Гольцман М., Дуванов А., Зайдельман Я., Первин Ю.</i> Исполнители	1	<i>Гельгищева Е., Селехова Г.</i> Как предупредить переутомление при работе с видеотерминалом?	4
<i>Дворина М., Довгялло А., Ивахненко Э., Коберник Е., Маргулис Е., Стрижак А., Чигорев А.</i> Деятельность младших школьников в компьютерной игровой учебной среде	3	<i>Гинзбург М.</i> Графика БК на уроке	3
<i>Казиев В.</i> Математическое моделирование и вычислительный эксперимент	6	<i>Горбунов В.</i> Сравнение текстов по моделям позиций слов	4
<i>Каймин В.</i> Курс информатики: состояние, методика и перспективы	1	<i>Григорьев С.</i> Графические средства системы Пролог-Д	6
<i>Каймин В., Григорьев С.</i> Пролог в школьной информатике	4	<i>Григорьев С.</i> Программирование на Прологе-Д	5
	2	<i>Григорьев С.</i> Работа системы Пролог-Д	4
	5	<i>Гуткин М., Иванов А., Новосельцев С., Христовский С.</i> Учебные персональные ЭВМ	6
	6	<i>Добряков В.</i> Эффектно и эффективно	5
	5	<i>Еремин Е.</i> Русскоязычная диагностика ошибок на «Ямахе»	2
	6	<i>Еремин Е.</i> Улучшение Турбо-Паскаля для «Ямахи»	6
	6	Итоги конкурса	3
	2	<i>Караваяев А.</i> Цветной демонстрационный монитор	4

Коробицын Д. Изучаем Бейсик за полчаса  
 Коцюба Е. Сделайте АОС сами!  
 Кочергин А. Задача о слухах  
 Кузнецов А. Графический планшет  
 КуМир и другие  
 Лепин А., Лепин Л., Трейкале Т. «Редактор  
 экрана» — инструмент разработчика  
 Матвеев А. «Микроша»: 48К байт  
 Настасенко В. Игровой пульт  
 Новости рынка  
 Новые пособия  
 О доплате за заведование кабинетом вы-  
 числительной техники и за работающие  
 компьютеры  
 Панков П. Глазомерные задачи  
 Положение о кабинете вычислительной тех-  
 ники  
 Программа печатает сама себя (компьютер-  
 ный аукцион)  
 Прохоров В., Кондин С. «Микроша» в  
 компьютерном обучении  
 Пустовойтой С. Об одной особенности  
 КУВТ-86  
 Решение задач с помощью Бейсика  
 Сабиров Р. Простой клавиатурный тре-  
 нажер  
 Тищенко В. Наладка простейшего компью-  
 тера  
 Тищенко В. Простейший компьютер  
 Файловый монитор FILMON для ПЭВМ  
 «Микроша»  
 Ходаков П. Спасение файлов на «Агате»  
 Шкрбец Ю. Неожиданное применение  
 Ядгаров Х. Программа — тренер

#### Клуб БК

Авсеев В., Авсеев А. Особенности трансля-  
 тора с языка Бейсик для БК-0010.01  
 Антонов Д. Увеличение тактовой частоты  
 Барсуков А. Убыстрение поиска файла  
 Бочаров А. Хитрости Бейсика БК  
 Бултко В. Вмешательство в работу про-  
 граммы  
 Бунцельман А. Программа «Музыкальная  
 заставка»  
 Василенко В. Спасение программ при за-  
 висании  
 Гвоздев С., Эрнестсонс Г. Еще раз о пла-  
 вающей арифметике Фокала  
 Зальцман Ю. Архитектура и ассемблер  
 БК-0010  
 Ивашильников С. Многоголосие на БК-0010  
 Иггорь А. Использование квазиподпрограмм  
 Калейдоскоп  
 Канивец И. Знаете ли вы, что...  
 Комаров С. Строки вместо матриц  
 Кузнецов А. Еще раз о восьмицветном  
 БК  
 Кумандин С. Новый подход к построению  
 баз данных на БК-0010  
 Ларкин М. Экономия памяти БК-0010.01  
 Луцкевич Ю. Подключение БК-0010 к те-  
 левизорам 4УСЦТ  
 Панченков И. Приручение БК  
 Смирнов С. Восьмицветная приставка  
 Советы и наблюдения

1 Тереховский А. Универсальный каталогизи-  
 затор  
 1 Умников Е. Спрайты для БК  
 5 Чирков П. Подключение БК к телевизо-  
 рам ЗУСЦТ  
 3 Яковлев В. Отсчет времени на БК-0010.01  
 5

