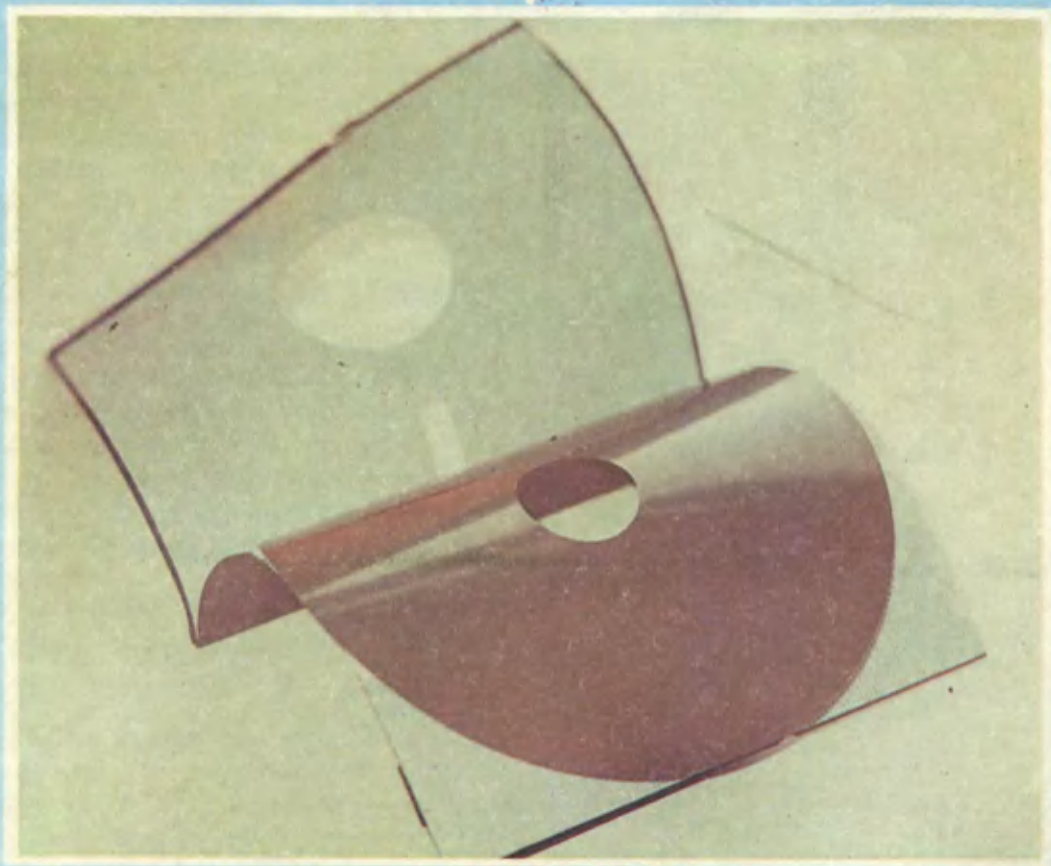
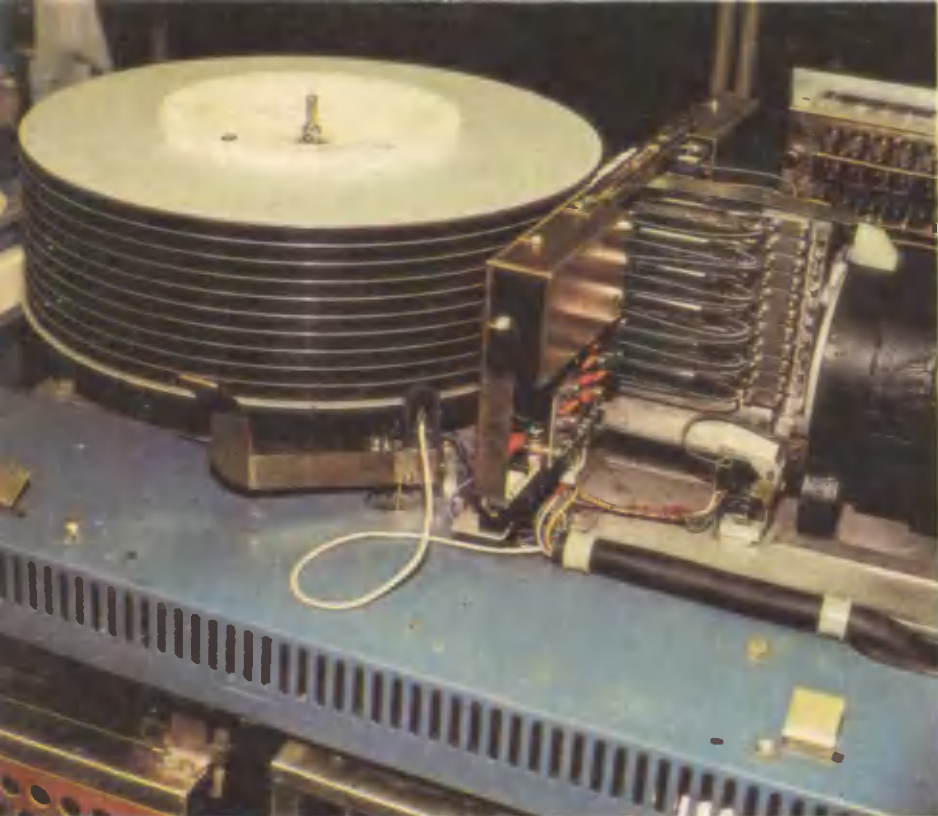


ISSN 0234-0453

ИНФОРМАТИКА И ОБРАЗОВАНИЕ

5 1988

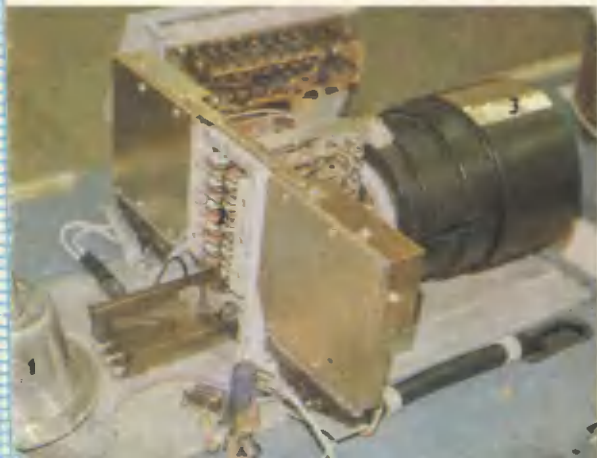




Накопитель на диск-пакетах. Для большей наглядности на него установлен диск-пакет, что, конечно, при снятом кожухе не делается. Справа от диск-пакета виден блок магнитных головок.

Накопитель на диск-пакетах в рабочем состоянии.

Накопитель на диск-пакетах со снятым кожухом: 1 — шпиндель, на который устанавливается диск-пакет, 2 — магнитные головки, 3 — электромагнитная катушка, управляющая положением головок.



На 1—3-ей страницах обложки — иллюстрации к статье А. Архангельского «Мир ЭВМ».

На первой странице обложки: чтобы показать читателям дискету «изнутри», мы разрезали конверт отслужившего свое диска.



ИНФОРМАТИКА И ОБРАЗОВАНИЕ

5 '88
сентябрь —
октябрь

Содержание

Общие вопросы

Белешапка В. Мир как информационная структура 3

Методика обучения

Дуванов А., Гольцман М. «Черные ящики» 10

Копылов Г. Из рукописи конкурсного учебника 19

Решения к «Первым шагам» 33

КВТ

Архангельский А. Мир ЭВМ 35

Зенкин А., Зенкин Г. Обучающая система-тренажер 43

Штернберг Л. Умеет ли считать ваш микрокалькулятор? 51

Подъяблонский Ю., Качаев С. Класс БК без ДВК 55

Варакс С., Яновский С. Объединение БК-0010 в сеть 55

О пересылке информации на принтер в КУВТ-86 58

«Микроша» 59

Педагогический опыт

Викентьев Л., Козлов О. Что дает учебный язык с русского лексикой 60

Вейцман В. Старт в информатику 65

Лунина Н. Пакет программ "LEARN ENGLISH" 72

Токарская И. Информатика и повышение квалификации 74

Якубов А. О компьютерном всеобуче в Чечено-Ингушской АССР 80

Внеклассная работа

Пахомова Н., Шрейбер Б. База данных школы 82

Катков В. Студенческие олимпиады 84

Молодежная инициатива 87

Точка зрения

Васильченко А., Федотов В. Об учебной ЭВМ 99

Христочевский С. Перспективный компьютер для сферы образования 101

Матюшкин-Герке А. Каким же быть школьному компьютеру? 103

Педагогические кадры

Кузьмин И., Кочуров В., Кочурова О. Реальный уровень компьютерной грамотности педагогических кадров: ничто или нечто? 106
Кузнецов Э. Каким быть учителю информатики?

Зарубежный опыт

Полат Е. Телекоммуникации в системе образования 110

Нам пишут

О программе курса 114
Снова бьем своих? 115
Что оценивать? 117
Открытое письмо директору завода «Квант» 118
Информатика в сельской школе 118
Это нужно учителю 120

Информация

Круглый стол: Информатика-88/89 122
Республиканский семинар-практикум 124
Львовские заметки 125

Веселый урок

Из жизни RAL126 128

Обложка Э. Бажилина

В оформлении номера принимали участие Э. Бажилин, С. Кустарева, А. Лобашинский, А. Пономарев, С. Расторгуев.

Главный редактор
академик
В. А. МЕЛЬНИКОВ

Редакционная
коллегия
И. М. БОБКО
Б. М. ГЕРАСИМОВ
Г. В. ГОДЖЕЛЛО
А. В. ДЕНИСЕНКО
А. П. ЕРШОВ
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

Редактор отдела Ф. Даниловский
Редактор отдела К. Шеховцев
Научный редактор Т. Драгныш
Заведующая редакцией Н. Игнатова
Художественный редактор Л. Розанова
Корректор М. Суворова

Издательство «Педагогика» Академии педагогических наук СССР и Государственного комитета СССР по делам издательства, полиграфии и книжной торговли

Почтовый адрес: 107005, Москва, Лефортовский пер., 8.
Телефон редакции: 249-97-77

Сдано в набор 22.07.88. Подписано в печать 30.08.88. А 05756.
Формат 70×100/16. Печать офсетная. Усл. печ. л. 10,40. Уч.-изд. л. 12,83. Усл. кр.-отт. 42,88. Тираж 95 340 экз. Заказ 1891. Цена 60 коп.

Ордена Трудового Красного Знамени Чеховский полиграфический комбинат ВО «Союзполиграфпром» Государственного комитета СССР по делам издательства, полиграфии и книжной торговли. 142300, г. Чехов Московской области.

© Издательство «Педагогика», «Информатика и образование», 1988

В. БЕЛОШАПКА

Мир как информационная структура

Стремительное распространение вычислительной техники и связанные с этим качественные преобразования в технологии, управлении, научных исследованиях, образовании и в быту, происходящие в промышленно развитых странах и вовлекающие в свою орбиту все современное человечество, имеют такой масштаб, что говорят о новой промышленной революции. В калейдоскопе вещей, событий и терминов, который предлагается нам, так много любопытных и пугающих новинок: гибкие автоматизированные производства, системы автоматизированного проектирования, персональные компьютеры, банки данных и знаний, компьютеризация образования, автоматизированные системы управления, робототехника, встроенные микропроцессоры, автоматизированные рабочие места, глобальные компьютерные информационные системы, новая информационная технология...

Откуда все это? Почему это происходит сейчас и не происходило раньше? Ассоциации, которые рождает в нас термин «революция», позволяют выделить этот комплекс событий, сконцентрировать на нем внимание, но мало способствуют нашему стремлению узнать смысл и причины происходящего, его исторические корни. А такое понимание, позволяющее увидеть события с общих, инвариантных позиций, нам нужно. Причем в особом положении на-

ходится учитель информатики, который отвечает за формирование представлений о происходящем у сегодняшних школьников.

Позвольте, глубокоуважаемый читатель, поделиться результатами размышлений об этом предмете, родившихся из опыта преподавания информатики, как основного курса, так и факультатива.

Начнем с примеров простых и практических. Пусть нам нужно вырыть яму, ну, скажем, чтобы посадить дерево или под фундамент дома. У этой простой задачи есть три аспекта. Первый: нужно изъять какое-то количество земли из одного места и переложить в другое. Эту землю можно охарактеризовать, например, объемом (в кубических метрах). Второй: эта работа потребует от нас затрат энергии, которую можно измерить (например, в джоулях). Третий аспект — так сказать, организационный. Ведь яму нужно вырыть в нужном месте, правильной формы и глубины, иначе придется переделывать.

Конечно, здесь размышлять долго не нужно и основные трудности связаны с собственно рытьем. Но рассмотрим теперь иной пример: завод по производству автомобилей. Здесь опять налицо все три рассмотренных выше аспекта. Действительно, для производства автомобилей нужно иметь определенные вещи, очень много вещей: оборудование, здания, сырье, детали и пр. Для того

3

чтобы привести эту гигантскую совокупность вещей в движение, необходима соответствующая энергия, причем в разных видах: электричество, бензин, газ и т. д. Но чтобы это движение было целенаправленным, а не хаотичным, чтобы действительно в итоге получались автомобили, необходимо продумать организационную сторону дела. Нужно, чтобы нужные вещи попадали в нужное место в нужное время, чтобы с ними производились нужные операции и многое другое. А для этого необходимо, чтобы там, где выполняются те или иные операции (перевозка, обработка, учет и пр.), в нужный момент оказались сведения, или, выражаясь формальнее, информация о том, что нужно сделать. Конечно, вещественный и энергетический компоненты такого производства — это серьезные и важные вопросы, но в последнее время вырелась точка зрения, что именно третий — организационный, структурный, информационный аспект — представляет в таких сложных объектах наибольший интерес, именно здесь следует ожидать наибольших трудностей.

Так что же? Значит, все в мире состоит из трех частей? Нет, но на каждый объект можно посмотреть с этих существенно разных и фундаментальных точек зрения.

Итак, выстраивается триада: вещество — энергия — информация. Еще раз подчеркнем, что это не три части, на которые мы поделили мир, а три точки зрения на него. Но здесь уместно задать вопрос: откуда взялась информационная точка зрения — ее недавно открыли или она появилась в связи с изобретением компьютера или чего-то еще?

Дело не в изобретении компьютера, хотя в данном круге событий оно играет важную роль. Информационную точку зрения вообще вряд ли можно признать принципиально новой. Видимо, никто никогда не возражал против того, что вещи (объекты) следует рассматривать как некие целостные совокупности (или, как мы сейчас говорим, системы), состоящие из более мелких частей (элементов); что элементы находятся в определенных взаимных зависимостях

(отношениях) и обладают теми или иными качествами (свойствами, характеристиками); что согласованное взаимодействие частей обусловлено обменом сигналами (информацией), в результате чего вещь остается равной самой себе (сохраняет свою структуру) и т. д.

Такие взгляды не были чужды ни новому времени, ни средним векам, ни античности. Но если все эти особенности системно-информационной точки зрения мы можем увидеть и в прошлом, то лишь в зачаточной или незрелой форме. Речь, если угодно, о познании мира человеком, о степени зрелости этого познания. И этим трем точкам зрения на мир соответствуют три фазы истории человечества. Эти фазы разделяют три промышленные революции, поэтому сначала проследим технологическую линию нашего сюжета.

Первая промышленная революция связана с использованием орудий, приспособлений, инструментов для обработки тех или иных материалов. Это начало активного освоения вещественного компонента мира. Вторая произошла в новое время, когда в промышленности стали использоваться первые двигатели. Это освоение энергетического компонента. Третья же происходит на наших глазах и связана с широким использованием вычислительной техники. Здесь речь идет об освоении информационного компонента нашего мира. Но, как хорошо известно, технологический или инженерный аспект всегда тесно связан с научным. Более того, уместна постановка вопроса о научной базе той или иной промышленной революции.

Обратимся поэтому к научной стороне дела. Начнем с математики (читатель, незнакомый с предметом, может пропустить эти абзацы). За вычетом протоматематической дисциплины — арифметики, к которой генетически восходят все собственно математические дисциплины, и метаматематической дисциплины — математической логики, которая, имея своим предметом саму математику, может быть названа математической лишь с оговорками, остаются геометрия, анализ и алгебра. Они знаменуют три этапа становления всей математики. В самом деле, античной мате-

матике геометрический взгляд на вещи был присущ настолько, что даже на такие, по нашим понятиям, негеометрические вопросы, как решение уравнений, греки смотрели геометрически. Греческая математика вполне реализовала все достоинства такой точки зрения и осталась связанной всеми ее ограничениями.

Для математики нового времени характерен расцвет математического анализа. Числовые функции в центре внимания, исчисление бесконечно малых дает мощный метод решения математических и прикладных (механика, физика) задач. Под знаком анализа оказывается вся математика. Геометрия становится аналитической и дифференциальной.

Но вот, возникнув в начале XIX в. из работ Галуа и Абеля, новое, современное направление в алгебре достигает в XX в. грандиозного масштаба, алгебраизируя всю математику. Достаточно отметить клейновскую программу построения геометрии [1] и роль теории непрерывных групп в современном математическом анализе. Аналогия, заключающаяся в сопоставлении рядов веществ — энергия — информация геометрия — анализ — алгебра вполне прозрачна: геометрия — это фигуры в пространстве, т. е. вещи и их положение; анализ — это время, это процесс, т. е. энергетика; наконец, современная алгебра — это структуры, отношения и операции, т. е. системно-информационный компонент.

Такой же ряд можно выстроить для каждой зрелой науки, например для физики, но ее сопряженность с математикой хорошо известна, поэтому заметим лишь, что аппарат современной физики — это аппарат теории групп как непрерывных, так и дискретных.

А как просматривается эта параллель в общенаучном контексте? Разным историческим периодам соответствуют различные представления о задаче и цели научного исследования, каждая фаза истории науки кристаллизовала, как сейчас говорят, свою парадигму научного исследования, получившую свое отражение в языке науки. И если древняя наука понимала свою задачу как

классификационно-описательную (всякой вещи надо было дать имя, установить ее место и свойства), то в новое время, когда математики занимались функциями и зависимостями и происходила вторая промышленная революция, в языке науки центральные позиции заняли такие термины, как развитие, процесс, эволюция. Всякая вещь рассматривалась как процесс становления самой себя, т. е. как совокупность своих функций. В начале же XX в. произошла кризисная перестройка языка науки [2]. Центральные термины новой системно-информационного языка — система, модель, язык, информация, структура, операция, алгоритм, инвариант и т. п.

Таким образом, эта тройственная стадийность истории имеет глубинный 5 характер, либо затрагивает основные виды человеческой деятельности.

Говорят, что в капле воды можно узанать океан и малое подобно великому. В качестве последнего примера рассмотрим логику частного научного исследования. Причем для простоты рассмотрим «чистый» случай: в поле зрения исследователя попадает совершенно незнакомый объект. Чтобы разговор был определеннее, давайте посмотрим в хороший микроскоп на обыкновенную живую клетку. С чего мы начнем? С внимательного рассматривания этой таинственной игры неведомых сил — ведь всякая реальность бесконечно глубока. Вот наш аналитический взгляд начинает различать части, т. е. те детали пейзажа, которые обладают известной устойчивостью. Мы учимся узнавать их, даем им названия — ядро, митохондрия, вакуоль, лизосома, описываем их свойства. При этом совершенствуется техника наблюдений и анализа (микроскопы, подготовка препаратов). Привлекая к использованию новые методы, исследование переходит на новый структурный уровень — биохимический, описываются классы важнейших для клетки соединений: жиры, углеводы, белки, ДНК, РНК. Анализируется биохимический состав открытых ранее оргanelл.

Постепенно картина проясняется: элементы выделены, описаны их свойства. Перед нами, так сказать, карта местно-

сти. Мы совершили пространственный, описательный анализ объекта. Однако дальнейшие исследования покажут, что объект так или иначе живет своей жизнью и полученная нами статическая картинка носит условный и приближительный характер. Свойства элементов меняются, среди совокупности этих изменений имеются связи, зависимости — это предмет временного, динамического анализа. Начнем описание внутриклеточных процессов: фаз деления, биохимических циклов и пр.

Но вот, преуспев в анализе динамики, мы устанавливаем некоторые связанные с объектом величины и характеристики, которые сохраняются во всех этих процессах и которые сами определяют всю эту динамику. Мы описываем структуры, ищем инварианты и открываем

законы. Перед нами проступает структура объекта. Это устройство ДНК, механизм синтеза белков, триплетный код и т. д. Мы как бы возвращаемся к статической картине, но это не та хромотопно-зависимая статика. Это нормативная природа объекта — закон, который им управляет. Мы знаем теперь о клетке неизмеримо больше. Так что же, мы знаем о ней все? Или хотя бы мы знаем о ней самое главное, существенное? Нет, мы не можем так сказать. Это неизвестно. Мы очень мало знаем об управлении активацией генов, о клеточных механизмах дифференцировки тканей. И клетка представляется нам сейчас, пожалуй, более таинственной и загадочной, чем раньше.

Теперь наши рассуждения можно подытожить в следующей таблице.

вещество	энергия	информация
геометрия	анализ	алгебра
выделение и описание элементов	исследование динамики	поиск инвариантов и законов
описательно-классификационный язык	процессуально-эволюционный язык	системно-информационный язык
инструмент	механизм	компьютер

Конечно, приведенная таблица не является ни в каком смысле полной. Просто мы попытались на нескольких основных примерах бегло очертить новую точку зрения, показать ее универсальность и поставить в смысловой и исторический ряды.

Может быть, теперь уместно, в соответствии с названием статьи, набросать величественную картину системно-информационного аспекта мироздания. Наверно, для этого необходимо представить космос, землю, живое, социум, человека как совокупность иерархически соподчиненных систем, пронизанных тонкой сетью вещественных, энергетических и информационных взаимозависимостей, отметив при этом, что все подсистемы этой грандиозной системы обладают структурно обусловленной стабильностью, которая реализуется как внутрисистемная детерминанта поддержания гомеостаза, причем необходимо вычленив все три упомянутых выше аспекта этой адаптивности: вещественную, энергетическую и информационную, отметив при этом имеющуюся

иерархичность, и особо остановиться на информационном аспекте, подчеркнув его доминирующий, главенствующий характер. Развивая эту тему, следовало бы воссоздать грандиозную систему информационных коммуникаций в природе и обществе: каналов, накопителей, фильтров и пр. Но все это уже описано — и у Н. Винера [3], и у С. Лема [4], и в других книгах. Поэтому, напомнив контуры этой панорамы, сделаем несколько замечаний.

1. Все богатство красок и выразительных возможностей, которые щедро демонстрируют нам эти авторы, — это возможности нового системно-информационного языка — языка описания формализованных моделей, языка, запечатлевшего в себе все основные концепции вызревшей в современном естествознании системно-информационной точки зрения на мир [2].

2. Это концептуальное богатство добыто совместными усилиями многих научных дисциплин. Но новая стадия зрелости освоения мира человеком, новая стадия осознанности этого системно-информационного взгляда на мир

и новая степень зрелости этих языковых средств породила специальную научную дисциплину — информатику, предметом которой и является проработка этих новых модельных представлений о мире, а также исследование и совершенствование языковых средств моделирования.

3. Развитие событий, как мы убедились, шло в направлении, заданном рядом: вещество — энергия — информация, и поэтому имеет смысл еще раз подчеркнуть семантическую рядоположенность этих трех понятий. Для этого рассмотрим некоторую динамическую систему, фазовое пространство которой расслаивается на семейство эквипотенциальных поверхностей. Такую систему можно представлять, как многоэтажный дом, где каждый этаж — горизонтальная плоскость, на которой расставлены стулья, шкафы и другая мебель, причем трение отсутствует. В этой ситуации можно данную конфигурацию (расстановку предметов) поменять на любую другую внутри того же этажа, затратив сколь угодно мало энергии, т. е. такой процесс будет обратим в классе чисто вещественных процессов (процессов без совершения работы).

Если же мы сместимся на другую поверхность (шкаф упал на другой этаж: предполагается, что имеются отверстия), то процесс сразу станет необратимым в этом классе. Энергия, выделенная при этом, и есть мера этой необратимости. Если вернуть эту энергию, то обратимость будет восстановлена. Итак, энергия — это мера необратимости для вещественных преобразований.

Теперь рассмотрим второе звено: энергия — информация. Напомним: возможность совершать работу за счет среды связана с ее неоднородностью (низкой энтропийностью). При этом обратимые процессы — те, которые эту неоднородность не портят. Если же неоднородность уменьшается, то процесс становится необратимым. Прирост энтропии и есть мера этой необратимости. Однако дело здесь не в геометрической однородности (горячие, «белые» частицы газа смешались с холодными, «черными») и вообще не в положении частиц. Покуда есть возможность следить

за трансформацией, необратимости нет. Она возникает в тот момент, когда информация потерялась, наше «зрение» ослабло, картина стала серой и мы не знаем, где что находится. Если восстановить эту утраченную информацию, то процесс опять становится обратимым (в частности, можно собрать «белые» частицы в одной половине сосуда, а «черные» — в другой). Таким образом, информация — это мера необратимости для энергетических преобразований. Аналогичность обоих звеньев стала вполне очевидной, и можно еще раз сформулировать наш тезис: ряд вещество — энергия — информация характеризует три фазы зрелости модельных представлений.

4. Мы далеки от того, чтобы утверждать, что последним, третьим членом ряда заканчивается процесс познания. Более того, мы не решаемся утверждать, что уже открыто самое главное и существенное, хотя порой масштабность происходящего располагает к такому оптимизму. В такие минуты полезно вспоминать, что когда в новое время в связи с промышленной революцией и успехами математики и механики энергетизм в форме механицизма получил широкое распространение, ему придавался именно всеобщий и окончательный характер.

Отметим, что в использовании тех широких и далеко идущих аналогий, к которым так располагает новый системно-информационный язык, нужна известная трезвость. Конечно, вся эта метафорика уместна, плодотворна, незаменима для построения моделей и гипотез. Однако, совершая насильственное отождествление объекта с его моделью и переводя наши построения в статус фундаментальных научных положений и, далее, делая их в таком их статусе достоянием культуры, мы совершаем рискованный шаг. Впрочем, в конце концов мы, взрослые люди, разберемся, что к чему. Но на страницах школьного учебника информационный фетишизм недопустим.

5. Течение времени, которое мы рассматривали сквозь призму ряда вещество — энергия — информация, сопровождалось также повышением степени

формализованности модельных представлений, повышением формализованности языков этих построений. Если сравнить степень формализованности построений античных авторов и моделей, построенных новыми средствами, этого нельзя не заметить. Однако следует подчеркнуть, что этот параметр находится, по-видимому, в прямой зависимости не только от эпохи, но и от степени зрелости процесса. Вспомним хотя бы представления, развитые древнекитайской медициной. Не останавливаясь на анализе их содержания, отметим, что эти модели (инь-янь, пять элементов и пр.) по степени формализованности и по средствам записи (ориентированные графы) находятся вполне на современном уровне [5].

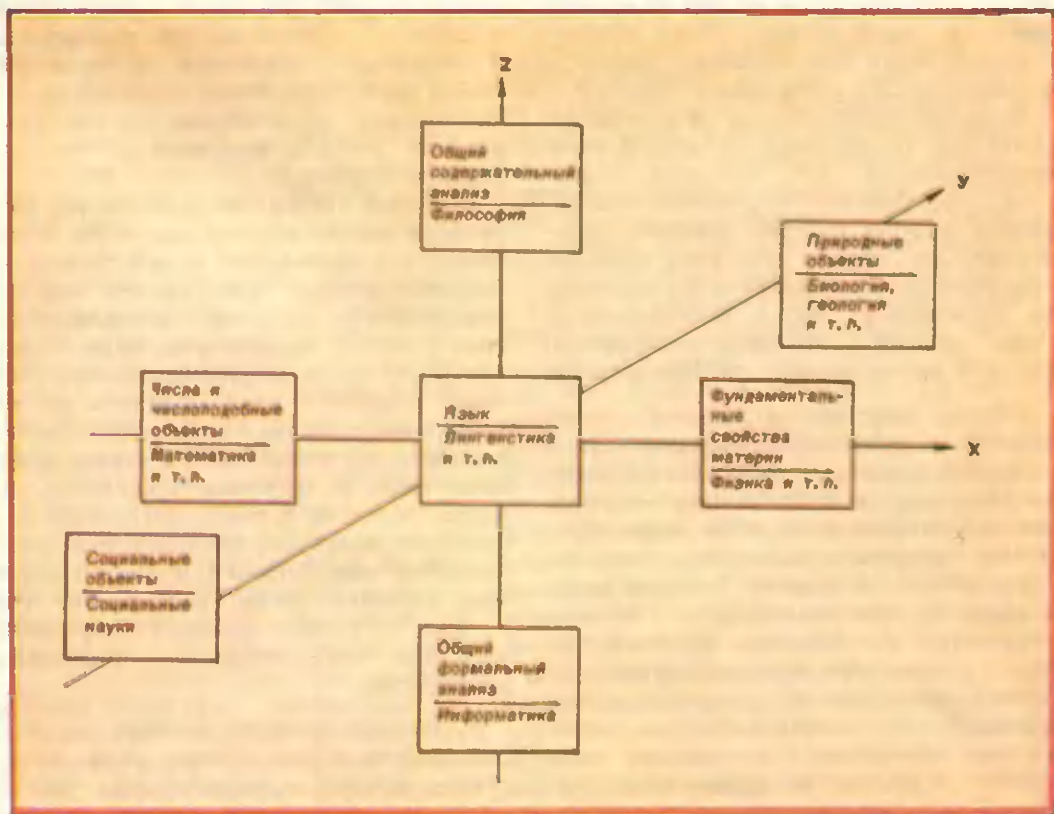
8

Здесь следует добавить, что степень формализованности значений находится в прямой зависимости с легкостью их отчуждения и обобщения.

6. Как было отмечено выше, основные концепции системно-информацион-

ного языка вызревали в конкретных исследованиях, однако достоянием информатики они стали после обобщения и формализации. Информатика — наука неконкретная, обобщенная. Это ставит ее на одну линию с философией. Именно из-за того, что предметом информатики являются общие закономерности, представители конкретных научных дисциплин так часто отказываются признать за ней статус науки. Впрочем, они отказывают в нем и философии. Это у информатики и философии общее. Однако важно понимать и видеть ту грань, которая разделяет эти дисциплины. Подобно тому как слово имеет два аспекта (слово как смысл и как знак), так и философия и информатика — это два аспекта общего научного взгляда на мир: философия своим предметом имеет общий содержательный анализ, тогда как предметом информатики является общий формальный анализ.

Поучительно найти точку в истории



науки, в которой произошло это расщепление. Мы берем на себя смелость указать ее: это то время, когда западная средневековая философия породила такое направление, как схоластика с ее подчеркнутым интересом к формальному анализу явлений. Конечно, это не само дифференцирование, а его начало, и современных формального аппарата и технической базы у схоластики не было. Но генетически и структурно аналогия правильная. Схоластика переживала большие трудности в связи с замкнуто-формальным ее характером, — провозглашалось ни много ни мало то, что человеческий разум из ряда положений с помощью формального исчисления способен получить полное знание, в частности все законы природы. Так же и современная схоластика — информатика несет в себе, пока не вполне осознанные, но те же интенции: например, раздаются заявления о скором упразднении конкретной экспериментальной науки средствами вычислительного эксперимента [6].

7. В заключение предложим некоторую классификацию научных дисциплин, которая отражает вышеизложенные соображения о месте информатики. Причем подчеркнем, что имеются в виду именно теоретические дисциплины, тогда как инженерные дисциплины здесь не рассматриваются. На рисунке каждая ось имеет содержательную интерпретацию: OX — это абстрактное — конкретное, OY — это социальное — природное, OZ — это формальное — содержательное. Точка O , в которой встречаются все оси, — это язык, в котором запечатлены основные концепции всех дисциплин.

Литература

1. Клейн Ф. Элементарная математика с точки зрения высшей. М., 1987.
2. Белошанка В. О языках, моделях и информатике // Информатика и образование. 1987. № 6.
3. Винер Н. Кибернетика. М., 1983.
4. Лем С. Сумма технологий. М., 1969.
5. Табеева Д. М. Руководство по иглорефлексотерапии. М., 1982.
6. Гелентер Д. Современное программирование // В мире науки. 1987. № 12.

9

Нарисованная музыка

Все знают, что музыку можно записать с помощью нот. Но как записать трель соловья или мерный стук сердца?

Сделать это совсем несложно, если вспомнить, что любой звук — это упругая волна более или менее сложной формы, которую с помощью микрофона можно легко преобразовать в электрический сигнал. Сигнал этот можно превратить затем в движение пера самописца, записать на магнитную ленту, увидеть на экране осциллографа.

Одна из американских фирм, разрабатывающих музыкальную аппаратуру, изготовила подключаемый к синтезатору компьютер, на дисплее которого исполняемая мелодия обретает графическую форму. Удары человеческого сердца, например, превратятся на экране такого компьютера в

ЧТО?
МОЖЕТ
ЭВМ

привычную кардиограмму, а чистый тон звукового генератора — в аккуратную синусоиду.

Компьютер снабжен программой-редактором, которая превращает его из обычного осциллографа в инструмент композитора. Музыкальный редактор с помощью простых средств (выбор строчки предлагаемого меню или имеющегося образца музыкального тона) позволяет изменять форму амплитудно-временного рисунка музыкального произведения. С помощью редактора

можно увеличить или уменьшить амплитуду какого-нибудь тона или гармоники, изменить их частоту и форму, вырезать или вставить фрагмент музыкального произведения, ввести задержку, инвертировать выбранный участок, добавить предлагаемый компьютером сигнал и даже нарисовать мышью новую мелодию. Причем последнее качество музыкального редактора позволяет сочинять музыку людям, не умеющим играть на музыкальных инструментах и не владеющим музыкальной грамотой, но знающим законы акустики.

Настоящие же музыканты найдут в музыкальном редакторе исполнительного помощника с надежной и вместительной памятью, в которую постепенно, звук за звуком, можно вносить очень тонкие и изящные «музыкальные» штрихи. Как с помощью клавиш синтезатора и тонкого слуха, так и с помощью клавиш компьютера и твердой руки.

А. ДУВАНОВ, М. ГОЛЬЦМАН

«Черные ящики»

В статье говорится об использовании известного способа исследования — метода «черного ящика» для формирования на уроках информатики навыков работы с алгоритмом как с правилом, по которому действует исполнитель. Метод хорош тем, что в качестве побочного эффекта формирует у учащихся исследовательские навыки и развивает творческую активность. Поэтому он, по мнению авторов, может с успехом использоваться при изучении других предметов.

«Черный ящик»

«Черным ящиком» называется метод исследования, при котором изучаемый объект рассматривается как неделимое целое, не имеющее структуры. Считается, что в оболочке объекта царит «черный джин», преобразующий входное воздействие x по некоторому правилу f в выходную реакцию y (рис. 1).

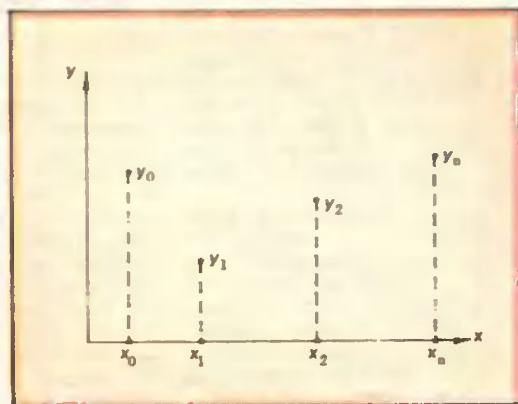
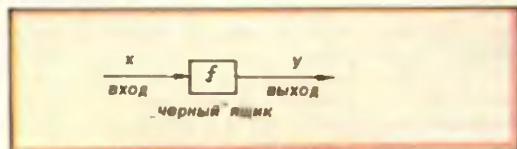
Вооружившись этим методом, необходимо сначала построить «черный ящик», т. е. выделить исследуемый объект (или его часть), определить для него входы и выходы, выяснить тип воздей-

ствий, на которые «черный ящик» реагирует, построить закон поведения f , проверить его на практике и затем, в случае удачи, рекомендовать свое открытие к практическому использованию.

Обычно работа сводится к организации серии опытов с объектом, в ходе которых на вход «черного ящика» подаются последовательно воздействия x_0, x_1, \dots, x_n и регистрируются реакции на выходе y_0, y_1, \dots, y_n соответственно. Результаты сводятся в таблицу или отмечаются точками на плоскости переменных x, y (рис. 2). Теперь необходимо найти такую функцию F , которая бы принимала в точках таблицы те же значения, что и функция джина f :

$$F(x_i) = f(x_i) = y_i \text{ для всех } i \text{ из } [0, n].$$

Геометрически это означает, что нужно подобрать такую функцию F , график



которой проходил бы через точки (x_i, y_i) для всех $i \in [0, n]$.

В математике задача построения функции F по таблице называется задачей интерполяции, которая, строго говоря, имеет не единственное решение. В самом деле, через точки (x_i, y_i) можно провести бесконечное множество кривых, каждая из которых может являться решением поставленной интерполяционной задачи.

Вводя дополнительные ограничения, можно добиться единственности интерполирующей функции F . Например, существует единственный полином n -й степени, решающий поставленную задачу.

Когда функция F построена, встает вопрос о точности приближения. Ясно, что если джин не работает случайным образом, то для входных воздействий функция F совпадает с f в узловых точках. А для произвольных входных воздействий? Этот вопрос может решить только практика. Либо найденная зависимость сравнительно точно предсказывает поведение «черного ящика», либо предсказывает его удовлетворительно, либо совершенно не годится для практического использования. В последнем случае задача интерполяции решается заново.

Классическим экспериментом по методу «черного ящика» являются опыты выдающегося физиолога И. П. Павлова на собаках. Собака объявлялась «черным ящиком», т. е. не учитывалась ее анатомия. Входными воздействиями являлись: освещенность, температура, влажность, присутствие еды и т. п., а выходной реакцией — наличие слюноотделения или выделение желудочного сока. Проведя свои опыты, И. П. Павлов установил закон возникновения условных рефлексов.

«Черный ящик» начинает светлеть

Метод «черного ящика» может быть использован в несколько иной интерпретации. В этом случае заранее декларируется функциональная характеристика объекта. Первоначально она заменяет объект в некоторой общей модели. За-

тем начинается изготовление «внутренностей» объекта для реализации декларированной функции.

Такой подход продуктивен при конструировании сложных систем. Сначала обозначается скелетный план, состоящий из взаимосвязей функционально-определенных «черных ящиков». На нем хорошо просматривается общая структура конструкции, взаимодействие всех ее частей. Затем проектируются внутренние устройства сначала наиболее важных, а потом и остальных «черных ящиков». При этом сохраняется неизменной каждая проверенная на скелетном плане внешняя декларированная функция.

Светлеющий «черный ящик» может оказаться слишком сложным для проектирования всех его частей: некоторые из них снова объявляются «черными ящиками» и так далее.

Такой метод, позволяющий прежде проектировать структуру, а потом детали, называется структурным подходом или нисходящей технологией. Он применяется, например, при проектировании самолета. Сначала определяются общие очертания, затем двигатель, устройство внутренних салонов, наконец обшивка кресел и крепежные детали. (На практике стараются проектировать салон так, чтобы можно было использовать стандартные узлы.)

Метод «Черного ящика» на уроках информатики

Метод «черного ящика» оказался удобным инструментом, способным помочь школьникам усвоить тему «Алгоритмы». Авторами проделаны соответствующие эксперименты, давшие положительные результаты.

Сразу оговоримся, что речь пойдет не об уроках информатики в IX и X классах, а об оригинальном эксперименте Роботландия, проводимом в г. Переславле-Залесском лабораторией школьной информатики ИПС АН СССР в III и V классах обычной средней школы.

Этот эксперимент по обучению информатике младших школьников базируется на следующих идеях.

1. В процессе изучения информатики

в общеобразовательной школе учащиеся должны получить знания о концептуальном программировании, т. е. у них необходимо воспитывать алгоритмический способ мышления и знания алгоритмической технологии решения задач.

2. Школьники должны изучать современные приложения информатики, т. е. использование ЭВМ для поиска, обработки, хранения информации, в частности применение ЭВМ в качестве средства коммуникаций.

3. Знания по информатике должны помогать учащимся при изучении других предметов, в том числе гуманитарных.

4. Информатика должна пробуждать исследовательские и конструкторские интересы детей, развивать фантазию и творческий потенциал.

12

Приведем фрагменты урока с комментариями.

Дети начинают четверть, знакомясь с понятием «исполнитель». Это некий обработчик информации, управляемый командами из фиксированного набора — системы команд исполнителя (СКИ). Различаются формальные и неформальные исполнители. Понятие алгоритма вводится как правило, по которому исполнитель обрабатывает информацию.

Ниже рассматривается один из апробированных фрагментов программы эксперимента. Этот материал использовался во второй четверти в III классе.

Учитель. Сегодня мы будем играть в новую игру «Отгадай меня». Для начала вспомним смысл слов «информация», «исполнитель», «алгоритм». Итак, что такое информация?

Г а л я. Информация — это когда кто-нибудь говорит.

У ч и т е л ь. Да, речь содержит информацию.

И р а. Информация хранится в компьютере.

О л я. Информация — это стихи, слова.

С а ш а. Информация — это то, что ты передаешь соседу.

У ч и т е л ь. Можно сказать, что информация всегда какое-нибудь сообщение, какие-либо сведения. А что можно делать с информацией?

В а д и м. Информацию можно хранить, передавать и обрабатывать.

И р а. Информация хранится в компьютере, в голове, на магнитной ленте.

Д и м а. Информацию можно передавать по проводам, по почте, голосом.

П а в е л. Обрабатывают информацию люди, компьютеры, животные, предметы.

У ч и т е л ь. Зачем нужно обрабатывать информацию?

У л я. Для того чтобы получить ответ, которого не знал.

У ч и т е л ь. Поясни примером.

У л я. Учитель задает пример 2×8 . Я обрабатываю и получаю ответ 16.

У ч и т е л ь. Верно, информацию обрабатывают для того, чтобы получить новую информацию, которая раньше была неизвестна.

Комментарий

Начальная небольшая часть урока опущена. Представленный фрагмент урока начинается с повторения ранее введенных понятий. Информации нельзя в силу ее первичности дать строгое определение, т. е. свести это понятие к какому-либо другим. Дети не смогли запомнить «определение», данное им на предыдущих уроках, гласящее, что информация — это сведения. Хотя слово «сведения» для детей более простое, они все же чувствуют тавтологию. Их определения сводятся к конкретным примерам информационных явлений. Учитель не настаивает на зазубривании данного им определения. На вопрос о том, что можно делать с информацией, дети отвечают очень четко и приводят много правильных примеров.

Конец комментария

У ч и т е л ь. Кто помнит определение исполнителя?

А н д р е й. Исполнитель — это коллектив, птицы, люди.

Н а т а ш а. Исполнитель — это человек, бригада, устройство.

У ч и т е л ь. Вы забываете, что исполнитель тот, кто исполняет. Люди, птицы, устройства не всегда выступают в роли исполнителей. Ваши определения неточны.

С а ш а. Исполнитель тот, кому дают команду.

Учитель. А если тот, кому дают команду, не выполняет ее?

Саша. Тогда он не исполнитель.

Учитель. Верно! А кому можно давать команду?

Саша. Собаке в цирке.

Учитель. Только собаке? Кто все же даст полный ответ?

Жанна. Исполнителем называется человек или коллектив, животные или растения, различные устройства, которые умеют различать и выполнять команды.

Учитель. Очень хороший ответ. В нем сказано, кто может быть исполнителем и при каких условиях. Приведите примеры исполнителей. Исполнитель-человек?

Павел. Когда я по просьбе мамы мою посуду.

Учитель. Исполнитель-коллектив?

Наташа. Бригада маляров красит дом.

Учитель. Исполнитель-животное?

Саша. Собака выполняет команды дирижера в цирке.

Учитель. Не дирижера, а дрессировщика. Исполнитель-растение?

Ира. Подсолнух.

Учитель. Объясни, когда подсолнух является исполнителем?

Ира. Когда поворачивается к солнцу.

Учитель. А кто отдает команду?

Ира. Солнце. Оно говорит: «Я здесь». Это команда подсолнуху повернуться к солнцу.

Учитель. Исполнитель-устройство?

Сережа. ЭВМ.

Учитель. Еще?

Сережа. Магнитофон, телевизор, пылесос, стиральная машина.

Уля. Автомобиль, поезд, самолет.

Комментарий

Сначала опрашиваются отстающие дети. Учитель отмечает ошибки в их ответах. Затем правильное определение дает Жанна. Понятие исполнителя достаточно хорошо усвоено детьми. Они знают, что исполнитель обрабатывает информацию, выполняя команды из СКИ; знают смысл, выдаваемой исполнителем диагностики: «не понял» на синтаксическую ошибку, «не могу» на ошибку в семантике (исполнителя

при помощи правильной команды заставляют выполнять запрещенные действия).