#### Педагогический опыт

Брудман Л., Распопов В. Задачи для ПМК  
 Воронова Л. Преподавание алгоритмических  
 языков  
 Гальвас Э., Фаст В. Физические задачи в  
 курсе информатики  
 Гварамия Г., Маргвелашвили И., Мосиашви-  
 ли Л. Опыт разработки компьютерных  
 учебных пособий по физике  
 Голицына И., Нарыкова И. Компьютер на  
 уроке физики  
 Демакова И., Годунова Е., Борисова В.,  
 Жамкочьян М., Матвеева Т., Олейник М.,  
 Петрова И., Пронина Е., Пятунин В.  
 На подступах к компьютерной педаго-  
 гике  
 Епина М. Об организации взаимодействия  
 вуза и школы  
 Житков Г., Прохорова М. Комплексное  
 применение средств ВТ при обучении по  
 курсу TOP  
 Каравиева Э. Нетрадиционные формы обу-  
 чения на уроках информатики  
 Караев Ж., Рах С. Использование компью-  
 теров в казахской школе  
 Киселев Б., Леонова Н. Педагогика сотrud-  
 ничества и компьютеризация учебного  
 процесса  
 Кузьмин Ю., Рибуде И., Кондратенко С.  
 Использование ЭВМ при обучении русско-  
 му языку  
 Лазарева Е. Помогает центр информатики  
 Маргулис Е., Косов Ю., Мележик Ю.  
 Компьютерные игры в обучении  
 Меш Г. ПЭВМ в помощь учителю хи-  
 мии  
 Михеев Ю. Программное сопровождение  
 курса планиметрии  
 Санжаров Л., Финьков А. Компьютер, цвет,  
 язык  
 Семionenков М. Обучение школьников про-  
 граммированию  
 Сиваковский А. Педагогические программ-  
 ные средства: объектно-ориентированный  
 подход  
 Фадеев С. Компьютерные игры и орфогра-  
 фия  
 Фадеев С. Сказка ложь, да в ней намек  
 Фадеев С., Фадеева А. От диалога с компью-  
 тером — к диалогу с больным  
 Шрайбер Г. Предлагаю программы для  
 ПМК  
 Штернберг Л. Уроки с программируемыми  
 микрокалькуляторами

#### Внеклассная работа

Белоножка О., Юдилевич А. Психологиче-  
 ская диагностика детей и компьютер



Зайцев А. Спутники, компьютеры, образование	1	Ищем спонсоров	6
Кельман Е., Литвинский В., Мельник С. Опыт внеклассной работы на СМ-4	3	Как быть?	2
Кузнецов С., Расповов В. Программирование логических игр	3	Как изучать тему	4
Кузнецов С., Расповов В. Формирование навыков параметрического программирования	3	Личность учителя	1
Ландо С., Манакова О. Компьютер в летнем лагере	1	Настоятельно просим	4
Манзюра В. Компьютер и ударение	1	О программах для «Корвета»	6
Очков В. Рассказ о трех Э	5	Письмо и комментарий	6
Пахомова Н. Юношеское научное общество школьной информатики	2	Чтобы не бегать по классу	6
Переход И., Касаткин В. Комбинаторные задачи	3	Неделя информатики	3
Переход И., Касаткин В. Моделирование и визуализация в программах учебной ориентации	6	Почтовый ящик	1, 3
Свердлов А. Моделирование многомерных процессов	1	Предлагаем всем	4
Сокол И., Шип С. Компьютер — помощник в музыкальном обучении	5	Сердитое письмо	1
«Уральские» задачи	2	«Сура ПК 8000» в школе	5
Шернаев Н. Взгляд на олимпиаду	2	Уважаемые коллеги!	4
	4	Школьной информатике быть	5
	5		
<b>Молодежная инициатива</b>	1, 2, 4	<b>Репортаж номера</b>	
		Ярмарка-2000	2
<b>ЭВМ в народном хозяйстве</b>		<b>Интервью номера</b>	
Диагноз ставит компьютер	3	«Альтернатива» существует	3
		<b>Точка зрения</b>	
<b>Зарубежный опыт</b>		Виленкин Н., Ратинский М., Ин А. Информатика или программика?	2
Борк А. Компьютеры в обучении: чему учит история	5	Лукьянов А., Салимжанов Р., Ситников В. О модели выпускника средней школы по курсу ОИВТ	4
Компьютер для университетов	5	Прохоров В. Уроки олимпиады	2
Нокс Дж. Что могут дать компьютеры педагогике: Взгляд из американской школы	1		
Обучающая система DIL	2	<b>Информация</b>	
Уваров А. ЭВМ в американской школе сегодня	2	Ассоциация создана	4
		Ассоциация учителей информатики проводит конкурсы	6
<b>НИТ в дошкольном образовании</b>		Банк педагогических данных МОПИ им. Н. К. Крупской	5
Глушкова Е., Леонова Л., Сазанюк З., Степанова М. Гигиенические требования к занятиям для дошкольников	6	Болгаро-советский семинар	1
Горвиц Ю. Развивающие игровые программы для дошкольников	4	Быть ли единой системе?	3
Зворыгина Е. Педагогические подходы к компьютерным играм для дошкольников	6	Всероссийская конференция «Компьютер в школе и педвузе»	3
Новоселова С. Проблемы информатизации дошкольного образования	2	Второй пленум Общесоюзного научно-методического совета информатизации образования	1
Приглашаем к сотрудничеству	2	Государственный реестр программ	6
		Дубна-89	4
<b>Педагогические кадры</b>		Ждем продолжения	1
Проблемы информатизации: взгляд из педвуза	4	«ИНФО», в новом году	6
		Конкурс школьных химических программ на ЭВМ	3
<b>Нам пишут</b>		Конференция в г. Ульяновске	3
Вам интересно?	1	Лебедева М. Международное совещание в Гатчине	1
Доска объявлений	5	Международный конгресс «Образование и информатика»	3
Доступно уже сегодня	1	Международный семинар «Образование и информатика»	6
Извините, Елена Константиновна!	2	Первый раз в «пилотный» класс	6
		Пленум УМО по информатике	3
		Проект «Пилотные школы»	5
		Семинар в Узбекистане	3
		Семинар на ВДНХ	4
		Совещание на заводе «Экситон»: перспективы БК	2