Конец комментария

Учитель. Что такое алгоритм?

Дима. Алгоритм — правило, по которому исполнитель выполняет работу.

Учитель. Хорошо. А теперь я расскажу правила новой игры. Я запишу на доске диалог между исполнителем и командиром, а вы попытайтесь разгадать алгоритм, по которому исполнитель обрабатывает информацию.

Командир	Исполнитель
кот	не понял
1	2
2	3
3	4
4	5

Оля. Исполнитель прибавляет один.

Учитель. Верно! Это объясняет обработку всех чисел в приведенном диалоге. Значит, можно считать, что алгоритм исполнителя состоит в том, что он добавляет к заданному командиром числу единицу. А почему на слово «кот» исполнитель выдал сообщение «не понял»?

Гая. Исполнитель не работает с буквами. Он обрабатывает только числа.

Учитель. Хорошо. А теперь следующий пример.

Командир	Исполнитель
1	не понял
кот	3
мороз	5
компьютер	9

Лена. Исполнитель работает со словами, он считает в них буквы.

Учитель. Продолжим игру немного по-другому. Я придумал алгоритм и буду работать по этому алгоритму как исполнитель, а вы отгадайте меня! Я рисую на доске столбик с надписью «Командир» — в него я буду писать ваши команды. Рядом рисую столбик с надписью «Исполнитель» — в него я буду писать свои ответы. Начинаем!

Павел. 10.

Учитель. Не понял. Почему я ответил так?

Дима. Потому что исполнитель не работает с числами.

Вадим. Корова.

Учитель. 2.

Ира. Енот.

Учитель. 1.

Оля. Ель.

Учитель. 0. Если кто-нибудь уже догадался, какой у меня алгоритм, может рассказать его. Ну если никто не догадался, продолжим испытания.

Лена. Сорока.

Учитель. 2.

Уля. Я знаю. Исполнитель считает слоги и отнимает 1.

Учитель. Давай проверим твое предположение. Задай мне информацию.

Уля. Самолет.

Учитель. Что я должен ответить?

Уля. 3.

Учитель. Мой ответ 1.

Дима. Исполнитель считает буквы «о»!

Учитель. Проверь.

Дима. Ого. Исполнитель должен ответить 2.

Учитель. Мой ответ 2. Еще проверка.

Дима. Ага. Исполнитель должен ответить 0.

Учитель. Мой ответ 0. Вынужден сознаться, что я задумал именно этот алгоритм. А теперь на роль исполнителя мы выберем ... Игоря.

Комментарий

В приведенном фрагменте урока начинается активное усвоение детьми понятия «алгоритм» в практической работе. Исполнители рассматриваются как «черные ящики», и от ребенка требуется определить алгоритм их поведения.

На первом уровне сложности предлагается протокол испытания исполнителя, т. е. таблица соответствий выходных реакций от нескольких входных воздействий. Требуется провести анализ заданной таблицы и выдвинуть гипотезу алгоритма, которая объясняла бы все табличные соответствия. Как отмечалось выше, поставленная задача интерполяции имеет бесчисленное множество решений. На практике это означает, что дети могут предложить не один алгоритм, объясняющий поведение исполнителя, а несколько, и это не является ошибкой.

Следующий шаг — заставить детей самих инициировать диалог с «черным

ящиком», составить таблицу опроса, предложить интерполирующий алгоритм и наконец проверить его на практике путем сравнения предсказанной реакции на входное воздействие с реальным поведением. По сути, здесь детям приходится строить алгоритмическую модель исполнителя, затем проверять на практике, отвергать и строить заново, если какой-то опыт не укладывается в ее рамки.

Преимущество такого способа усвоения материала, во-первых, в активных действиях детей, без которых знание остается формальным, во-вторых, во множестве полезных «побочных» эффектов.

Дети начинают игру с «черными ящиками» без машины. Ящики «светлеют» при активной работе коллективной мысли. Наконец наступает момент, когда все поняли, чего от них ждет учитель, увлеклись игрой, и каждый хочет отгадать алгоритм совершенно самостоятельно. Когда такой момент, по мнению учителя, наступает, включается компьютер.

Конец комментария

Учитель. Сейчас вы будете работать с исполнителями. У вас на экранах светится меню из 7 блюд. На первое — исполнитель номер 1, на второе — исполнитель номер 2, и на третье, и на седьмое тоже исполнители. Сначала вы выбираете исполнителя номер 1 и отгадываете его алгоритм. Потом принимаетесь за исполнителя 2 и так далее, сколько успеете. Выбор, как всегда, осуществляется перемещением курсора к нужному номеру и нажатием клавиши выполнения. Итак, отгадаем исполнителя номер 1.

Комментарий

На экранах детей представлено изображение (рис. 3).

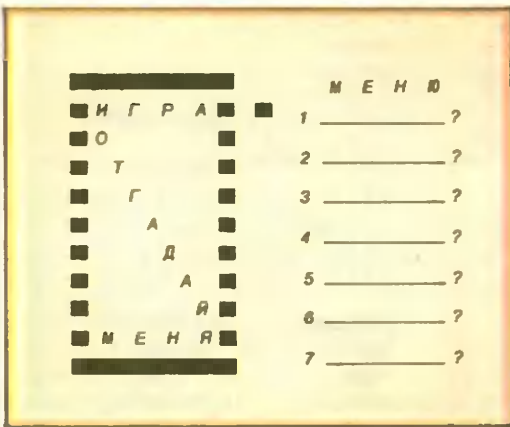
Нажимая на клавишу выполнения, они входят в среду исполнителя, которая на экране изображается следующим образом:

Отгадай алгоритм 1

Командир Исполнитель

Приглашение «>» и курсор предлагают начать диалог с исполнителем.

Конец комментария



Учитель. Сейчас вы должны начать свой диалог с исполнителем. Вы вводите в него информацию, набирая ее на клавиатуре, и смотрите, как он ее обрабатывает. Как только вы догадаетесь, какой алгоритм у исполнителя, поднимите вверх палец.

Комментарий

Протокол работы Лены с исполнителем:

Командир	Исполнитель
пн	не понял
1	2
10	20
5	10

Палец поднят.

Конец комментария

Учитель. Я вижу много поднятых пальцев. Но проверять ваши алгоритмы буду не я, а ЭВМ. Нажмите клавишу ESC — вход в режим проверки. Теперь роли меняются: вы работаете за исполнителя по его правилу. Машина предлагает информацию, а вы ее обрабатываете так, как это сделал бы исполнитель, и набираете на клавишах ваш ответ.

Комментарий

Ниже приводится протокол работы Лены с ЭВМ в режиме проверки.

Проверим алгоритм 1.

Командир	Исполнитель
7	14 +
11	23 —
9	18 +
13	26 +
17	34 +

Алгоритм угадан!

Конец комментария

Учитель. Откройте тетради и запишите решение первой задачи:

Алгоритм 1 — исполнитель умножает число на 2.

Теперь приступайте к работе со следующими исполнителями, выбирая их при помощи меню. Ответы, так же как на первую задачу, записывайте в тетрадь.

Комментарий

Далее дети работают самостоятельно. Учитель помогает им преодолевать технические трудности, возникающие при диалоге с новой для них программой. Работа заканчивается за 3 мин до конца урока. Дети выключают компьютеры, сдают тетради на проверку, получают домашнее задание, заранее подготовленное на листочках. Трое детей из класса успевают отгадать все 7 алгоритмов. На следующем уроке они получают дополнительное задание повышенной трудности, в то время как остальные дети продолжают работу с этой же программой.

Ниже приводятся ответы для всех 7 алгоритмов, протокол работы с исполнителем 7 и текст домашней работы.

Ответы к алгоритмам программы.

- 1) $2x$ 2) $2x+1$ 3) $x-1$ 4) $2x-1$ 5) $x/2$
6) $x/3$ 7) $x/2-1$

Протокол работы Иры с исполнителем номер 7.

Отгадай алгоритм 7.

Командир	Исполнитель
0	не могу
1	не могу
2	0
3	не могу
4	1
5	не могу
6	2

Нажата клавиша ESC.

Домашняя работа.

Угадайте алгоритмы исполнителей.

1.

Командир	Исполнитель
2	8
3	12
4	16

Ответ: $4x$.

2.

Командир	Исполнитель
1	3
2	7
3	11
4	15

Ответ: $4x-1$.

3.	Командир	Исполнитель
	вода	2
	липа	2
	ура	1

Ответ: число согласных в слове.

4.	Командир	Исполнитель
	а	0
	киса	3
	ель	2
	мороз	4

Ответ: $x+1$, где x — число согласных в слове.

Конец комментария

16

На уроках Роботландии использовалось много программно-реализованных «черных ящиков». Это ящики «букведы», обрабатывающие символьную информацию, графические ящики, музыкальные и координатные. При работе с координатным ящиком обнаружилось, что дети плохо его воспринимают из-за отсутствия навыков работы с координатной системой. Когда же выяснилось, что многие из детей не знают игры «Морской бой», такая специализированная игра-исполнитель, оптимально адаптирующаяся (по ответной стратегии) к уровню играющего, была срочно реализована на ЭВМ. На одном из уроков был проведен чемпионат класса по этой игре, после которого сложный координатный «черный ящик» был активно разгадан.

Приведем список наиболее интересных, на наш взгляд, задач, связанных с алгоритмической интерполяцией протоколов работы «черных ящиков». Эти задачи либо давались детям в качестве домашнего задания, либо решались коллективно в классе.

Отгадайте алгоритмы исполнителей.

1.	Командир	Исполнитель
	программа	1
	яма	0
	домашнее	1
	группа	1
	процессор	1
	ввод	1
	введенный	2

2.	Командир	Исполнитель
	август	1
	галоп	2
	караван	3
	магнитофон	4

	кукуруза	3
	лопух	1
	робот	1

3.	Командир	Исполнитель
	23	6
	2	не могу
	12	2
	10	0
	15	5
	42	8

4.	Командир	Исполнитель
	друг	есфд
	враг	гсбд
	мясо	не могу
	кот	лпу
	мышь	ныщэ

5.	Командир	Исполнитель
	Вася	не могу
	кот	1
	трубач	2
	шалаш	не могу
	программа	

6.	Командир	Исполнитель
	ворон	в
	леопард	а
	процессор	е
	жук	ж
	Кузя	з

7.	Командир	Исполнитель
	12	1
	26	4
	49	5
	55	0
	81	7
	19	8

«Черный ящик»: предложение и идеи

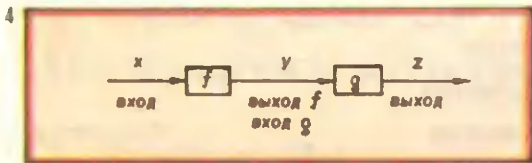
Как нам кажется, метод «черного ящика» может быть использован наряду с другими средствами для пропедевтики понятий управляющих структур.

Простое следование (композиция) «черных ящиков»

Очертим круг задач, которые можно ставить перед учащимися в данном случае (рис. 4).

1. Заданы алгоритмы f и g , требуется описать алгоритм композиции.

2. Задан алгоритм fg , реализующий композицию. Требуется подобрать со-



ставляющие алгоритмы f и g .

3. Задан протокол опроса системы ящиков $x \rightarrow z$. Требуется найти алгоритмы fg , f и g , которые соответствовали бы данному протоколу.

4. Задан один из алгоритмов f (или g) и протокол опроса системы $x \rightarrow z$. Требуется подобрать другой алгоритм g (или f), такой, чтобы данный протокол соответствовал композиции алгоритмов.

5. Заданы протоколы $x \rightarrow y$ и $y \rightarrow z$. Требуется найти алгоритмы f , g и алгоритм их композиции fg .

6. Задан один из протоколов $x \rightarrow y$ (или $y \rightarrow z$) и протокол системы $x \rightarrow z$. Требуется подобрать реализующие алгоритмы f и g .

Ниже приводятся примеры для каждой группы задач.

1. $f: 3x$.
 $g: 5y$.

Ответ: $fg: 15x$.

2. fg : считает число окошек на рисунке, изображающем домик, и отнимает от него 1.

Ответ: f : считает число окошек.
 $g: y - 1$.

3.

x	z
корова	4
пень	0
молоко	6
кол	2

Ответ: f : считает число букв «о» в слове x .
 $g: 2y$.
 fg : умножает на 2 число букв «о» в слове.

4.

x	z	
1	4	$f: x + 1$.
2	6	
3	8	
4	10	

Ответ: $g: 2y$.

5.

x	y	y	z
1	0	1	3
2	1	2	6
3	2	3	9
4	3	4	12

Ответ: $f: x - 1$.

$g: 3y$.

$fg: 3(x - 1)$.

6.

x	z	y	z
1	не могу	1	3
2	3	2	6
3	не могу	3	9
4	6	4	12

Ответ: $f: x/2$.

$g: 3y$.

$fg: x/2 \times 3$.

«Черные ящики»

с ветвящимися алгоритмами

Алгоритм ящика f ветвится по условию. Рассмотрим несколько примеров.

1.

x	y
1	2
2	3
3	6
4	5
5	10

Ответ: $f: \begin{cases} 2x, & \text{если } x \text{ нечетное число;} \\ x+1, & \text{если } x \text{ четное число.} \end{cases}$

2.

x	y
дерево	2
кот	1
молоко	0
ель	0
ельник	1
полос	2

Ответ: $f: \begin{cases} \text{считает буквы «е», если число} \\ \text{букв в слове четно;} \\ \text{считает буквы «о», если число} \\ \text{букв в слове нечетно.} \end{cases}$

«Черные ящики»

с циклическим алгоритмом

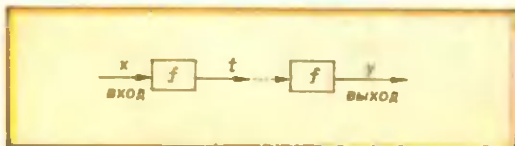
Алгоритм ящика f циклически применяется несколько раз так, будто выход замкнут на вход, т. е. циклический алгоритм эквивалентен простому следованию нескольких ящиков с одинаковым алгоритмом f (см. рис. 5).

Известно, что f применяется 2 раза. Определить f по следующему протоколу:

x	y
1	2
2	6
3	10
4	14
5	18

Ответ: $f: 2x - 1$.

Другой случай повторения получается



циклическим повторением алгоритма f , пока выполняется некоторое условие. Рассмотрим задачи двух типов.

1. Известно, что алгоритм f состоит в вычеркивании нулей в исходном числе и умножении его на 5, пока в числе есть нули. Как преобразует ящик числа 40, 15 и 201?

Ответ: 40 в 5, 15 в 15, 201 в 75.

2. Предложить циклический алгоритм f , объясняющий следующий протокол.

x	y
1	1
2	1
3	3
4	1
80	5
112	7

Ответ: f : деление на 2, пока делится.

18

Метод «черного ящика» в других предметах

Русский язык

Командир

Исполнитель

кот	кот		
мышь	кот	мышь	
солнце	кот	мышь	солнце
компьютер	кот	мышь	солнце
ЭВМ	компьютер		
	кот	мышь	солнце
парта	компьютер	ЭВМ	
	кот	мышь	солнце
	компьютер	ЭВМ	парта
стек	?		
липа	?		
молоко	?		

В данном примере исполнитель классифицирует слова (помещает их в нуж-

ную графу таблицы) по родовому признаку.

Биология

Командир

Исполнитель

собака	собака	
кошка	собака	кошка
тигр	собака	кошка
		тигр
волк	собака	кошка
	волк	тигр
шакал	?	
рысь	?	

Классификация по принадлежности к семействам кошачьих или собачьих.

Химия

Командир

Исполнитель

CH_4	CH_3OH
C_2H_6	C_2O_5OH
C_3H_8	C_3H_7OH
C_4H_{10}	?
C_5H_{12}	?

Исполнитель, получая формулу предельного углеводорода, преобразует ее в формулу соответствующего спирта.

Физика

Командир

Исполнитель

$a=1$	
$t=1$	0.5
$a=2$	
$t=1$	1
$a=2$	
$t=2$	4
$a=2$	
$t=3$?
$a=3$	
$t=4$?

Исполнитель вычисляет путь при равноускоренном движении по формуле $at^2/2$.

Авторы далеки от мысли считать метод «черного ящика» панацеей от всех бед педагогики. Однако они надеются, что эта работа даст в руки учителя инструментарий для успешного введения детей в мир информатики.

Из рукописи конкурсного учебника

2. Устройство ЭВМ

Тысячи лет человечество изготавливало и использовало различные виды исполнителей. Но вот появились электронно-вычислительные машины. Первоначально ЭВМ предназначались для решения вычислительных задач, возникающих в математике и физике. Об этом свидетельствует и само слово «вычислительная», т. е. ЭВМ была исполнителем, который может быстро и точно вычислять. Однако с развитием вычислительной техники область применения ЭВМ существенно расширилась: экономика, медицина, музыка, рисование, спорт, образование, космос... ЭВМ может быть и устройством управления другими исполнителями. Появился и будет крепнуть союз традиционной, привычной нам техники с ЭВМ. Не просто автомобиль, а автомобиль с управляющей им ЭВМ, позволяющей экономить бензин и облегчать работу водителя. Не просто станок, а станок с программным управлением, обрабатывающий деталь по заранее написанной программе; не электровоз, а электровоз с управляющей ЭВМ, позволяющей экономить энергию. Таким образом, ЭВМ является качественно новым исполнителем.

Как ЭВМ может решать столь разные задачи? Каждая ЭВМ может понимать и выполнять какое-то сравнительно небольшое число различных элементарных команд. Из этих команд составляются программы. Как из 33 букв русского языка можно написать и школьное сочинение, и роман «Война и мир», так и из этого небольшого числа команд можно составить небольшую учебную программу, а можно и сложную программу из миллиона команд.

2.1. Принципиальная схема ЭВМ

Говорить о том или ином сложном

приборе можно с разных точек зрения. Что такое телевизор, по-разному понимает первоклассник, десятиклассник, режиссер телепередачи, мастер из телеателье, инженер завода по изготовлению телевизоров, конструктор, изобретающий новую модель телевизора. Об автомобиле разные понятия у пассажира такси, шофера, механика, автогонщика. Дело не только в том, что один знает больше, а другой меньше, — разные люди смотрят на один и тот же предмет с разных точек зрения, им нужны различные знания.

Мы будем знакомиться с ЭВМ как пользователи, т. е. люди, использующие ЭВМ для решения своих задач. Поэтому нас будет интересовать принципиальная, логическая схема ЭВМ и очень мало реальные технические элементы, из которых она состоит.

19

1. *Устройство выполнения команд* — та часть ЭВМ, которая фактически выполняет все команды. Часто оно называется *арифметическим устройством*.

2. *Устройство управления* анализирует, какую команду надо выполнить, и отдает соответствующий приказ арифметическому устройству.

Когда арифметическое и управляющее устройства не отделяют одно от другого, их называют общим именем — *процессор*. Можно сказать, что процессор включает в себя устройство управления и арифметическое устройство.

3. Полученные результаты надо где-то хранить. Для этого необходимо иметь *запоминающее устройство*. Для человека запоминающим устройством может быть его память, лист бумаги, записная книжка, дневник.

4. Необходимо иметь *устройство ввода* и *устройство вывода* информации. При работе с микрокалькулятором устройством ввода являюся клавиши, устройством вывода — индикатор.

Итак мы перечислили 5 основных устройств: ввода, вывода, памяти, управления, арифметическое.

Опишем подробнее память и устройства ввода — вывода.

Память ЭВМ состоит из элементов, каждый из которых может хранить наименьшую единицу информации — 1 бит. С физической точки зрения такой элемент — некое устройство, способное находиться в одном из двух состояний. Можно считать, что это лампочка, которая либо горит, либо нет. Если лампочка горит, то в элементе хранится 1. Если лампочка погашена, то записан 0. Итак, в одном элементе можно хранить один разряд двоичного числа (один двоичный разряд).

Все вы, наверное, знаете об азбуке Морзе. В ней каждая буква, каждая цифра или специальный символ кодируется в виде последовательности точек и тире. С помощью точек и тире по телеграфу можно передавать любую информацию. Так, в азбуке Морзе букве «а» соответствует .—, букве «я» соответствует .—.—. Известный сигнал бедствия SOS кодируется в виде ...— — —... . В азбуке Морзе различным символам соответствует разное количество точек и тире. Часто встречающиеся символы кодируются короткими наборами точек и тире. Например, букве «е» соответствует код . из одного символа, а букве «ш» код — — — — из 4 символов. Такие коды называют неравномерными.

В памяти ЭВМ вся информация кодируется с помощью двух цифр 0 и 1. В отличие от азбуки Морзе в памяти современных ЭВМ для записи каждого символа используется одинаковое число битов. Если на один символ отводить К бит, то можно закодировать 2^K различных символов. Обычно для кодирования используется *байт* — единица информации, равная 8 битам. 1024 байта образуют 1 *килобайт**. (Мы привыкли, что приставка «кило» означает тысячу. Здесь 1024 взято потому, что $1024=2^{10}$.) 1024 килобайта составляют 1 *мегабайт*, а 1024 мегабайта составляют 1 *гигабайт*.

Подсчитаем, сколько потребуется памяти, чтобы записать обычный школьный учебник. Пусть на одной странице

40—50 строк. В одной строке 60 символов. Так как на запись одного символа приходится один байт, то для записи одной страницы текста потребуется около 3000 байт, или примерно 3 килобайта. Если в учебнике 330 страниц, то для его записи нужен примерно 1 мегабайт.

Для записи чисел, как правило, используется ячейка, содержащая 4 байта. Для повышения точности вычислений на многих ЭВМ для записи чисел можно использовать двойную ячейку, содержащую 8 байт.

Современные ЭВМ имеют различные виды памяти. Прежде всего память делится на *внутреннюю* (или оперативную) и *внешнюю*. Проводя аналогию с человеком, можно сказать, что внутренняя память это то, что у человека в голове, а внешняя — то, что записано в конспектах, шпаргалках, книгах. Внешняя память может быть практически неограниченной, но обращение к ней требует большого времени. Обращение к внутренней памяти происходит существенно быстрее, но ее объем ограничен. Отметим, что в большинстве случаев мы можем и записать что-либо в память, и прочитать, т. е. извлечь из памяти ранее записанную информацию. В некоторых случаях мы можем только прочитать и не имеем права записывать. Такая память называется *постоянной*. Аналогом постоянной памяти является книга или грампластинка.

Устройство ввода, как мы уже говорили, предназначено для записи программы и необходимой для ее выполнения информации в *оперативную* память ЭВМ. Обычно программа и данные записываются в формализованном виде на какой-то носитель информации. Таким носителем информации может быть *перфокарта*, или *перфокарта*, или *магнитный диск*, или *магнитный барабан*, или *магнитная лента*. На перфокартах и перфокартах один символ кодируется одной вертикальной колонкой. Если в соответствующем месте колонки имеется отверстие, то этот разряд считается равным 1. Отсутствие отверстия соответствует значению 0. У магнитных носителей информации принцип действия аналогичен записи на магнитной ленте.

* 1024 байта равны 1К байту; К — это «ка», а не «кило», именно потому, что $1024=1000$ (прим. ред.).

Намагничивая и размагничивая соответствующие участки магнитного слоя, устройство подготовки данных записывает единицы и нули.

Часто устройства ввода и вывода называют внешними устройствами, так как они связывают ЭВМ с внешним миром. ЭВМ одной модели могут содержать различный набор внешних устройств.

Перечислим основные внешние устройства. Устройства ввода: клавиатура, устройство ввода с перфокарт и с перфолент. Устройства вывода: экран дисплея, алфавитно-цифровое печатающее устройство, устройства вывода на перфокарты и перфоленты.

Часто одно и то же устройство является и устройством ввода и устройством вывода. Так работают все устройства, связанные с магнитной памятью, — *дискководы, магнитофоны* и т. д. С помощью устройства ввода и вывода человек может вести диалог с ЭВМ. Раньше, как правило, сообщения человека и ответы машины печатались на бумаге. В последнее время такой диалог ведется в основном на экране дисплея. При этом клавиатура является устройством ввода — на ней пользователь набирает свои сообщения, а экран дисплея является устройством вывода — на нем для контроля повторяются сообщения пользователя и печатаются сообщения ЭВМ.

В памяти ЭВМ хранится не только необходимая для выполнения программы информация, но и сама программа, что позволяет ЭВМ работать очень быстро.

Что было бы, если бы программа не хранилась в памяти ЭВМ? В этом случае команды ЭВМ подавались бы извне и, выполнив одну команду, машина должна была бы ждать следующую. Производительность такой машины из-за потери времени на ожидание была бы очень низкой. Аналогичная ситуация имеет место при поездке из Москвы в Ленинград на самолете. При таком путешествии нужно около часа, чтобы добраться от Москвы до аэропорта, час проходит регистрация, час длится полет, затем пассажир еще час ждет багаж и добирается до Ленинграда. Увеличение скорости полета самолета практически не уменьшит времени такого путешествия.

Высокая скорость работы ЭВМ (а современные ЭВМ работают со скоростью десятки и сотни миллионов операций в секунду) во многом достигается благодаря тому, что сама программа находится в памяти ЭВМ.

Отметим еще один важный принцип, позволяющий ЭВМ работать эффективно. В описании команды указываются не числа, с которыми надо проделать ту или иную операцию, а адреса ячеек, в которых эти числа хранятся. Если бы мы указывали сами числа, то команда умножения выглядела бы так: возьми число 13 и число 7 и перемножь их. Для перемножения других чисел пришлось бы применить другую команду. Кроме того, такую команду невозможно записать заранее, когда мы не знаем, какие числа надо перемножить. Согласно адресному принципу команда запишется так: возьми число в ячейке 1240 и число в ячейке 2320, перемножь их и запиши результат в ячейку 2360. Адресный принцип позволяет заранее составлять программу, когда начальные данные еще неизвестны. При различном содержимом ячеек 1240 и 2320 эта команда позволит перемножать разные числа.

21

2.2. Исполнитель калькулятор

Перед вами программируемый микрокалькулятор. Микрокалькуляторами называются простейшие электронные устройства, позволяющие быстро и точно проводить вычисления (от латинского слова *calcular* — считать). В настоящее время промышленность выпускает три вида микрокалькуляторов: арифметические, инженерные и программируемые. Арифметические калькуляторы «умеют» выполнять четыре арифметические операции; некоторые модели, кроме того, могут вычислять проценты и извлекать квадратные корни. Инженерные микрокалькуляторы могут кроме этого вычислять значения различных функций — синуса и косинуса, тангенса и котангенса, логарифма и экспоненты и т. д. Программируемый микрокалькулятор уже является настоящей, хотя и маленькой ЭВМ. Он способен запомнить программу и выполнить ее.

3. Основы алгоритмизации

3.1. Понятие алгоритма

В своей деятельности человеку приходится решать различные задачи. В некоторых случаях мы заранее уверены в том, что сможем решить предложенную задачу. Сможем ли мы перемножить 1234 на 3457? Хотя мы никогда не перемножали этих двух чисел, мы уверены, что если нам дадут карандаш, бумагу и достаточное количество времени, мы ее выполним. Почему мы в этом уверены? Потому, что знаем общий способ, метод решения подобного рода задач. Часто такой общий способ, метод решения какой-либо задачи называют алгоритмом. Дадим следующее определение.

22 В математике под алгоритмом понимают совокупность предписаний (правил, приказов, команд), выполнение которых приводит к решению поставленной задачи.

Разумеется, это не определение в строгом математическом смысле, а лишь объяснение понятия алгоритма.

Алгоритмы в математике изучаются очень давно. Еще в Древней Греции решались геометрические задачи на построение с помощью циркуля и линейки. Само слово алгоритм произошло от имени великого математика IX века аль-Хорезми.

Рассмотрим примеры.

Алгоритм 1. Постройка дома.

1. Сделай фундамент.
2. Сложи стены.
3. Сделай крышу.

Алгоритм 2. Получение кипятка.

1. Набери в чайник воду.
2. Зажги плиту.
3. Поставь чайник на плиту.
4. Подожди, пока вода закипит.
5. Выключи плиту.

Алгоритм 3. Деление отрезка пополам.

1. Проведи окружности из концов отрезка радиусом, равным его длине.
2. Соедини отрезком точки пересечения окружностей.
3. Отметь точку пересечения получившегося отрезка с исходным.

Рассмотрим характерные особенности этих алгоритмов.

1. **Результативность.** Выполнение алгоритма должно обязательно привести к решению поставленной задачи.

Выполнив алгоритм 1, мы обязательно построим дом, выполнив алгоритм 2 — получим кипяток, выполнив алгоритм 3 — найдем середину отрезка.

2. **Дискретность.** Решение задачи алгоритм сводит к решению отдельных, более простых задач, т. е. разбивает исходную задачу на более мелкие части.

Так, алгоритмы 1 и 3 разбивают задачи на 3 более простые подзадачи. Алгоритм 2 разбивает задачу на 5 подзадач.

3. **Конечность.** Действуя по алгоритму, мы обязательно за конечное число шагов получим решение задачи.

4. **Массовость.** С помощью алгоритма мы, как правило, можем решить не одну задачу, а любую задачу из некоторого множества.

Так, по алгоритму 1 мы можем построить не один конкретный дом, а множество различных домов. Алгоритм 3 найдет середину любого отрезка на плоскости.

5. **Конкретность.** Каждый алгоритм предназначен для некоторого исполнителя и должен быть понятным ему. Отдельные шаги алгоритма должны быть обязательно выполнимы данным исполнителем.

Так, команда на турецком языке не будет понята исполнителем — человеком, не знающим турецкого языка. Исполнитель-машина не понимает команд голоса. Нельзя разделить число на ноль. Выполнимость многих команд часто зависит от конкретного исполнителя. Команду «двигайся со скоростью 60 км в час» легко выполнит шофер на автомобиле при хорошей дороге или машинист тепловоза и не сможет выполнить пешеход, велосипедист и летчик на реактивном самолете.

6. **Однозначность.** Применение одного алгоритма в одинаковых ситуациях всегда должно приводить к одинаковому результату.

В алгоритме 1 такой однозначности нет. По этому алгоритму могут быть построены разные дома. Поэтому в строгом математическом смысле система

предписаний постройки дома алгоритмом не является. Также не является алгоритмом рецепт из поваренной книги. Два опытных повара изготовят по одному рецепту и из одинаковых продуктов разные пироги. Это произойдет потому, что некоторые отдельные предписания рецепта недостаточно конкретны и могут быть выполнены по-разному. Неоднозначность иногда бывает и в математических алгоритмах. Так, предписание «извлеки квадратный корень из 2» разными исполнителями может быть выполнено с разной степенью точности и привести к разным результатам.

Рассмотренные примеры показывают, что часто алгоритмы записываются недостаточно подробно, недостаточно однозначно. Во многих случаях большая точность и не нужна. Иногда необходимо составить более подробный алгоритм. Так, например, точность при изготовлении пирога и лекарства должна быть разной. При приготовлении пирога мы можем брать составляющие компоненты на глазок, а при изготовлении лекарства необходимо взвешивать отдельные компоненты на точных аптекарских весах.

3.2. Разработка алгоритмов и их запись

Создание алгоритма требует во многих случаях существенно больше знаний, чем исполнение уже заданного алгоритма.

Многие задачи физики и математики с точки зрения исполнителя имеют вид: подсчитать величину по соответствующей формуле. Исполнитель должен только уметь подставить данные ему числа в формулы и провести нужные вычисления. Для построения алгоритма требуются знания физики или математики.

Рассмотрим следующую задачу: определить время падения тела с высоты h . Знание физики позволяет построить следующий алгоритм.

Алгоритм вычисления времени падения. Вычислим t по формуле $t = \sqrt{\frac{2h}{g}}$, где g — ускорение свободного падения.

Вычислить время по этой формуле может и исполнитель, не знающий физики.

Рассмотрим другой пример. Требуется определить площадь треугольника, зная координаты его вершин. Пусть $A(a_1, a_2)$, $B(b_1, b_2)$, $C(c_1, c_2)$ — вершины треугольника. Площадь треугольника S можно найти с помощью такого алгоритма.

Алгоритм 1 нахождения площади треугольника.

1. Найти длины сторон треугольника

$$a = \sqrt{(b_1 - c_1)^2 + (b_2 - c_2)^2}$$

$$b = \sqrt{(a_1 - c_1)^2 + (a_2 - c_2)^2}$$

$$c = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}$$

2. Найти полупериметр треугольника $p = (a + b + c) / 2$

3. Найти площадь по формуле Герона

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

Ту же задачу можно решить проще.

Алгоритм 2 нахождения площади треугольника.

Вычислить площадь по формуле

$$S = |a_1 b_2 - a_2 b_1 + a_1 c_1 - a_1 c_2 + b_1 c_2 - b_2 c_1| / 2$$

Ясно, что алгоритм 2 существенно проще алгоритма 1.

Во многих случаях человек умеет решать задачи некоторого класса, но не может сформулировать свои умения в виде алгоритма. Так, чемпион мира по шахматам очень хорошо умеет определить, какой ход надо сделать в той или иной позиции. Однако он не может передать свои знания другому так, чтобы этот другой, действуя по какому-то алгоритму, стал чемпионом мира. И гениальный художник, и гениальный поэт не могут сформулировать алгоритмы рисования картин или сочинения стихов, хотя сами прекрасно это делают.

Если человек придумал алгоритм решения какой-либо задачи, то возникает вопрос — как его записать? В простейших случаях для небольших алгоритмов это можно сделать на обычном языке. В более сложных случаях для записи алгоритмов необходимы специальные языки. Дело в том, что обычный человеческий язык недостаточно однозначен. Так, некоторые слова (например, *коса*) могут означать разные понятия; слова «ты сегодня отличился» могут озна-

чать и одобрение, и осуждение и т. д. Мы будем рассматривать два способа записи алгоритма — язык *блок-схем* и *алгоритмический язык*.

3.3. Блок-схемы

24

Для наглядного представления алгоритмов часто используют блок-схемы. Блок-схема состоит из отдельных блоков и соединяющих их стрелок. Стрелки указывают, в каком порядке должны выполняться блоки. При записи алгоритма решения задачи в виде блок-схемы мы должны разбить процесс решения на отдельные более мелкие этапы. Каждый этап будет соответствовать одному блоку в блок-схеме. Блок-схема может быть записана с разной степенью подробности. Затем отдельные блоки мы распишем более подробно. В конечном счете мы должны уметь расписывать блок-схему так, чтобы один блок соответствовал команде, которую исполнитель может выполнить.

3.4. Основные виды блоков

Блоки начала и окончания. Каждая блок-схема должна начинаться и оканчиваться определенными блоками, которые изображаются в виде овалов. Внутри одного пишется слово «начало», а внутри другого — слово «конец» или «стоп». Блок «начало» является единственным блоком, в который не входит ни одна стрелка. Блок окончания является единственным блоком, из которого не выходит ни одной стрелки.

Рабочие блоки. Основные блоки, в которых происходит решение задачи, называются рабочими блоками и изображаются в виде прямоугольников. Внутри каждого такого прямоугольника записывается то, что должен выполнить этот блок. Из каждого такого блока выходит одна стрелка, которая укажет на блок, который должен выполняться следующим.

Изложенные в 3.1. алгоритм постройки дома можно изобразить следующим образом (рис. 1).

В данном случае блок-схема состоит из 5 блоков. При выполнении алгоритма всегда начинаем с блока «начало». Затем, идя по стрелке, мы придем к блоку 2, далее 3, 4 и 5. Дойдя до блока «стоп», исполнитель закончит свою работу.

Как составлять блок-схемы? Нельзя составить блок-схему решения задачи, которую сам не умеешь решать. Надо спросить себя: как я буду решать эту задачу? Необходимо представить себе, что надо сделать для ее решения, и свои ответы записать в виде блок-схемы. Каждая выделенная более мелкая часть задачи будет являться отдельным блоком. Затем надо определить, в каком порядке следует выполнить блоки. Иногда блоки можно расставить в любом порядке, но чаще порядок блоков не может быть произвольным. Нельзя строить стены, не построив фундамента, и нельзя строить крышу, не построив стен. записать в виде блок-схемы. Каждая выделенная более мелкая часть задачи будет являться отдельным блоком. Затем надо определить, в каком порядке следует выполнить блоки. Иногда блоки можно расставить в любом порядке, но чаще порядок блоков не может быть произвольным. Нельзя строить стены, не построив фундамента, и нельзя строить крышу, не построив стен.

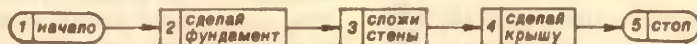
*Рабочий блок
в вычислительных задачах*

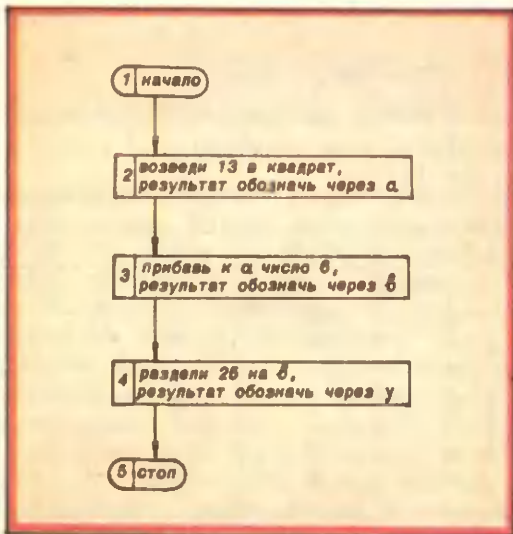
Пусть мы хотим вычислить y по формуле

$$y = \frac{25}{13^2 + 6}$$

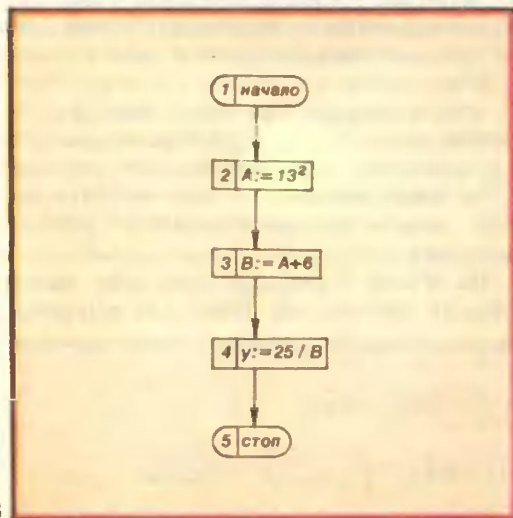
Это можно сделать при помощи такой блок-схемы (рис. 2).

Чтобы не писать слова «результат обозначь» и т. д., для записи таких вычислительных блоков используется *знак присваивания*. Он изображается





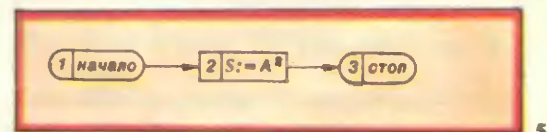
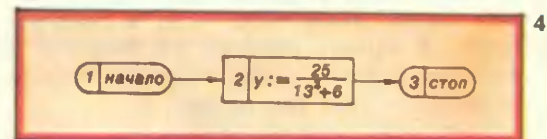
в виде $:=$ (двоеточие и знак равенства) и означает, что необходимо вычислить величину по формуле, записанной справа от знака равенства, и результат запомнить как новое значение величины, название которой записано слева от знака равенства. С использованием знака присваивания блок-схема предыдущей задачи примет вид (рис. 3). При выполнении блока 2 переменная А примет значение 169. При выполнении блока 3 переменная В примет значение 175. При выполнении блока 4 величина у примет значение $1/7$. После этого выполнение алгоритма закончено.



Как уже отмечалось, блок-схемы можно писать с разной степенью подробности. Блок-схема, нарисованная на рис. 3, выполнена очень подробно: в одном рабочем блоке записано одно арифметическое действие. Во многих случаях такая подробность не обязательна и в одном блоке можно объединять несколько действий. Такая блок-схема тоже будет правильной (рис. 4).

Блок ввода. В предыдущем примере по заданной блок-схеме мы могли решить единичную задачу. Во многих случаях необходимо составить алгоритм и блок-схему решения некоторой общей задачи.

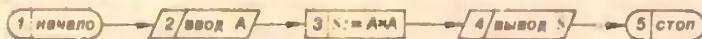
Пример: «Вычислить площадь квадрата со стороной 13». Это единичная задача (рис. 5).



«Вычислить площадь квадрата со стороной А» — это общая задача. Задавая различные значения А, будем иметь различные единичные задачи. В этом случае блок-схема позволит найти площадь любого квадрата по заданной длине А его стороны.

При нахождении площади конкретного квадрата необходимо указать значение А. При $A=3$ $S=9$, при $A=5$ $S=25$ и т. д. Таким образом, в данной задаче число А является начальным данным, позволяющим конкретизировать задачу. Для задания начальных данных используется специальный блок, называемый блоком ввода. Он изображается на блок-схеме в виде параллелограмма. Внутри этого блока мы будем писать слово «ввод» и указывать имена переменных, значение которых надо задать.

Блок вывода. Часто при решении той или иной задачи нас не интересуют результаты промежуточных вычислений. Для указания, значения каких именно величин нас интересуют, используется

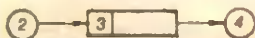


блок вывода. Как и блок ввода, он изображается в виде параллелограмма, внутри которого пишется слово «вывод» и перечисляются имена переменных, значения которых являются ответами в данной задаче. Если в блок-схеме нет блоков вывода, то исполнитель не сообщит нам результаты работы. С учетом блоков ввода и вывода блок-схему на рис. 5 можно представить в виде рис. 6.

Если нам необходимо изобразить часть блок-схемы, то ссылка на ненарисованные блоки пишется в виде окружности, внутри которой указывается номер блока. Например, если в рабочий блок 3 входит стрелка из блока 2 и выходит стрелка в блок 4, то это можно нарисовать так, как на рис. 7.

26

7



Вычисления по блок-схеме

Если какой-либо алгоритм задан в виде блок-схем, то провести вычисления по этому алгоритму может любой исполнитель, понимающий и умеющий выполнять каждый блок данной блок-схемы.

Давайте попробуем провести вычисления по блок-схеме на рис. 8.

Как будет происходить вычисление по такой блок-схеме? Для записи вычислений и контроля их правильности мы будем использовать таблицу, первый столбец которой служит для записи номера блока, выполняемого на этом шаге. Далее идет столько столбцов, сколько переменных встречается при описании алгоритма. В данном примере нужно два

столбца — для переменных A и B (табл. 1).

Таблица 1

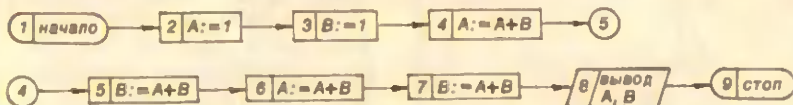
№ п/п	№ блока	A	B
1	1		
2	2	1	
3	3		1
4	4	2	
5	5		3
6	6	5	
7	7		8
8	8	5	8
9	9		

Логический блок. До сих пор отдельные шаги алгоритма выполнялись в строго определенном, заранее фиксированном порядке. Такие алгоритмы и блок-схемы называются линейными. Во многих случаях при составлении алгоритма мы должны учитывать дополнительные факторы, неизвестные в момент его составления.

Пшеницу какого сорта посеять в этом году? Если ожидается засушливое лето, то надо отдать предпочтение засухоустойчивым сортам, если ожидается влажное лето, то более влаголюбивым. Какую одежду одеть завтра на прогулку? Ответ на этот вопрос зависит от погоды. Как провести субботний вечер? Ответ зависит часто от того, удастся ли купить билет в театр, цирк или кино, что будет показано по телевизору, пригласит ли кто-нибудь из друзей к себе в гости и т. д.

Эти и другие примеры приводят к необходимости научиться записывать в алгоритме проверку каких-либо условий и в зависимости от их выполнения или невыполнения совершать разные действия.

На языке блок-схем для этих целей служит логический блок. Он рисуется





в виде ромба. Внутри этого блока записывается некоторое условие, которое может выполняться (быть справедливым, истинным, соответствовать действительности) или не выполняться (быть ложным, неверным, не соответствовать действительности). Так, условие $2 \leq 3$ справедливо, истинно. Условие $x > 3$ справедливо, если $x=4$, и ложно, если $x=2$. Из логического блока должно выходить две стрелки. Около одной из них пишется слово «да», около другой — слово «нет». Если условие, записанное внутри логического блока, выполнено (истинно), то мы идем по той стрелке, около которой написано слово «да». Если условие, записанное внутри логического блока, не выполняется (ложно), то идем по той стрелке, около которой написано слово «нет».

Рассмотрим примеры (рис. 9).