Третья Ленинградская научно-методическая конференция

## Книги

Алгебра на дисплее  
Алгоритмический язык — примеры и задачи  
Методички из США  
Через задачи — к программированию

## Библиография

Напечатано в 1990 г.

	Новые журналы	6
3	По страницам тематических планов издательств на 1991 г.	6
	<b>Веселый урок</b>	
3		
1		
3	Жаворонков В. Барскиана, или Новые операторы языков программирования	3
1	Жиров А. Общедоступный словарь по вычислительной технике	1
6	Тихонов В. Теория ошибок	2

## Гибкий диск — 20М байт

128

История развивается по спирали. Во внешней памяти ЭВМ основную роль играли магнитные барабаны (весьма жесткие), затем появились магнитные ленты (совершенно гибкие), потом диски, потом гибкие диски. Предпоследнее явление — жесткие «винчестерские» диски. Последнее — диски «Бернулли».

В соответствии с логикой развития они гибкие — настолько, что завихрения воздуха между быстро вращающимся диском и магнитной головкой в полном соответствии с уравнением Бернулли подтягивают к головке находящейся под ней участок диска. Точно рассчитанные аэродинамические эффекты поддерживают оптимальное расстояние между головкой и диском; это позволяет использовать высококачественные головки, не опасаясь их повреждения магнитным слоем.

Накопитель такой конструкции более надежен и долговечен, чем винчестер, имеет сравнимую емкость (40М байт на 5,25-дюймовом диске) и, что немаловажно, позволяет легко менять носители информации.

## ЧТО МОЖЕТ ЭВМ

Есть и другие способы увеличить емкость гибкого диска. Например, фирма Insite Peripherals в своих магнитно-оптических накопителях использует стандартные 3,5-дюймовые диски (такие же, как в школьных «Ямахах»), на поверхности которых лазером нанесены концентрические кольца разметки. Оптическая система накопителя отслеживает эту разметку, благодаря чему резко увеличивается точность установки магнитной головки и появляется возможность сделать информационные дорожки более узкими и расположить их ближе друг к другу. Стандартный диск имеет от 45 до 133 дорожек на дюйм, размеченный лазером — до 1250, что позволяет записать на него более 20М байт данных.

Компания Brier Technology для более точного позиционирования головки использует двухслойные магнитные диски. На верхнем слое записывается информация, а на нижнем (при изготовлении) — нестираемый синхросигнал позиционирования. Кроме того, на

внешних дорожках предусмотрена запись больших объемов информации, чем на внутренних (ведь первые длиннее!), что не делается в стандартных дисководов. Результат — более 40М байт на диске диаметром 3,5 дюйма.