Из блока 1 мы выходим по стрелке «да», так как условие, записанное в этом блоке, справедливо, истинно. Условие, записанное в блоке 2, может быть и истинным, и ложным. Мы не можем заранее сказать, по какой стрелке мы выйдем из блока 2. Отметим также, что блок 2 записан недостаточно подробно. Что такое хорошая погода? Исполнитель, выполняющий данный алгоритм, должен иметь четкие инструкции, какую погоду считать хорошей, а какую плохой.

Задача 1. Отметим, какие из следующих условий истинны, а какие ложны. Для тех условий, которые могут быть и истинными, и ложными, привести соответствующие примеры.

1. $5 > 6$
2. $\sqrt{10} > \pi$
3. $x^2 + y^2 \leq 3$
4. Наш класс дружный
5. 1987 год високосный
6. Если номер года делится на 4, то этот год високосный
7. Число 5 простое
8. Простое число не может делиться на 2.

Задача 2. Придумайте сами аналогичные примеры.

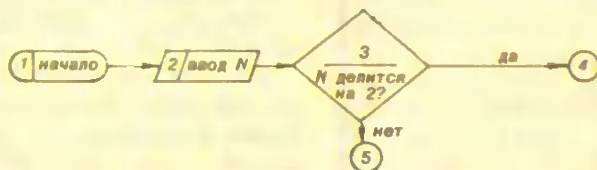
Составление блок-схем с логическими блоками

27

Задача. Определить, является ли данное двузначное число простым или составным.

Напомним, что натуральное число называется простым, если делится только на единицу и на само себя. Если число делится нацело еще на какое-нибудь натуральное число, то оно называется составным. Отметим, что 1 не является ни простым, ни составным числом.

Как проверить, является ли данное число составным? Одним из простейших способов является деление этого числа на различные целые числа. Если наше число разделится на что-либо нацело, то оно не простое. На какие числа необходимо делить? В общем случае при проверке числа N достаточно проверить все простые делители от 2 до \sqrt{N} . В данном случае для проверки любого двузначного числа достаточно

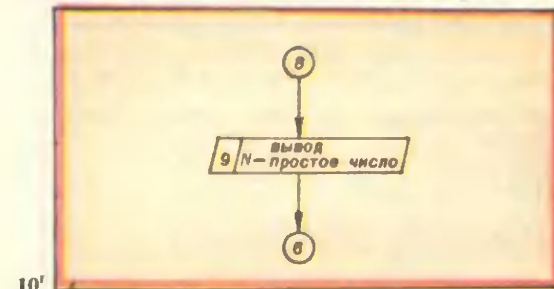
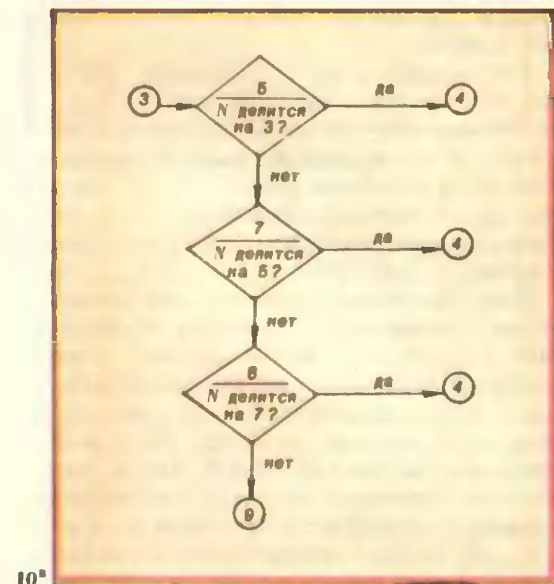
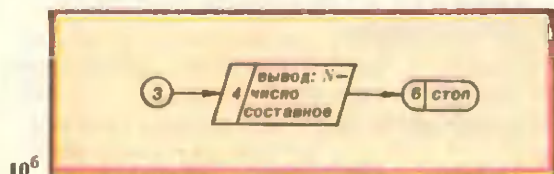


проверить, что оно не делится ни на 2, ни на 3, ни на 5, ни на 7. Если хотя бы раз число разделится нацело, то оно не простое, и дальнейшую проверку можно не проводить. Это словесное описание алгоритма. Давайте запишем этот алгоритм в виде блок-схемы (рис. 10, а, б, в, г).

Оформим выполнение этого алгоритма в виде табл. 2. При этом в таблице появится еще один столбец — проверка логического условия. Выполним алгоритм по блок-схеме при $N=37$.

Отметим, что в ранее рассмотренных линейных алгоритмах номер блока и номер по порядку всегда совпадали. Теперь порядок выполнения блоков не со-

№ п/п	№ блока	N	Проверка условия
1	1	37	37 делится на 2. Условие ложно, идем к 5 37 делится на 3. Условие ложно, идем к 7 37 делится на 5. Условие ложно, идем к 8 37 делится на 7. Условие ложно, идем к 9
2	2		
3	3		
4	5		
5	7		
6	8	37 простое число стоп	
7	9		
8	6		



ответствует их номерам и при разных начальных данных может быть разным.

Задания.

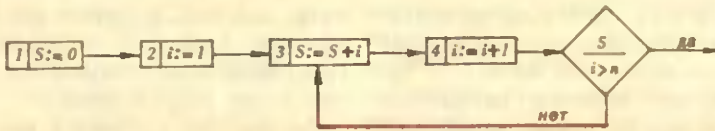
1. Выполнить этот алгоритм при нескольких различных значениях N . Результаты записать в виде таблицы.

2. Будет ли блок-схема давать правильный ответ для трехзначных чисел?

3. Как изменить блок-схему, чтобы она давала правильный ответ при любом N от 20 до 200?

Циклы. До сих пор каждая часть алгоритма выполнялась только один раз. Чтобы описать алгоритм, требующий для выполнения 100 шагов, приходилось писать блок-схему, состоящую из 100 блоков. На обычном языке у нас часто есть возможность коротко сформулировать большое по объему задание. Мы можем сказать: «делай 100 раз» или «делай, пока не добьешься успеха». При записи на языке блок-схем тоже есть такая возможность. Существуют и способы организации алгоритмов, при которых определенные шаги повторяются много раз. Такие повторяющиеся шаги алгоритма называются циклами, а алгоритмы, блок-схемы и программы, содержащие циклы, называются циклическими. На языке блок-схем циклы можно отобразить, не привлекая новых видов блоков, а используя логические блоки.

Рассмотрим следующую блок-схему. В ней нет начала и конца, блоков ввода и вывода. Это возможно в том случае, если мы изображали не всю блок-схему, а какой-либо ее фрагмент (рис. 11).



Как будет «работать» такая блок-схема при $n=4$? Сначала $S=0$, $i=1$. Затем S станет равно 1, $i=2$. При выполнении блока 5 произойдет проверка условия $i > n$, т. е. $2 > 4$. Так как это условие ложно, то мы пойдем по той стрелке, у которой записано слово «нет», и снова выполним блок 3. S станет равно 3, i примет значение 3. Снова проверим условие, оно опять ложно и т. д. Блоки 3, 4, 5, записанные в блок-схеме по одному разу, при выполнении алгоритма выполняются несколько раз. В результате работы этого алгоритма будет подсчитана сумма $S=1+2+\dots+n$. Результаты вычисления при $n=4$ запишем в табл. 3.

Таблица 3

№ п/п	№ блока	S	i	n	Проверка условия
1	1	0		4	
2	2		1		
3	3	1			
4	4		2		$2 > 4$, неверно, идем к 3
5	5	3			
6	3		3		
7	4				
8	5	6			$3 > 4$, неверно, идем к 3
9	3		4		
10	4				
11	5	10			$4 > 4$, неверно, идем к 3
12	3		5		
13	4				
14	5				$5 > 4$, верно, идем к 6

Задание.

1. Выполнить блок-схему при $n=5$.
2. Сколько блоков выполнится в блок-схеме при $n=100$.
3. Что надо изменить в блок-схеме, чтобы вычислить сумму $S=1^2+2^2+\dots+n^2$.

4.3. О быстродействии ЭВМ

Первые ЭВМ имели быстродействие около 1000 операций в секунду. С развитием вычислительной техники их скорость росла и сейчас достигает десятков и сотен миллионов операций в секунду. Беспредель ли этот рост скорости или каждое следующее достижение дается все более дорогой ценой?

Оказывается, что скорость ЭВМ не безгранична. Как известно из курса физики, самой большой из скоростей является скорость света в вакууме, равная $3 \cdot 10^{10}$ см/с. Этот предел скорости ограничивает и быстродействие ЭВМ. Если для совершения одной операции необходимо электрону пройти расстояние, равное диаметру атома водорода (10^{-8} см), то на это потребуется время

$$t = S/v = 10^{-8} \text{ см} / 3 \cdot 10^{10} \text{ см/с} = \frac{1}{3} \cdot 10^{-18} \text{ с}$$

Это значит, что быстродействие ЭВМ не может быть больше, чем $3 \cdot 10^{18}$ операций в секунду. Нужны ли такие большие скорости?

Предположим, что ученые создали сверхЭВМ с быстродействием 10^{18} оп/сек. Рассмотрим следующую задачу. Пусть на станке необходимо обработать n деталей, а мы хотим установить определенный порядок их обработки. Первой мы будем обрабатывать любую из деталей, второй — любую из оставшихся $n-1$ и т. д. Общее количество вариантов обработки равно $n! = n \cdot (n-1) \cdot \dots \cdot 3 \cdot 2$. Предположим, что мы хотим перебрать все варианты и выбрать оптимальный. Будем считать, что на рассмотрение одного варианта нам потребуется одна операция. Сколько времени нам потребуется на перебор всех $n!$ вариантов?

Значение $n!$ можно приближенно подсчитать по формуле Стирлинга $n! \approx \sqrt{2\pi n} \cdot n^n \cdot e^{-n}$

При $n=10$ число вариантов примерно 3,6млн., т.е. $3 \cdot 10^6$ и ЭВМ справится с задачей за миллиардные доли секунды. При $n=20$ число вариантов $2 \cdot 4 \cdot 10^{18}$ и ЭВМ затратит на перебор вариантов 1 секунду. При $n=30$ число вариантов $2 \cdot 6 \cdot 10^{32}$. На решение этой задачи потребуется порядка 10^{14} секунд. Один год содержит $3 \cdot 10^7$ секунд, поэтому даже на такой сверхсовременной ЭВМ нам потребуется для решения больше миллиона лет.

В настоящее время одним из резервов увеличения быстродействия ЭВМ является создание многопроцессорных ЭВМ. Создав систему, где одну задачу будет одновременно решать десять или сто ЭВМ, можно значительно ускорить время выполнения задачи. Однако и такое повышение скорости не беспредельно. Если одна ЭВМ будет иметь размеры 1 мм^3 , то в системе размером с земной шар будет примерно 10^{30} таких машин, что позволит увеличить быстродействие не более чем в 10^{30} раз. Даже такая система не сможет перебрать 50! вариантов, так как $50! \approx 3 \cdot 10^{64}$ и на такой перебор потребуется более миллиона лет.

6.8. Об эвристических алгоритмах

Будем считать, что для двух любых вариантов мы всегда можем сказать, какой из них лучше (или сказать, что они одинаковые). Если число вариантов невелико, то мы можем перебрать все возможные варианты и выбрать наилучший. При большом числе вариантов такой метод может оказаться неприемлемым, так как потребует слишком много времени. Характерным примером являются шахматы. Общее число всех шахматных партий конечно. Теоретически можно перебрать все возможные продолжения и доказать, что в данной позиции при правильной игре обеих сторон белые выигрывают, или черные выигрывают, или партия закончится ничьей. Однако это можно сделать только в сравнительно простых позициях, где либо одна из сторон быстро получает мат, либо число фигур невелико. В подавляющем большинстве позиций дать точный ответ, как закончится партия, невозможно. Поэтому почти

всегда гроссмейстер не может гарантировать, что ход, который он сделал, сильнейший в данной позиции. Поэтому и гроссмейстеры ошибаются, делают плохие ходы и проигрывают.

Во многих задачах у нас нет точного алгоритма решения задачи, отыскивающего за приемлемое время самый лучший вариант из возможных. При этом приходится отказаться от поиска наилучшего варианта и искать какой-то достаточно хороший. Алгоритмы, отыскивающие не обязательно оптимальный, а какой-то «достаточно хороший» вариант, часто называют *эвристическими*.

Рассмотрим следующую задачу. Два пирата нашли клад с драгоценностями. Известно количество драгоценных камней и стоимость каждого камня. Как разделить эти камни между собой?

При небольших n можно перебрать все возможные варианты и выбрать наилучший. Из двух вариантов раздела лучшим будем считать тот, для которого модуль разности между стоимостями куч меньше. Число вариантов дележа растет, как 2^n . При $n=10$ число вариантов уже 1024, при $n=20$ число вариантов больше миллиона. Какая бы быстрая ЭВМ у нас ни была, при больших n перебрать все варианты невозможно.

Необходимо придумать какой-либо эвристический алгоритм, который будет делить быстро, хотя и не всегда наилучшим способом.

Простейшим эвристическим алгоритмом является алгоритм «клады в меньшую кучу». Он заключается в том, что сначала камни упорядочивают по стоимости и дележ начинают с самых дорогих. Пусть $n=10$. Стоимости камней представим как элементы массива. Пусть массив A задан таблицей 4.

Посмотрим, как будет выполняться такой алгоритм. Сначала первый пират возьмет себе самый дорогой камень, т. е. пятый камень стоимостью 89. Затем второй пират возьмет самый дорогой камень из оставшихся, т. е. десятый ка-

Таблица 4

A(1)	A(2)	A(3)	A(4)	A(5)	A(6)	A(7)	A(8)	A(9)	A(10)
13	21	34	55	89	44	33	77	10	87

Таблица 5

Но- мер шага	Номер и стоимость самого дорогого камня из оставшихся		Номер пирата, взявшего камень	Суммарная стоимость после этого шага	
				у первого	у второго
1	5	89	1	89	0
2	10	87	2	89	87
3	8	77	2	89	164
4	4	55	1	144	164
5	6	44	1	188	164
6	3	34	2	188	198
7	7	33	1	221	198
8	2	21	2	221	219
9	1	13	2	221	232
10	9	10	1	231	232

мень стоимостью 87. Так как второй взял меньше, то теперь он берет самый дорогой камень из оставшихся камней, т. е. камень 8. Теперь у первого меньше. Он берет камень 4 и т. д. Запишем процесс дележа в виде таблицы 5.

Таким образом, в результате дележа по этому алгоритму первый пират получил камни 5, 4, 6, 7 и 9. Второй — камни 10, 8, 3, 2, 1. Разница в стоимостях долей равна 1. Так как общая сумма нечетная, то поровну разделить невозможно. В этом примере эвристический алгоритм является наилучшим способом раздела добычи. Однако это происходит не всегда. Так, если $n=5$ и $A_1=5$, $A_2=A_3=4$, $A_4=3$, $A_5=2$, то общий вес равен 18 и легко разделить камни поровну. Надо одному взять 5 и 4, а другому 4, 3 и 2. Однако, действуя по алгоритму, мы сначала дадим первому камень 1, затем камни 2 и 3 отдадим второму, камень 4 первому и камень 5 любому пирату. В результате стоимость добычи у одного пирата равна 10, а у другого 8.

Почему мы так подробно остановились на задаче о двух пиратах? Дело в том, что подобные задачи в еще более сложной формулировке возникают ежедневно и их надо уметь быстро решать. Как перевезти груз на двух автомобилях?

6.9 Сверхточные вычисления

Как уже неоднократно подчеркивалось, во многих случаях ЭВМ считает

приближенно. В некоторых случаях точность вычислений можно увеличить, записывая одно число в нескольких ячейках памяти.

Рассмотрим следующую задачу. Пусть необходимо вычислить

$$n! = n \cdot (n-1) \cdot \dots \cdot 2 \cdot 1.$$

Это можно сделать с помощью следующей циклической программы

```
10 REM N! — N ФАКТОРИАЛ
20 INPUT N
30 LET P=1
40 FOR I=2 TO N
50 LET P=P*I
60 NEXT I
70 PRINT "N="N, "N!="P
80 END
```

При работе с этой программой можно получить точные значения лишь при небольших значениях n . Это связано с тем, что числа в памяти ЭВМ записываются как числа с плавающей запятой, т. е. в виде $m \cdot 10^p$, причем для записи мантииссы используется 7 десятичных разрядов. Числа, в десятичной записи которых используется более 7 цифр, округляются и записываются приближенно. Таким образом с помощью рассмотренной выше программы мы можем узнать лишь первые 7 цифр числа $n!$. При $n \geq 14$ замена остальных цифр нулями даст нам приближенное значение. Отметим еще раз, что для большинства практических расчетов такая точность работы ЭВМ вполне достаточна.

Послесловие к журнальной публикации

Рукопись, представленная автором на конкурс, не является законченным учебником. Ее можно рассматривать как подготовку будущего учебника или пособия для учителей и учеников.

В рукописи чуть больше 200 страниц. Из них около 40 посвящены ПМК и около 40 Бейсику. При возможности выбора надо предпочесть Бейсик, но при отсутствии ЭВМ с Бейсиком и наличии программируемых микрокалькуляторов их надо обязательно использовать. Основная часть рукописи посвящена общим вопросам алгоритмизации. Предложены

два способа записи алгоритмов — язык блок-схем и алгоритмический язык из действующего школьного учебника. При этом не обязательно изучать и то и другое. Выбор способа записи алгоритмов нужно предоставить школе, учителю. По мнению автора, начинать надо с языка блок-схем ввиду его большей наглядности. Овладев букварем и столкнувшись с более сложными задачами, человек в дальнейшем может перейти к языкам высокого уровня, но начинать надо с букваря.

Имея существенно меньшую стоимость, ПМК помогает продемонстрировать:

наличие исполнителя, понимающего и выполняющего наши просьбы, приказы, команды;

32 возможность давать исполнителю не отдельные приказы, а составить из этих приказов программу, запомнить ее, а затем по специальной команде выполнить;

необходимость четкого понимания того, какую задачу мы хотим решить, что будет начальными данными, а что результатом;

понятие алгоритма с основными видами алгоритмов: линейными, разветвляющимися, циклическими;

возможность многократного применения один раз составленной программы;

возможность решить задачу по составленной кем-то программе, не зная и не понимая алгоритма ее решения.

Г. Н. Копылов

Читатель может получить представление о структуре учебника из приведенного ниже содержания.

1. Введение
- 1.1. Информация и информатика
- 1.2. Исполнитель
- 1.3. Исполнитель Карандаш
2. Устройство ЭВМ
- 2.1. Принципиальная схема ЭВМ
- 2.2. Исполнитель калькулятор
- 2.2.1. Арифметические операции
- 2.2.2. Лабораторная работа № 1
- 2.2.3. Действия с памятью
- 2.2.4. Вычисление стандартных функций
- 2.2.5. Невозможные операции. Сигнал ошибки
- 2.2.6. Числа с плавающей запятой. Самое большое число
- 2.2.7. Вычисление тригонометрических функций
- 2.2.8. Возведение в степень. Вычисление логарифмов

- 2.2.9. Лабораторная работа № 2
- 2.2.10. Вычисление по формулам. Понятие о стеке
- 2.2.11. Приближенный характер вычислений на ЭВМ
- 2.2.12. Лабораторная работа № 3
3. Основы алгоритмизации
- 3.1. Понятие алгоритма. Свойства алгоритма
- 3.2. Разработка алгоритмов и их запись
- 3.3. Блок-схемы
- 3.4. Основные виды блоков
- 3.5. Рабочий блок в вычислительных задачах
- 3.6. Вычисления по блок-схеме
- 3.7. Лабораторная работа № 4
- 3.8. Логический блок
- 3.9. Составление блок-схем с логическими блоками
- 3.10. Алгоритмический язык
- 3.11. Запись разветвляющихся алгоритмов на алгоритмическом языке
- 3.12. Реализация сложных условий блок-схемами
- 3.13. Проверка сложных условий в алгоритмическом языке
- 3.14. Циклы
- 3.15. Составление циклических программ на алгоритмическом языке
- 3.16. Составление циклических блок-схем
4. Основы вычислительной техники
- 4.1. Поколения ЭВМ
- 4.2. Операционная система
- 4.3. О быстродействии ЭВМ
5. Основы программирования
- 5.1. Программы на МК
- 5.1.1. Составление простейших программ на МК
- 5.1.2. Лабораторная работа № 5
- 5.1.3. Составление разветвляющихся программ на МК
- 5.1.4. Лабораторная работа № 6
- 5.2. Бейсик ДВК-1
- 5.2.1. Включение ЭВМ. Знакомство с клавиатурой
- 5.2.2. Работа в режиме исполнения
- 5.2.2.1. Оператор печати
- 5.2.2.2. Оператор присваивания
- 5.2.3. Составление простейших программ
- 5.2.4. Лабораторная работа № 7
- 5.2.5. Реализация на языке Бейсик разветвляющихся алгоритмов и логических блоков
- 5.2.6. Запись циклических алгоритмов на языке Бейсик
- 5.2.7. Массивы
- 5.2.8. Бейсик ДВК-2
6. Решение задач на ЭВМ
- 6.1. Этапы решения задач на ЭВМ
- 6.2. Составление простейших игровых программ
- 6.3. Решение уравнений
- 6.4. Вычисление площади
- 6.5. Вычисление площади криволинейной трапеции на языке Бейсик
- 6.6. Вычисление площади криволинейной трапеции с помощью МК
- 6.7. О трудоемкости алгоритмов
- 6.8. Об эвристических алгоритмах
- 6.9. Сверхточные вычисления
- 6.10. Библиотека алгоритмов и программ ЭВМ в обществе
7. Обучающие и контролирующие программы
- 7.1. Продажа железнодорожных билетов
- 7.2.

Решения к «Первым шагам»

Чтобы сделать использование опубликованного в прошлом номере журнала задачника* более доступным для тех учителей информатики, кто еще не имеет большого опыта преподавания, редакция публикует некоторые решения. К сожалению, отсутствие вычислительной техники сильно сужает наши возможности, поэтому мы ограничились теми задачами, которые включают интересные с точки зрения программирования моменты и в то же время имеют достаточно компактные решения.

Запись с помощью алгоритмической нотации легко перевести на конкретный язык программирования, хотя в ней и не используются операторы ввода-вывода. Для наглядности вывод включен в текст решения задачи 6; во всех задачах подразумевается необходимость ввода аргументов основных алгоритмов.

Задача 6.

алг СТЕПЕНЬ (цел A, цел N, цел P)

арг A, N
рез P

нач

выбор

при $N < 0$: ВЫВОД («Ошибка при вводе N»)

при $N = 0$: ВЫВОД («A в степени N равно 1»)

при $N > 0$: ВЫЧИСЛЕНИЕ(A, N, P)

все

кон

алг ВЫЧИСЛЕНИЕ(цел A, цел N, цел P)

арг A, N
рез P

нач цел I, K, Ю

Ю:=0; P:=1;

для I от 1 до N

нц

если $A > 0$

то для K от 1 до A

нц

Ю:=Ю+P

кц

иначе для K от A до -1

нц

Ю:=Ю-P

кц

все

P:=Ю

Ю:=0

кц

кон

Здесь в основном алгоритме проверяется правильность задания N (полезно также пре-

дусмотреть проверку N и A на целочисленность, однако способ этой проверки зависит от конкретной ЭВМ и нами не приводится) и вызывается вспомогательный алгоритм. Во вспомогательном алгоритме ВЫЧИСЛЕНИЕ в цикле по I вычисляются нарастающие суммы — A, A², A³, ..., A^N. Каждая очередная степень A хранится в переменной P и используется для вычисления следующей степени.

Задача 11. В этой задаче можно обойтись не только без дополнительных массивов, но даже и без переменных, за исключением параметра цикла. Воспользуемся для этого следующим алгоритмом обмена значений переменных A и B:

A:=A+B

B:=A-B

A:=A-B

алг ОБМЕН (вещ таб A[1:D], цел D)

арг A, D

рез A

нач цел K

K:=2

пока K ≤ D

нц

A[K-1]:=A[K-1]+A[K]

A[K]:=A[K-1]-A[K]

A[K-1]:=A[K-1]-A[K]

K:=K+2

кц

кон

Задача 17. Троичная система счисления базируется на цифрах 0, 1, 2. При любом x $0 \cdot x = 0$, $1 \cdot x = x$. Ну а $2 \cdot 2$ в троичной системе равно 11.

Для восьмеричной системы приведем только часть таблицы умножения:

3 · 0 = 0 3 · 3 = 11 3 · 6 = 22

3 · 1 = 3 3 · 4 = 14 3 · 7 = 25

3 · 2 = 6 3 · 5 = 17

Задача 20. Основная трудность в этой задаче — избежать чрезмерного возрастания значений переменных. Уже $13! \approx 10^{10}$. Такое целое число, как правило, не может быть представлено в ЭВМ. Если же использовать вещественные переменные, то при их изменениях на десятки порядков из-за ошибок округления сильно пострадает точность вычислений. Поэтому, чтобы не ограничивать себя лишь малыми значениями N, воспользуемся для вычислений следующей формулой:

$$\frac{J!}{J!(1-J)!} = \frac{(J+1) \cdot (J+2) \cdot \dots \cdot 1}{(1-J)!} = \frac{J+1}{1} \cdot \frac{J+2}{2} \cdot \dots \cdot \frac{1}{1-J}$$

* Климов Д., Петров М., Урнов В. «Первые шаги» и «Большие проекты».

Для удобства реализации переписем последнюю часть формулы, «перевернув» знаменатель относительно числителя,

$$\frac{I!}{J!(I-J)!} = \frac{J+1}{I-J} \cdot \frac{J+2}{I-J-1} \cdot \dots \cdot \frac{1}{1},$$

и в соответствии с ней построим вспомогательный алгоритм.

алг КОЭФФИЦИЕНТ (цел I, цел J, вещ P)

```

арг I, J
рез P
нач цел И, К, Л
P:=1
если J>I-J
  то K:=I-J; Л:=J+1
  иначе K:=J; Л:=I-J+1
все
для И от Л до I
нц
  P:=P·И/(К-(И-Л))
кц
кон
  
```

34

Здесь предусмотрено сокращение первоначальной формулы вычисления биномиального коэффициента на $J!$ или на $(I-J)!$ в зависимости от того, что нам выгоднее.

Основной алгоритм будет выглядеть просто.

алг БИНОМ (вещ таб С[0:N, 0:N], цел N)

```

арг C, N
рез C
нач цел M, H
для M от 0 до N
нц
  для H от 0 до N
  нц
    выбор
      при M<N: C[M, H]:=0
      при M=N: C[M, H]:=1
      при M>N: КОЭФФИЦИЕНТ
        (M, H, C(M, H))
    все
  кц
кц
кон
  
```

Задача 23. Дополнительные массивы здесь, конечно, не нужны; не нужен даже, в отличие от задачи 11, параметр цикла.

алг СЖАТИЕ (вещ таб A[1:N], цел N)

```

арг A, N
рез A
нач если int (N/2) · 2=N
  то для A[N] от 2 до N/2
  нц
    A[A[N]]:=A[A[N]·2-1]
  кц
  иначе для A[N-1] от 2 до (N+1)/2
  нц
    A[A[N]]:=A[A[N]·2-1]
  кц
все
кон
  
```

Здесь функция $\text{int}(x)$ — имеющаяся в том или ином виде на любой ПЭВМ функция вычисления наибольшего целого, не превосходящего x . Алгоритм работоспособен при $N>3$.

Задача 40. Ее удобно решать, используя тот факт, что каждый символ в ЭВМ представлен неким числом. Будем считать, что у нас есть вспомогательный алгоритм вычисления значения функции цел КОД (лит A), который вычисляет код символа A (его конкретная реализация зависит от используемой ЭВМ и не может быть описана с помощью алгоритмической нотации). Будем считать также, что код буквы A равен 1, буквы Б — 2 и т. д. Всего разных кодов будет не более 60 (с учетом знаков препинания и цифр). Теперь можно составлять алгоритм решения задачи.

алг СТАТИСТИКА (лит A, цел таб P [1:60])

```

арг A
рез P
нач цел И
для И от 1 до 60
нц
  P[И]:=0
кц
для И от 1 до длин(A)
нц
  P[КОД(A[И:И])]:=P[КОД(A[И:И])] + 1
кц
кон
  
```

Задача 60.

алг СЖАТИЕ НУЛЕЙ (вещ таб V[1:M], вещ таб P[1:M], цел M, цел N)

```

арг V, M
рез P, N
нач цел K, Л
N:=1; K:=0
для Л от 1 до M
нц
  выбор
    при V[Л]=0: K:=K+1
    при V[Л]≠0 и K>0: P[N]:=0;
      P[N+1]:=K;
      P[N+2]:=V[Л]; N:=N+3; K:=0
    при V[Л]≠0 и K=0: P[N]:=
      V[Л]; N:=N+1
  все
кц
кон
  
```

Задача 84. Пусть x — масса яблока. $Nx>M$, $x>M/N$. Таким образом, ПЭВМ для решения этой задачи совсем не нужна, однако не все замечают это сразу. К сожалению, и в жизни электронно-вычислительная техника применяется далеко не всегда адекватно, так что «провокационные» задачи, подобные рассмотренной, очень полезны.

А. АРХАНГЕЛЬСКИЙ

Мир ЭВМ

6. Диски, ленты, дисководы

6.1. Для чего нужно форматирование

Сегодня мы в очередной раз включили ДВК, и вскоре на экране зажглась надпись

?BOOT — U — I/O ERROR

— ошибка ввода / вывода. Мы видели ее в прошлый раз в нулевой блоке при изучении загрузчика.

Ну и что делать? Попробуем еще раз?

Но надпись появилась снова. Что ж, будем разбираться.

Электронная часть накопителя на гибких дисках в порядке — ведь нулевой блок носителя, в котором записано это сообщение, прочитан. Попробуем загрузить систему с DX1:!. Для этого в ответ на приглашение наберем D1.

Накопитель пощелкал, на экране появилась новая надпись: ?BOOT — W — ERROR READING HANDLER

6.2. Форматы носителей информации

Настал момент подробно рассказать о дискетах и дисководах.

Дискета (дискетовая кассета) похожа на гибкую грампластинку (как в журнале «Кругозор») и запакована в пластиковый пакет. Диаметр диска равен 198 мм, диаметр центрального отвер-

стия — 38,1 мм. Индикаторное отверстие в диске (диаметром 2,5 мм) предназначено для синхронизации вращения. Информация записывается в тонком магнитном слое, который нанесен на поверхность диска.

35

Как и магнитные ленты и грампластинки, дискеты очень чувствительны к грязи, пыли и отпечаткам пальцев, поэтому их упаковывают в защитные конверты размером 203×203 мм, изготовленные из материала, обеспечивающего минимальное трение, так как при чтении и записи эти защитные конверты не снимаются с дискеты.

Конверт имеет три отверстия: большое круглое в центре — для шпинделя, радиальное продолговатое — для головки записи/чтения и маленькое, несколько в стороне от центра, — для индикаторного фотодатчика (фото 1).

Для дополнительной защиты дискеты в перерывах между работой применяется дополнительный конверт из тонкого картона, похожий на те, в которых хранятся грампластинки. Обычно на нем напечатаны правила обращения с дискетой.

Гибкий диск был разработан в середине 60-х гг. фирмой IBM. Как и многое другое, разработанное там, он стал стандартом de facto в вычислительной технике. Несмотря на то что сейчас появились более компактные дискеты с большей емкостью, дискета стандарта

Продолжение. Начало см.: Информатика и образование. 1987. № 1—3; 1988. № 3—4.



Дискета, «установленная» на полуразобранный дисковод. Индикаторное отверстие слева от центрального, прорезь для магнитной головки — справа, под рычагом с фетровым прижимом (фото 1).

IBM 3740 остается одним из основных средств совместимости по носителям между ЭВМ различных типов и фирм, так что далее речь будет идти именно о ней.

Первоначально дискеты имели магнитный слой только на одной стороне. Сейчас этот слой наносится на обе стороны, так что после переворачивания мы можем использовать информацию, записанную на обратной стороне дискеты. Однако для этого в конверте должно быть еще одно отверстие для индикаторного фотодатчика, или (что лучше) в накопителе должен быть установлен еще один фотодатчик (это позволит отличать, какой стороной вставлена дискета; правда, в RT-11 такая возможность не предусмотрена). В некоторых накопителях (например, ГМД-7022) для чтения/записи данных доступны

обе стороны одновременно.

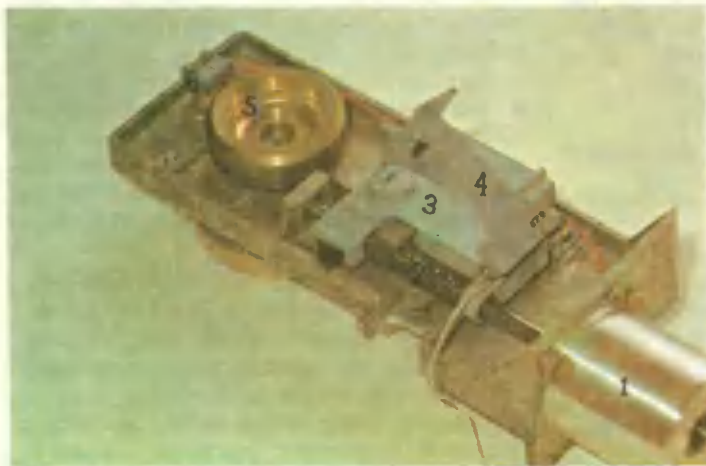
На дисках информационные дорожки расположены в виде концентрических окружностей, а не спирали, как в случае грампластинок. Эти окружности называются дорожками. Внешние дорожки длиннее внутренних, поэтому плотность упаковки информации на внутренних дорожках выше, следовательно, и вероятность сбоя или ошибки больше. Так что информацию лучше хранить ближе к краю диска.

В качестве привода дискет обычно используется синхронный двигатель со скоростью вращения шпинделя 360 об/мин. Скорость вращения должна быть стандартной, чтобы можно было обмениваться дисками, записанными на разных устройствах.

При работе ЗУ на гибких дисках магнитные головки находятся в постоянном контакте с поверхностью дискеты. Хотя это и приводит к дополнительному износу трущихся элементов, однако позволяет упростить конструкцию. Износ сильно ограничивает сроки службы как головок, так и дискет. Обычно устройства с гибкими дисками устроены так, что в паузах между операциями записи/чтения магнитная головка приподнимается, чтобы уменьшить износ. Впрочем, магнитные головки легко заменяются.

А теперь посмотрим, что у дисковода внутри (фото 2).

Устройство для управления положением головки включает шаговый дви-



Основная часть «механики» накопителя на гибких магнитных дисках:

1 — шаговый двигатель, 2 — винт, перемещающий держатель магнитной головки, 3 — сама головка (темная полоска — ее рабочая поверхность), 4 — рычаг с фетровым прижимом, 5 — шпиндель синхронного двигателя, 6 — индикаторный фотодатчик, сигнализирующий о начале очередного оборота диска (фото 2).

гатель, вращающий винт, который перемещает держатель головки вдоль поверхности дискеты.

Дискета прижимается к магнитной головке с помощью фетрового выступа. Его давление таково, что в этом месте дискета немного изгибается и как бы облегает магнитную головку (так же, как магнитная лента в магнитофоне).

Положение «дорожка 00» для магнитной головки определяется с помощью флажка, закрепленного на держателе магнитной головки и перекрывающего луч оптического датчика.

Обмен данными между дисководом и системой обработки данных обеспечивает контроллер — электронное устройство, состоящее обычно из двух частей. Первая обеспечивает обмен данными между буфером дисковода и дискетой и, как правило, встроена в накопитель. Вторая, встроена в ЭВМ, обеспечивает согласование стандартного канала передачи данных НГМД с конкретной ЭВМ.

Из-за относительно большого времени доступа к дисковому ЗУ невыгодно записывать данные, размещая их в произвольных местах на диске. Значительно меньшее среднее время доступа обеспечивает метод хранения данных в виде довольно крупных «кусков» — блоков, так как для каждого конкретного этапа обработки данных целиком вся информация, хранящаяся на дискете, никогда не требуется. Читающий книгу видит перед собой только одну страницу, необходимую ему для ознакомления с текстом. В дисковых ЗУ блок данных, которые требуются программе, читается однократно, слово за словом, и записывается в рабочем ЗУ системы, например в ОЗУ. При записи данных на диск эта процедура протекает точно так же, но только в обратном направлении. Внимание! Не перепутайте! Это совсем не те блоки, которые мы видели в программах DIR или DUMP. Их можно было бы назвать секторами, так как каждый (как правило) такой блок записывается на отдельный участок магнитной дорожки, называемый сектором. Каждый сектор имеет фиксированную длину и собственный адрес,

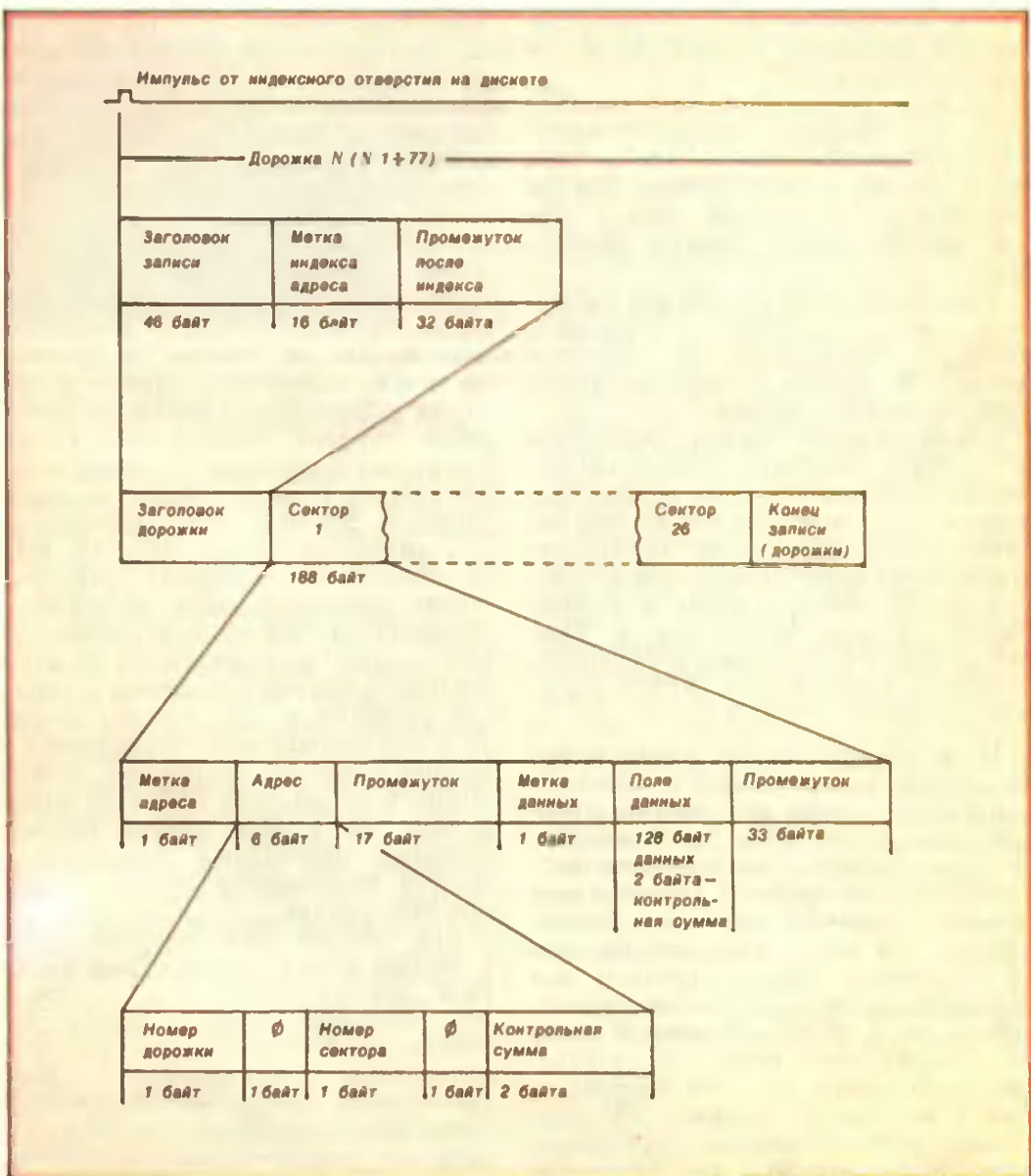
определяющий место расположения этого сектора на поверхности диска. Адрес записывается в начале каждого сектора. Контроллер не разрешает записывать в место, где хранится адрес сектора, какую-либо другую информацию. Адреса секторов можно только считывать.

Операция записи на чистый диск служебной информации (рис. 1) называется разметкой, или форматированием, диска. Разметка позволяет контролировать положение головки на дорожке, увеличить надежность записи и хранения информации, а также уменьшить объем буфера контроллера. Однако служебная информация занимает место. На гибком диске без разметки можно записать 398,68К байт информации, а с разметкой только 250,25К байт. В технических описаниях различают объем накопителя (или носителя) в форматированных или неформатированных данных. Для увеличения полезной емкости диска при сохранении достоинств разметки и стандартного формата стали использовать так называемый формат с двойной плотностью, отличие которого от формата IBM 3740 только в том, что в поле данных (области хранения информации пользователя) сектора записывается 256 байт полезной информации.

При попытке ЭВМ получить доступ к данным контроллеру сообщаются адреса нужных дорожки и сектора и код операции, которую необходимо произвести. Контроллер через устройство управления положением головки устанавливает ее над требуемой дорожкой. Затем производится поиск сектора и запись или считывание соответствующего блока данных.

На рис. 1 показана схема, описывающая стандарт IBM 3740. Его называют также стандартным IBM-форматом. Он одинаков для всех 77 дорожек. Расшифровка содержимого служебных полей в журнальном варианте опускается. Впрочем, о заголовке записи сказать стоит.

Эти 46 байтов заполнены нулями. Зачем? Дело в том, что кроме нулевых битов данных на них записаны тактовые импульсы — сигналы синхрониза-



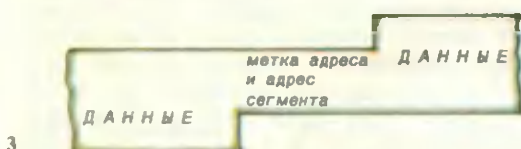
ции. По ним настраивается внутренний генератор тактовых импульсов — таймер.

Сигналы синхронизации записаны по всей длине дорожки вперемежку с данными: бит данных — бит синхронизации (всегда равный 1) — бит данных — бит синхронизации и т. д.

Взглянем на магнитную дорожку еще раз — уже не с точки зрения информатики, а с точки зрения физики (рис. 2).

Ширина дорожки 0,4 мм, расстояние между дорожками 0,1 мм. Чтобы из-за случайного сдвига магнитная головка не записала данные на соседнюю дорожку, по ее бокам и чуть сзади расположены стирающие головки.

Так как запись при работе происходит только в поле данных, а стирание — по всей дорожке, то из-за неточной работы механизма привода, неодинакового положения головки относительно оси дискеты у разных нако-



пителей (различной юстировки) служебные поля сектора (они записываются, как вы помните, только при разметке) стираются до такой степени, что прочесть их становится невозможно, в то время как данные легко читаются (рис. 3). Дисковод не может найти сектор, который следует прочесть.

При записи на несовместимом дисковом дисковом может быть испорчена не только записываемая дорожка, но и соседняя, которую задела стирающая головка.

Рассмотрим несколько типичных случаев «нечитаемости» дискет.

Если дискету неаккуратно вставили в накопитель и замяли ей край ее центрального отверстия, то дорожки будут смещены относительно головки на величину замятия. Однако в таком случае будут читаться правильно два сектора на дорожке (подумайте, почему!). Во избежание подобных неприятностей нужно вставлять дискету аккуратно и закрывать крышку дисковода не сразу, а покачивающим движением, чтобы дискета могла сдвинуться в конверте и найти свое место на конусе шпинделя.

Дискета не читается, если шаговый двигатель неправильно отсчитывает расстояние от центра вращения. При этом мы не сможем правильно прочесть ни одного сектора: дорожки будут чуть большего или меньшего диаметра. Этот случай несложный — правильно устано-

вить головку можно поворотом шагового двигателя в пределах $+15...-15$ градусов. Делается это с помощью осциллографа (или даже милливольтметра) и небольшой программки всего за пять минут.

Если не читается одна, две или несколько дорожек (как правило, в таком случае не читается и первая дорожка), то, скорее всего, кто-то записывал информацию на дискету на другом (несовместимом) дисковом дисковом. А первая дорожка не читается, потому что происходила и запись в каталог.