## Нас подслушивают?

Нередко, разговаривая по телефону, вы произносите: «Это не телефонный разговор», и откладываете его до личной встречи. Японская фирма Advantec Co., Ltd. выпускает прибор, который может обезопасить телефонную линию от нежелательных «собеседников» и ускорить решение многих личных и производственных проблем.

Компактный ящик (6×16×20 см), подключенный параллельно телефону, постоянно «прислушивается» к вашей линии. Как только в нее включается перерывчик, компьютер фиксирует вмешательство и подает световой и звуковой предупреждающие сигналы. Заметив это, вы должны решить, с кем и на какую тему теперь лучше продолжать беседу. Любой желающий может установить электронный сторож всего за 5 минут.

*Уважаемые читатели,  
поздравляем вас с Новым 1991 годом!  
Желаем успехов в нашем общем нелегком деле  
— компьютеризации образования.*

**Волжский завод ЭВТ  
предлагает  
персональные ЭВМ «Агат»!**

*Если вы хотите* создать в своем учебном заведении компьютерный класс, организовать разумный и полезный досуг для молодежи, подготовить учащихся к производственной деятельности в условиях широкого применения вычислительных машин, обращайтесь на Волжский завод электронно-вычислительной техники:  
**мы производим и продаем персональные ЭВМ  
«Агат-7» и «Агат-9».**

***Основное достоинство этих компьютеров,***

наряду с простотой обслуживания и сравнительно невысокой стоимостью, возможность обеспечить непрерывное, многоступенчатое и индивидуальное обучение для всех возрастных категорий учащихся.

***Именно для «Агата»***

разработаны наиболее удачные и широко распространенные программы и методики обучения. Полная автономность каждого компьютера делает кабинеты учебной ВТ, оснащенные ПЭВМ «Агат», особенно надежными. Вместе с тем при необходимости потребитель может заказать и использовать локальную сеть передачи данных.

***Важная особенность:***

вместе с купленным «Агатом» вы получите комплект программных средств — мощный инструмент обработки данных, позволяющий не только учащемуся, но и экономисту, инженеру, бухгалтеру, нормировщику, секретарю применять этот компьютер в своей деятельности. Завод ЭВТ взял на себя расходы по поставке учебным заведениям программ в счет общей стоимости ПЭВМ.

***Основные характеристики наших «Агатов»:***

система программирования (входной язык) — Бейсик, а также Рапира, Паскаль; системный модуль с ОЗУ емкостью 64К байт для «Агат-7» и 128К байт для «Агат-9»; накопители на гибких магнитных дисках ЕС-5088.02 емкостью 140К байт или ЕС-5323.01 емкостью 1М байт;

блок клавиатуры со стандартным расположением клавиш русского и латинского алфавитов, со специальными функциональными клавишами; профессиональный монитор монохромного изображения «Электроника-МС-6105» с разрешающей способностью 256×256 точек («Агат-7») или 512×256 точек («Агат-9») и размером алфавитно-цифрового поля 25 строк по 80 знаков; печатающее устройство СМ-6337;

потребляемая мощность от сети 220 В не более 135 Вт; средняя наработка на отказ — 3000 часов, средний срок службы — не менее 10 лет при условии проведения профилактических работ.

**Если вы не умеете работать с этим компьютером, мы поможем освоить его.**

Никаких хлопот с получением:

централизованную доставку, ввод в эксплуатацию, гарантийный ремонт и техническое обслуживание выполняют наше предприятие и объединения вычислительной техники и информатики, расположенные практически во всех регионах страны.

С целью обеспечения надежной работы наших ПЭВМ мы готовы предложить комплекты ЗИП в различном наполнении.

Имея их, вы сможете сами до прибытия специалистов восстановить «Агат» заменой отказавшей ячейки.

***Итак, мы ждем ваших заказов!***

Наш адрес: 404110, г. Волжский Волгоградской обл.,  
Волжский завод электронно-вычислительной техники.

Телетайп: 310112, «Радио».

Телефоны для коммерческих справок: 7-68-92, 7-34-43 (междугородный код 84459).

Цена 60 коп.  
70423

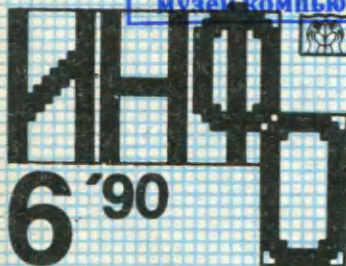
*Душин 17 БИД*

OldPC.ru

7003

музей компьютеров

*(2)*



6 '90

**ИНФОРМАТИКА  
И ОБРАЗОВАНИЕ**