В таких случаях, если ошибок (или, как говорят, плохих блоков) немного и известно, что файлы содержат текстовую информацию, лучше взять заведомо хорошую, чистую (т. е. свободную от нужной информации) дискету и скопировать все с плохой дискеты на хорошую с игнорированием ошибок. В текстовом файле один блок содержит в среднем 16 строк текста, которые восстановить легче, чем весь файл. Кроме того, возможно, не читается адрес сектора или испорчен только один байт, в результате чего не совпадает контрольная сумма, т. е. ошибки незначительны. В любом случае с помощью программы DUMP можно достаточно полно установить размеры катастрофы. Программа DUMP при анализе плохого блока будет делать несколько попыток чтения, и по характерному движению головки в накопителе (многократные периодические возвраты к нулевой дорожке) его легко можно определить.

Следует также учесть, что программа DIR при своей работе использует первый блок, а программы копирования — нет.

Иногда внешние дорожки читаются заметно лучше, чем внутренние (или наоборот); чем больше расстояние между дорожками, тем заметнее разница.

Под такую дискету и дисковод не отрегулируешь — нос вытащишь, хвост увязнет.

Подобные ошибки вызываются некачественным изготовлением винта, по которому скользит головка (что легко определяется в начале эксплуатации) или износом прижимной гайки, которую легко заменить.

Часто не читаются несколько блоков, расположенных кучкой на соседних дорожках (эту кучность можно установить, заметив, что номера плохих блоков отличаются на 26). Скорее всего, кто-то пальцем прикоснулся к поверхности диска и на ней осталось жирное пятно.

Это довольно частый случай. Аккуратно поворачивая дискету в конверте, можно найти это место и стереть пятно изопропиловым спиртом. Однако эффект пятна дает и приближение магнита или провода, по которому протекает электрический ток. В этом случае информация потеряна навечно.

Подытоживая, можно сказать, что в отношении к носителям информации проявляется культура человека. Чем лучше относишься к ним, а также к оборудованию, тем надежнее будет сохраняться информация. Часто встречающийся совет — делать побольше копий — далеко не лучший. Это дорого (дискет не напасешься), нужно вовремя вносить в них (во все!) изменения, а кроме того, они с таким же успехом могут быть испорчены неаккуратным обращением.

6.3. Жесткие диски

Мы уже знаем, что к нашей машине можно подключать накопители не только на гибких дисках, но и на жестких, которые представляют другой класс запоминающих устройств.

Кассетный диск разработан также фирмой IBM и имеет два общих обозначения — IBM 2315 и IBM 5440. Он заключен в пластмассовую кассету диаметром 14 дюймов, отсюда его второе (менее распространенное) название — 14-дюймовый жесткий диск. Кассета делает удобными переноску и установку.

Сам диск выполнен из сравнительно толстого алюминия. Это позволило сделать его больше и вращать с большей скоростью, следовательно, увеличить объем информации, записываемой на него. Выше и скорость обмена информацией.

Однако то, что диск больше по размеру, не главное для увеличения объема записываемой информации. Главное

то, что изготовить его можно с более высокой точностью и крепится он на прочной ступице, которая позволяет более точно зафиксировать его на шпинделе двигателя. Все эти меры позволяют удвоить плотность дорожек — их шаг равен 0,254 мм, а ширина 0,175 мм.

Высокая плотность дорожек дает побочный результат: размеры магнитных головок уменьшились, а это увеличило их пропускную способность и плотность записи.

Все параметры отказываются взаимосвязанными — высокая скорость вращения требует и более качественного магнитного лака, а это также позволяет увеличить плотность записи.

Теперь подведем итоги.

Диски типа IBM 2315 и IBM 5440 имеют емкость от 1,2М до 6,0М байт; 100 или 200 рабочих (плюс две или четыре запасных) дорожек; вращаются со скоростью 1500 (IBM 2315) или 2400 об/мин (IBM 5440).

Что же касается форматов записи, то отличия не принципиальные. Количество секторов в зависимости от исполнения может быть 12, 16, 24, 32. В нашем случае используется 12 секторов, но в каждом записано по 512 байтов данных. Число дорожек — 200 основных и 4 запасных. В остальном принципы записи, необходимость разметки, адреса дорожки и сектора сохранились.

Вместе с диском изменился и дискотвод. Если при высокой скорости вращения прижать магнитную головку к поверхности, то она сгорит в прямом смысле слова. А если на диск попадет пылинка, то она и головку испортит, и борозду в магнитном лаке оставит.

Поэтому магнитная головка «летает» над поверхностью диска на высоте 2 мкм. Рабочая поверхность головки обрамлена достаточно большой круглой пластиной, выполняющей функции крыла. При движении диска создается воздушная подушка, предохраняющая головку от контакта с диском.

Две головки (по одной на каждую сторону диска) закреплены на рычагах; шаговый двигатель заменен на бо-

лее быструю конструкцию — электромагнитную катушку (как у большого громкоговорителя) (фото на 3-й странице обложки).

При включении дисководов диск плавно раскручивается; его скорость постоянно измеряется специальным датчиком. Когда установится нужная скорость, в дело вступают дворники — щетки, которые выдвигаются из угла копителя, заботливо очищают поверхность диска от пыли, и только после этого головки «выстреливаются» на нулевую дорожку диска.

Для экономии места и оборудования на одном приводе закрепляется два диска: один встроенный (он имеет четный номер), другой съемный (имеет нечетный номер).

По сравнению с гибкими дисками здесь более длинные рычаги головок, имеющие большую массу, и так как скорость перемещения головок очень высока, в электромагнитной катушке выделяется много тепла. От изменения температуры меняются длина рычагов и положение головок, что приводит к такому же эффекту несовершенства дорожки и головок, как у гибких дисков. Свою роль в этом аспекте играет и температура в машинном зале.

4.4. Диск-пакеты и винчестеры

Диск-пакет — это те же самые диски, что и IBM 5440, но их 11 штук и надеты они на одну ось. Конечно, кассета здесь не подходит; вместо этого на них одет колпак, который снимается только при установке в накопитель, а при переноске снизу прикручивается пластмассовая крышка. Диск-пакеты также были разработаны фирмой IBM и получили обозначение IBM 2316 (фото на 2-й странице обложки и фото 3).

Раз в них 11 дисков, значит, должно одновременно двигаться 20 головок (две поверхности являются защитными). Поэтому выделение тепла электромагнитной катушкой существенно больше. В результате диск-пакеты более чувствительны к температуре помещения.

На персональных компьютерах ДВК-3 и «Электроника-85» стоят несъемные жесткие диски, называемые винчесте-



Накопитель на диск-пакетах (тот же, что и на второй странице обложки) — вид с лицевой стороны (кожух снят) (фото 3).

рами. Пока это самое удобное внешнее запоминающее устройство большой емкости. История винчестерских дисков очень интересна, но сначала — об их конструкции.

На больших скоростях вращения дисков самым страшным врагом для них является пыль. Кроме того, при больших скоростях головки, которые становятся все легче, плохо ведут себя в воздушных завихрениях. Вот и решили разработчики фирмы IBM и диск, и головку, и все механизмы заключить в прочный герметичный корпус и откачать из него воздух (вместе с пылью). Головка стала скользить не на воздушной подушке, а на слое специальной смазки, покрывающей диск. Естественно, диск из такого накопителя нельзя вынуть и заменить другим. Но за счет уменьшения всех размеров, увеличения точности изготовления, отсутствия таких мешающих факторов, как пыль, воздух и влажность, удалось получить компактное, надежное и удобное запоминающее устройство с большой (а в настоящее время и очень большой) информационной емкостью.

А название появилось случайно. Одно из подобных устройств было выпущено фирмой IBM в сдвоенном виде (два накопителя в одном корпусе) и имело обозначение 30/30 в соответствии с его информационной емкостью 30/30М байт. Такое же обозначение имеет и ружье «Винчестер», но в соответствии с калибрами его двух стволов. С чьей-то легкой руки это устройство назвали винчестером, а потом и все жесткие герметизированные диски, изготовленные по этой тех-

нологии, стали называть винчестерскими.

6.5. Магнитная лента

Последнее из распространенных запоминающих устройств — магнитофон. Запись на магнитной ленте надежна, именно на ленте легче всего перенести информацию с одной машины на другую.

Магнитофон ЭВМ очень похож на бытовые магнитофоны высокого класса, одно из отличий — датчик для измерения скорости движения ленты.

42

Второе, главное отличие — число дорожек. Данные записываются параллельно по байтам. Раньше байты были разного размера (от 5 до 8 бит) и магнитофоны имели от 6 до 9 дорожек. Со временем остановились на 9 дорожках. При этом данные записываются таким образом, что 9-й бит показывает, четное или нечетное число единиц в записываемом байте, и тем самым служит контрольным разрядом. Побочный эффект такого способа в том, что если 9-й бит равен 1 при четном числе единиц в байте, то в таком 9-разрядном коде всегда есть хоть одна единица (т. е. импульс на ленте) и скорость магнитной ленты можно подстраивать по самым записанным данным. Это упрощает конструкцию магнитофона и повышает надежность записи.

На ленте нет разметки (отдельные виды магнитофонов и форматов не в счет), данные записываются блоками от 18 до 2048 байт, и для каждого блока записывают по два проверочных кода. Первый из них называется байтом циклического контроля (это связано со способом вычисления контрольного числа). Второй — байт продольного контроля: подсчитываются все единицы на дорожке, и если их число нечетно, то в соответствующий разряд этого байта записывается единица, т. е. число единиц на дорожке, включая байт продольного контроля, должно быть четным.

Между блоками вставляется так называемый межзонный, или межблочный, промежуток (рис. 4).



Знакомый нам по программе DUMP ленточный маркер (иногда в документации называется кодом файла — КФ) представляет собой блок, состоящий из двух байтов со значениями 023 (восьмеричное), разделенных пустым промежутком, равным по времени 8 тактам записи. С каждой стороны ленточного маркера имеется по одному межблочному промежутку, который по размеру равен нашему блоку в 512 байт. Таким образом, каждый ленточный маркер «крадет» у файла примерно один блок.

Одно из основных неудобств работы с лентой — то, что ее нельзя разметить на блоки, как магнитный диск. Это вызвано следующими причинами.

Во-первых, мы можем определить длину блока только числом байтов, а не длиной записанной ленты, которая зависит от точности поддержания скорости движения и от изменений размеров самой ленты. На коротком кусочке это может быть незаметно, но при длине 375 м (как в нашем магнитофоне) можно просто промахнуться мимо нужного номера блока, затерев его заголовок предыдущей записью.

Во-вторых, межблочный промежуток также определен приблизительно. Время остановки и пуска магнитофона при записи отдельного блока (так называемые старт-стопные времена) зависит от его регулировки и свое у каждого экземпляра.

Из того, что сейчас было сказано, вытекает правило и даже психология работы пользователя с магнитной лентой: последняя по времени запись является последней на ленте.

Продолжение следует

Обучающая система-тренажер

Одна из главных задач, стоящих перед реформой школьного образования, — воспитание и развитие навыков творческого мышления учащихся. Однако на сегодня не известно ни одной сколько-нибудь универсальной методики обучения творческому мышлению. (Собственно, познание механизмов и разработка методологии обучения творческому мышлению является актуальной проблемой современной психологической и педагогической науки.) С другой стороны, хорошо известно, что взаимосвязь между творчеством и рутинной в процессе познания вообще и в процессе обучения в частности носит диалектический характер: рутинные элементы являются необходимым условием творческого процесса — и в любом творческом процессе существуют этапы, когда рутинная работа начинает проявлять значительный тормозящий эффект и от нее желательнее максимально освободиться. Таким образом, в ряде случаев (особенно в процессе обучения) возникает необходимость регулировать дозировку рутинного компонента. Важную роль в решении этой непростой проблемы может сыграть использование возможностей современной информационной технологии.

В статье А и Г. Зенкиных предложен оригинальный алгоритмический подход к конструированию стратегий решения широкого круга вычислительных задач по геометрии, основанный на процедуре построения так называемой разрешающей последовательности треугольников. Подход реализован в виде диалоговой системы ZGEOM, которая позволяет фактически разделить эвристический и рутинный этапы решения задачи, передать все вычислительные операции ЭВМ и тем самым сосредоточить все внимание и время пользователя на наиболее важном и обычно наиболее сложном эвристическом этапе построения логической схемы (стра-

тегии) решения задачи. При разработке системы был использован целый ряд перспективных концепций методологии искусственного интеллекта — одного из ведущих направлений современной информатики, а именно: реализация на уровне режимов работы одной интеллектуальной системы возможностей информационных, экспертных и обучающих систем, систем автоматизации научных исследований и т. п.; концепция автоматического синтеза вычислительного алгоритма в зависимости от типа исходных данных; использование компьютерной графики как средства активизации творческого воображения пользователя; полиоконная система вывода разнородной (тексты, формулы, рисунки, схемы, графики) информации; использование музыкальных возможностей персональной ЭВМ (звукового канала связи с проблемой) как средства фиксации внимания пользователя на специфических свойствах абстрактных математических объектов; развитая система диагностики ошибок пользователя и система методических подсказок в процессе поиска решения; ориентация на общение не столько с ЭВМ (как мы привыкли), сколько с самой исследуемой проблемой и т. п. Методической продуктивностью и большой наглядностью и удобством отличается предложенный метод построения дерева поиска стратегии решения задач.

Диалоговая система ZGEOM, без сомнения, обладает многими дидактическими достоинствами и вполне может быть рекомендована для использования в школьном курсе по геометрии.

*Д. ПОСПЕЛОВ,
зам. председателя Совета
по искусственному интеллекту
при Президиуме АН СССР,
доктор техн. наук, профессор*

Одним из наиболее сложных в школьном курсе геометрии является раздел «Многогранники». При решении задач из него нужно особенно широко использовать логическое рассуждение, эвристический поиск, построение и анализ альтернативных вариантов решений и т. п.

Процесс решения любой вычислительной задачи раздела «Многогранники»

можно разбить на два основных этапа: эвристический (построение логической схемы, стратегии, решения) и рутинный, или вычислительный (выполнение некоторой последовательности вычислительных процедур, соответствующей выбранной стратегии решения, для получения числового результата). Очевидно, что активная выработка творческих навыков

происходит именно на эвристическом этапе, поэтому в тех, например, случаях, когда ученик уже достаточно хорошо освоил технику тригонометрических вычислений, но испытывает трудности логического характера, было бы желательно существенно уменьшить (а в идеале вообще устранить) затраты труда и времени на выполнение рутинных операций и тем самым сконцентрировать внимание (и время) ученика на вопросах конструирования стратегии решений геометрических задач.

44 К сожалению, довольно часто (особенно при самостоятельной работе) единственным критерием проверки правильности выбранной стратегии решения задачи является сравнение полученного числового результата с ответом учебника, поэтому при традиционном подходе рутинный этап (нередко весьма трудоемкий) оказывается, как правило, неустраняемым.

Ниже приведено описание основных алгоритмов и некоторых возможностей реализованной на их основе диалоговой обучающей системы-тренажера ZGEOM, которая позволяет: фактически разделить эвристический и рутинный этапы решения геометрических задач указанного типа; передать все вычислительные операции компьютеру; существенно интенсифицировать процесс решения больших массивов типовых геометрических задач, акцентируя внимание именно на логике их решения.

Алгоритм разделения логики и рутинности

Опишем основную идею предлагаемого подхода, следуя в основном работе [1]. Решение многих задач стереометрии сводится к определению некоторых линейных и/или угловых элементов пространственных фигур и/или к вычислению некоторых заданных функций от этих элементов (периметра, площади, объема и т. п.). В достаточно общем случае задачу нахождения указанных элементов можно представить как процедуру решения некоторой последовательности треугольников

$$\Delta_1, \Delta_2, \dots, \Delta_p, \dots, \Delta_n \quad (n \geq 1), \quad (1)$$

обладающей следующими свойствами:

исходной информации задачи («Дано...») достаточно для того, чтобы решить треугольник Δ_1 , т. е. рассчитать все его стороны и углы;

можно решить любой треугольник Δ_i в последовательности (1), если уже решены все предшествующие треугольники $\Delta_1, \Delta_2, \dots, \Delta_{i-1}$;

искомые элементы соответствующей многогранной фигуры являются некоторыми элементами последнего треугольника Δ_n последовательности (1).

В целях стандартизации задания информации об элементах любого треугольника введем следующие соглашения: а) значения всех элементов треугольника (трех его сторон и трех углов) будем записывать в виде упорядоченной шестерки чисел

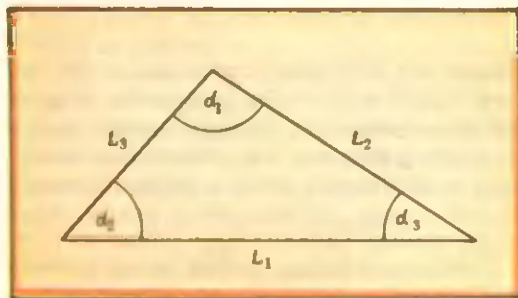
$$L_1 \ L_2 \ L_3 \ \alpha_1 \ \alpha_2 \ \alpha_3 \quad (2)$$

где первые три позиции занимают значения сторон (пронумерованных произвольным образом), а вторые три позиции занимают значения противолежащих соответствующим сторонам углов, так что угол α_i является противолежащим стороне L_i , $i=1, 2, 3$ (рис 1); б) если на данном этапе решения задачи какой-либо элемент треугольника неизвестен, т. е. его значение еще предстоит рассчитать, то в соответствующей ему позиции запишем ноль.

В таком случае весь процесс решения последовательности (1) разбивается на n шагов, каждый из которых включает в себя два основных этапа:

основная операция — собственно процедура решения очередного треугольника Δ_i ;

вспомогательная операция (одна или несколько) — пересылка значений некоторых из рассчитанных элементов тре-



угольника Δ_i в соответствующие позиции шестерок (2) некоторых из последующих треугольников $\Delta_{i+1}, \dots, \Delta_n$.

Оба этапа легко автоматизируются: первый состоит в применении соответствующей модификации одного из стандартных алгоритмов решения треугольников, причем выбор конкретной модификации вычислительного алгоритма однозначно определяется структурой записи информации в шестерке (2), а второй однозначно определяется ранее выбранной логической схемой решения задачи, зафиксированной в соответствующей последовательности треугольников (1).

Полное (включая выдачу числового ответа на экран дисплея) решение задачи с помощью системы ZGEOM состоит из следующих основных этапов:

I. Анализ исходной информации («Дано...»).

II. Уяснение конечной цели («Найти...»).

III. Построение последовательности треугольников (1) от «Найти...» к «Дано...».

IV. Составление массива DATA, однозначно определяемого выбранной последовательностью треугольников (1). Этот массив состоит из шестерок (2), соответствующих последовательности (1), и специальных чисел, диктующих ЭВМ способ обработки этих данных (подробно массив будет описан ниже).

V. Ввод массива DATA в компьютер.

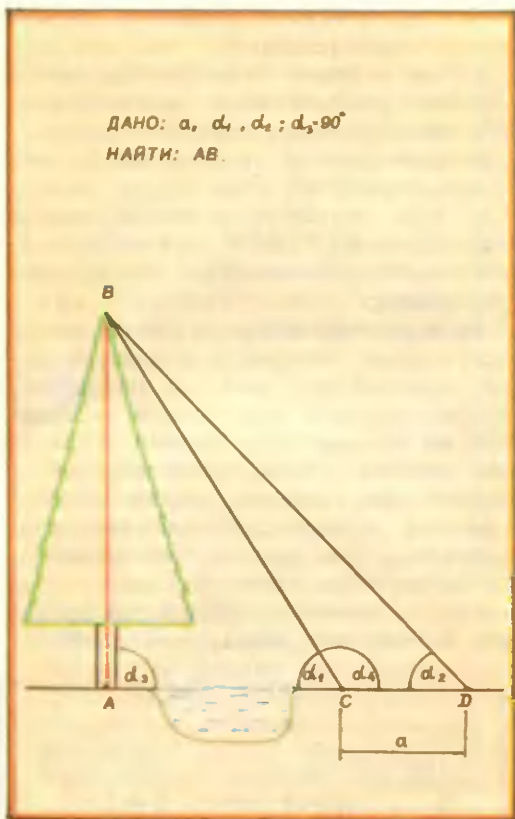
VI. Выполнение компьютером вычислительных операций.

Этапы I—III — эвристические, этапы IV—VI — чисто формальные. Минимизируя затраты времени на выполнение формальных этапов, мы и получаем возможность сосредоточить основное внимание на эвристических этапах решения задачи.

Эвристический этап решения задачи

Особенности реализации этого этапа в рамках системы ZGEOM продемонстрируем на конкретном примере.

Задача 1. Найти высоту дерева, растущего на одном берегу реки, если на другом ее берегу заданы две точки C и D и известны расстояние $CD=a$ между эти-



ми точками и углы $\angle ABC = \alpha_1$ и $\angle ADB = \alpha_2$ (рис. 2).

Предварительный анализ условия задачи («Дано...») позволяет установить неявно («дерево, растущее...») содержащийся в нем математический факт: $\alpha_3 = 90^\circ$. С его учетом записываем:

Дано: $a, \alpha_1, \alpha_2; \alpha_3 = 90^\circ$.

Найти АВ.

Далее строится так называемое дерево поиска стратегий решения задачи (рис. 3) согласно следующей логике рассуждений («от цели»):

1. Какой элемент (длину, угол) требуется найти?

2. Если этот элемент невозможно определить из более простых соображений, то найти все треугольники (в общем случае — фигуры), в которые входит искомым элемент (или элементы).

3. Среди этих треугольников (фигур) выбрать те, в которых имеются элементы из «Дано...», причем это «Дано...» выступает здесь в роли активной под-

сказки при выборе правильного направления поиска решения.

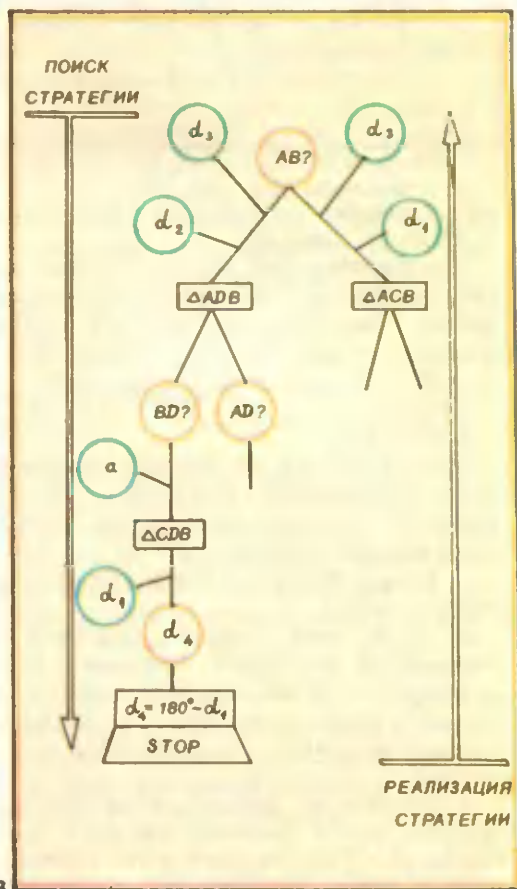
4. Для каждого из выбранных таким образом треугольников проанализировать «проблему разрешимости»:

а) если данный треугольник можно решить, то СТОП;

б) если данный треугольник нельзя решить, то ПЕРЕЙТИ к 1 и продолжить поиск решения уже для данного треугольника.

Очевидно, что это лишь общая схема рассуждений, которая дополняется целой системой методических подсказок, содержащихся в базе знаний системы ZGEOM (аналог базы знаний экспертных систем). Приведенная на рис. 3 форма представления дерева поиска стратегий решения позволяет наглядно отобразить сам процесс рассуждений, его логику. При этом появляется возможность довольно эффективно проверить и саму эту логику, что особенно

46



3

важно для самоконтроля учащегося: после построения одной из ветвей дерева (стратегии решения) двигаемся из ее верхней вершины $AB?$ вниз и фиксируем содержимое зеленых «листочков», т. е. использованных при построении данной ветви элементов из «Дано...» в виде последовательности:

$$a_3, a_2, a, a_1. \quad (3)$$

Очевидно, что в подавляющем большинстве случаев множество элементов последовательности (3) должно совпадать с множеством элементов в «Дано...». Если это не так, то ход рассуждений, скорее всего, неправилен.

При желании можно построить полное дерево стратегий и проанализировать все альтернативные варианты решения данной задачи.

Из рис. 3 видно, что если поиск (фактически конструирование) стратегий решения задачи осуществляется при построении дерева сверху вниз, то реализация этих стратегий осуществляется при движении снизу вверх: выписывая в ряд встречающиеся при этом треугольники, мы и получаем искомую последовательность (1). В нашей задаче такая последовательность состоит всего из двух треугольников:

$$\triangle CDB, \triangle ADB. \quad (1a)$$

Теперь нам остается записать массив DATA для последовательности (1a), ввести его в ЭВМ и решить задачу. Однако прежде, чем перейти к описанию структуры массива DATA, сделаем следующее добавление.

Только что описанный этап построения дерева поиска стратегий решения задачи является по своей сути эвристическим и не поддается полной формализации. Но, как уже отмечалось выше, и здесь можно эффективно использовать компьютер. Для этого в меню системы ZGEOM имеется режим «подсказки» (HELP), который позволяет пользователю оперативно получить информацию из различных банков данных (БД) системы ZGEOM. Например, из БД ТРИГОНОМЕТРИЯ пользователь может получить информацию о методах

решения треугольников, об основных тригонометрических соотношениях, о свойствах тригонометрических функций, причем сама динамика поведения этих функций при изменении, например, таких параметров, как амплитуда, фаза или частота, отображается на экране дисплея многоцветными графиками, а особые точки этих графиков (максимумы, минимумы, точки перегиба, нули) в процессе их построения на экране отмечаются различными музыкальными звуками. Уместно подчеркнуть, что одновременное использование графики, цвета и звука является новым и важным аспектом компьютерной технологии обучения, способствующим повышению активности учащихся, развитию их творческого воображения. Поэтому отличительной особенностью системы ZGEOM является изначальная ориентация на максимальное использование возможностей интерактивной компьютерной графики, особенно при изучении сложных математических понятий, объектов и их свойств [2]. Далее из БД ГЕОМЕТРИЯ можно оперативно получить информацию об определениях и свойствах основных геометрических фигур, об основных геометрических соотношениях и расчетных формулах и т. д. Наконец, из БД ПОИСК можно получить информацию о самой стратегии поиска решений, об основных положениях системного анализа, а также рекомендации о тех или иных методических приемах и т. п.

Описание структуры массива исходных данных DATA

Для организации массива DATA использован метод специальных чисел [1]. Суть его состоит в том, что фиксируется некий конечный набор достаточно больших целых чисел (в системе ZGEOM все такие числа больше 5000), которые можно вставлять в любом месте массива DATA и которые информируют ЭВМ о необходимости выполнения вполне определенных сложных операций. Обычно за каждым спецчислом следует фиксированная порция информации, т. е. некоторая конечная последовательность чисел, каждое из

которых выполняет заранее оговоренную и раз навсегда фиксированную роль, так что спецчисло является как бы указателем, именем некоторой сложной операции, которую должна выполнить ЭВМ, а связанная с ним последовательность цифр является векторным операндом этой операции.

Например, если в массиве DATA встречается спецчисло 5555, то ЭВМ интерпретирует семерку следующих за ним чисел как последовательность

$$i, L_1, L_2, L_3, a_1, a_2, a_3,$$

где i — номер треугольника в последовательности (1), а остальные шесть чисел — значения элементов этого треугольника, записанные в форме (2), и вычисляет неизвестные (равные нулю) значения элементов шестерки. Если в массиве DATA встречается спецчисло 5000, то машина выведет на экран дисплея численные значения элементов треугольника, предшествующего этому спецчислу, т. е. последнего из уже решенных. Таким образом, спецчисло 5555 можно назвать признаком треугольника, а 5000 — признаком печати.

Приведем значения еще нескольких спецчисел, которые понадобятся нам в дальнейшем.

Указатель пересылок

$$5013, n, a_1, b_1, c_1, a_2, b_2, c_2, \dots, a_n, b_n, c_n,$$

где n — количество пересылок, a_i — номер (от 1 до 6) элемента шестерки последнего (только что решенного) треугольника, b_i — номер одного из последующих треугольников, c_i — номер (от 1 до 6) элемента этого треугольника. По спецчислу 5013 осуществляется n пересылок a_i -го элемента текущего треугольника на место c_i -го элемента b_i -го треугольника в массиве DATA.

Указатель сложения

$$5001, a, b, n, c_1, d_1, c_2, d_2, \dots, c_n, d_n,$$

задающий сложение a -го и b -го элементов текущего треугольника и n -кратной засылки полученного результата на места d_i -х элементов c_i -х треугольников в массиве DATA ($i=1, 2, \dots, n$).

Указатель конца массива DATA 6666.

Вернемся теперь к нашей задаче и решающей ее последовательности треугольников (1а). Информация о треугольниках последовательности (1а) может быть записана в виде табл. 1 (вторые строки — примечания, обозначения соответствующих элементов треугольников по рис. 2).

Таблица 1

△СДВ: 5555, 1,	a_1	0	0	0	α_1	α_2	
№ СД	ВД	ВС	∠ДВС	∠ВСД	∠СДВ		
△АДВ: 5555, 2,	0	0	0	α_3	α_2	0	
№ ВД	АВ	АД	∠ВАД	∠АДВ	∠ДВА		

Очевидно, что △СДВ рассчитывается по стороне СД= a и двум прилежащим к ней углам ∠ВСД= α_1 и ∠СДВ= α_2 .

48 После выполнения данной операции, определяемой первым спецчислом 5555, вместо нулей в массиве DATA появятся рассчитанные значения сторон ВД и ВС и угла ∠ДВС. Для того чтобы после этого можно было решить △АДВ, нужно организовать одну пересылку (указана стрелкой) рассчитанного значения элемента № 2 (стороны ВД) треугольника № 1 (СДВ) в треугольник № 2 (АДВ) на место его элемента № 1, т. е. после описания треугольника № 1 нужно вставить спецчисло

5013, 1, 2, 2, 1.

В таком случае △АДВ решается по стороне ВД и двум известным углам: ∠ВАД= α_3 и ∠АДВ= α_2 . После расчета △АДВ ставим спецчисло 5000 (выдача

на экран значений всех его элементов, в том числе и искомой высоты АВ) и спецчисло 6666 — признак конца массива DATA.

Для удобства построения массива DATA и унификации формы записи информации используется типовой бланк (табл. 2). Заметим, что в нем строки можно пропускать или использовать для внесения любых примечаний пользователя.

Теперь для получения полного решения задачи фактически достаточно ввести в ЭВМ следующую последовательность чисел:

5555, 1, a , 0, 0, 0, α_1 , α_2 , 5013, 1, 2, 2, 1, 5000, 5555, 2, 0, 0, 0, α_3 , α_2 , 0, 5000, 6666.

После нажатия клавиши RETURN на экране дисплея практически мгновенно появляются две выдачи: шестерка рассчитанных значений элементов △СДВ (по первому спецчислу 5000) и шестерка рассчитанных значений элементов △АДВ (по второму спецчислу 5000), после чего машина остановится (по спецчислу 6666).

Во избежание отождествления со спецчислами все элементы треугольников в массиве DATA должны быть меньше 5000. Как правило, для школьных задач это условие выполняется. В противном случае проводится тривиальное масштабирование линейных элементов соответствующих треугольников.

Добавим, что в системе ZGEOM предусмотрены два режима решения задач — интерактивный и пакетный (в последнем имеется чисто программное огра-

Таблица 2

Типовой бланк для записи массива DATA

Признак треугольника	№ треугольника	Информация об элементах треугольника						Вспомогательные операции	
		стороны			углы			спец-число	
5555 △СДВ	1	a СД	0 ВД	0 ВС	0 ∠ДВС	α_1 ∠ВСД	α_2 ∠СДВ	5013	1, 2, 2, 1
5555 △АДВ	2	0 ВД	0 АВ	0 АД	α_3 ∠ВАД	α_2 ∠АДВ	0 ∠ДВА	5000 5000 6666	

нение: количество одновременно решаемых задач не должно превышать 500).

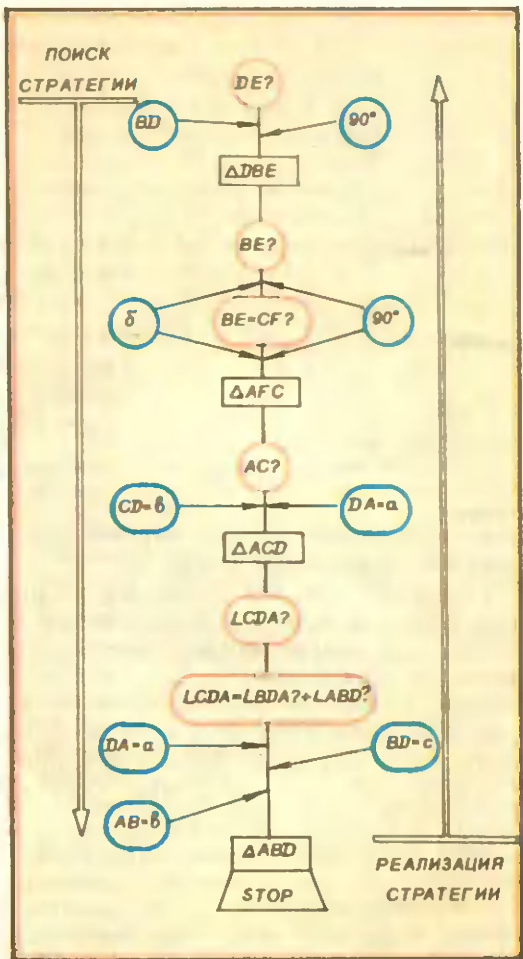
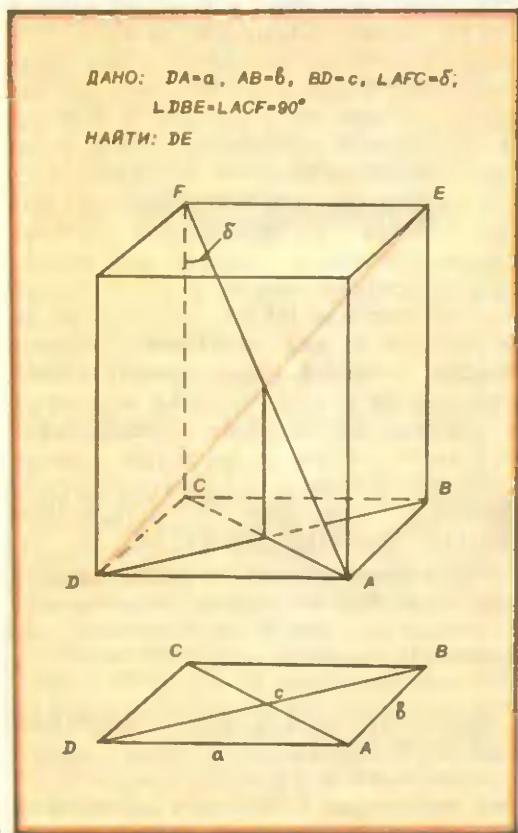
По желанию пользователя рассчитанный треугольник может быть построен на экране дисплея и в виде цветной картинка.

В системе ZGEOM имеется блок диагностики, который автоматически анализирует синтаксическую правильность ввода массива DATA и хода решения задачи с выдачей соответствующей диагностической информации на экран дисплея и рекомендаций пользователю о том, какие учебные файлы системы ему полезно посмотреть.

Стереометрические задачи

Рассмотрим конкретный пример.

Задача 2. В основании прямого параллелепипеда (рис. 4) лежит параллелограмм со сторонами a, b и диагональю c . Найти диагональ DE этого параллелепипеда, если известен угол δ ,



На дереве поиска зеленым цветом выделено то, что дано, красным цветом — то, что нужно найти (включая промежуточные неизвестные). Вопросительными знаками отмечены те элементы, которые должна вычислить ЭВМ.

образованный другой его диагональю AF с ребром CF .

Логика ее решения приведена на рис. 5 в виде дерева поиска стратегий решения. Поднимаясь по нему снизу вверх, получаем искомую последовательность треугольников

$\Delta ABD, \Delta ACD, \Delta AFC, \Delta DBE,$ (16)

решающих нашу задачу.

Основываясь на последовательности (16), используя чертеж (рис. 5) и дерево поиска (рис. 4), заполняем типовой бланк решения задачи, внося в него

Типовой бланк для записи массива DATA

Признак треугольника	№ треугольника	Информация об элементах треугольника						Вспомогательные операции	
		стороны			углы			спец-число	операнды
5555 △ABD	1	a	b	c	0	0	0	5001	4, 5, 1, 2, 6, ∠ABD + ∠BDA
5555 △ACD	2	DA	AB	BD	∠ABD	∠BDA	∠DAB	5000	
5555 △ACD	2	a	b	0	0	0	δ	5013	1, 3, 3, 1, 1, 2, 4, 2
5555 △AFC	3	DA	DC	AC	∠ACD	∠DAC	∠CDA	5000	
5555 △AFC	3	0	0	0	δ	0	90°	5013	
5555 △DVE	4	AC	CF	AF	∠AFC	∠CAF	∠ACF	5000	
5555 △DVE	4	c	0	0	0	0	90°	5000	
		DV	VE	DE	∠VED	∠EDV	∠DVE	6666	

50

для наглядности и самоконтроля необходимые примечания (табл. 3).

Таким образом, для получения числового решения задачи 2 нужно ввести в ЭВМ следующую последовательность чисел (массив DATA):

5555, 1, a, b, c, 0, 0, 0, 5001, 4, 5, 1, 2, 6, 5000, 5555, 2, a, b, 0, 0, 0, 0, 5013, 1, 3, 3, 1, 5000, 5555, 3, 0, 0, 0, 0, δ, 0, 90, 5013, 1, 2, 4, 2, 5000, 5555, 4, c, 0, 0, 0, 0, 90, 5000, 6666.

Ввод этих чисел занимает 1—2 мин, решение — 7—10 с, так что рутинный, чисто вычислительный этап действительно перестает быть лимитирующей по времени стадией решения задачи (без ЭВМ эта стадия занимает 20—30 мин).

Использование спецчисел (их в системе ZGEOM около 20) при построении массива DATA не вызывает особых трудностей, поскольку все их полные описания и инструкции по применению хранятся в одном из файлов системы и по обращению HELP выдаются на экран дисплея. Нет необходимости специально запоминать эти спецчисла, их назначение и возможности — все это естественным образом усваивается по мере накопления опыта решения задач с помощью системы ZGEOM.

Некоторые другие возможности и особенности системы

В настоящее время в банке данных систем ZGEOM содержится свыше

100 типовых задач по различным разделам школьного курса геометрии. Банк данных имеет структуру системы взаимосвязанных файлов: в текстовом файле хранятся описания условий задач в форме, близкой к учебнику; в файле стратегий решения хранятся соответствующие массивы исходных данных DATA (при стандартном режиме обучения этот файл не доступен для пользователя); в файле графических данных хранятся чертежи и рисунки к задачам. В стандартном обучающем режиме работы с системой ZGEOM на экране дисплея резервируются текстовые окна для организации интерактивного режима взаимодействия с задачей, для выдачи текста условия задачи и (если требуется) массива DATA, а также графические окна для чертежей, деревьев поиска решений, для представления результатов и т. п. Чертежи выводятся в статическом режиме, однако, если позволяет объем оперативной памяти (64К байт и более), то к системе ZGEOM возможно подключение подсистемы ROUTE системы ДИАХИМ [1, 3], которая обеспечивает построение любых пространственных каркасных объектов и работу с ними в динамических режимах (вращения, деформации произвольного типа и др.).

Эффективным, а нередко и единственным, методом решения многих задач геометрии является метод дополнительных построений. Обычно его применение связано с кажущимся усложнением за-

дачи: увеличением количества альтернативных вариантов, формальным удлинением схемы рассуждений, возрастанием объема вычислений и т. п., что порождает определенный психологический барьер на пути его освоения учащимися. В рамках системы ZGEOM реализация альтернативных вариантов решения задачи сводится лишь к частичным изменениям соответствующего массива DATA. Четкое, многократно практически подтвержденное осознание того факта, что увеличение количества вариантов задачи и усложнение логики рассуждений не приводит к возрастанию трудозатрат на вычисления, будет, по-видимому, способствовать снижению указанного психологического барьера и постепенному превращению процедуры решения задач в процесс сравнительного исследования альтернативных вариантов их решения.

Совершенно очевидно, что эффективность системы ZGEOM будет существенно зависеть от научного обоснования конкретной методики ее применения в учебном процессе. Создание такой методики потребует привлечения широкого круга специалистов по школьному образованию.

В настоящее время система ZGEOM представляет собой программный комплекс, реализованный на языке BBC BASIC для персональной ЭВМ «Torch»;

минимальный рабочий вариант системы занимает около 15К байт оперативной памяти. Перевод системы на другие алгоритмические языки и ЭВМ не связан с принципиальными трудностями. Желющие получить дополнительную информацию могут обратиться по адресу: 119899, ГСП, Москва, В-234, Ленинские горы, МГУ, химический факультет, Зенкину А. А.

Литература

1. Зенкин А. А. Некоторые вопросы автоматизации химических научных исследований: Автореф. канд. дис. М.: изд-во МГУ, 1974.

2. Зенкин А. А. Применение компьютерной графики в теоретических исследованиях // Известия АН СССР. Сер. Техническая кибернетика, 1987, № 5.

3. Зенкин А. А., Устынюк Ю. А., Федосеева Н. Н. Стандартизация представления информации о геометрии молекулярных систем // Доклады АН СССР, 218, № 6. 1362—1364 (1974).

От редакции. Готовя к публикации эту статью, мы надеялись, что она заинтересует разработчиков педагогических программных средств — как профессиональных, так и самодеятельных. Многие особенности системы ZGEOM могут оказаться эффективными в других обучающих программах.

Видимо, система ZGEOM заинтересует и учителей, однако им придется подождать. Авторы не имеют возможности высылать тексты программ и тем более адаптировать систему на школьные ЭВМ. Будем надеяться, что ИИВТ АПН СССР займется этим, не откладывая дела в долгий ящик.

Л. ШТЕРНБЕРГ,
канд. физ.-мат. наук

Умеет ли считать ваш микрокалькулятор?

Эта статья адресована учителям информатики, использующим на своих уроках любую вычислительную технику, в частности микрокалькуляторы (программируемые или даже инженерные). В процессе работы они могут столкнуться с некоторыми проблемами, способными поставить неопытного человека в тупик. Например, проверяя с помощью калькулятора математические тождества, можно без труда «опровергнуть» любое, даже ассоциативный закон сложения или умножения (см., например: Дроздов В. О проверке математических формул с помощью калькулятора // Информатика и образование. 1987.

№ 6). Выполняя расчеты, учитель и ученики могут столкнуться с необъяснимыми, на первый взгляд, ошибками в результатах. Разобраться в этом им поможет предлагаемая статья.

Примеры в статье приведены для программируемых калькуляторов типа БЗ-34, но все описанные эффекты можно получить и на ЭВМ серий ЕС, ДВК, IBM PC или IBM PS/2 и всех остальных, взяв данные с учетом длины машинного слова конкретной ЭВМ.

Рассмотрение вычисления площади криволинейной трапеции связано с тем, что при его изучении в пробном учебном пособии

затрагиваются вопросы точности, однако в пособии для учителей в соответствующем месте допущена ошибка: двоичная система счисления, в которой работает большинство ЭВМ, к погрешности отношения не имеет.

Всем известно еще со школы, что $a \cdot b \cdot c = (a \cdot b) \cdot c = a \cdot (b \cdot c)$ — это так называемый сочетательный закон умножения (ассоциативность). Пусть у нас $a=10^{40}$, $b=10^{60}$, $c=10^{-20}$. И без калькулятора ясно, что $a \cdot b \cdot c = 10^{80}$, но при счете по формуле $(a \cdot b) \cdot c$ получим вместо результата ЕГГОГ (переполнение), а если сначала умножить b на c , то результат будет верный. Но здесь сигналом ЕГГОГ калькулятор хоть предупредил нас о том, что он не может посчитать правильно. А теперь пусть $a=10^{-40}$, $b=10^{-80}$, $c=10^{90}$, тогда при умножении $a \cdot (b \cdot c)$ получим правильный результат, а при умножении $(a \cdot b) \cdot c$ получим ноль (!): он возникает при перемножении a и b . Итак, с ассоциативностью наш калькулятор знаком плохо.

Известно, что $(a-b) \cdot c = a \cdot c - b \cdot c$ — это распределительный закон (дистрибутивность). Знаком ли калькулятор с ним? Пусть $a=1000030$, $b=1000000$, $c=2000030$, тогда $(a-b) \cdot c = 6000900$, а $a \cdot c - b \cdot c = 6\ 000\ 000$ — отличие на 900 единиц! Итак, с дистрибутивностью у калькулятора дела тоже обстоят неважно.

— Ну, это специально подобранные примеры, — может сказать читатель. — Я сколько считал на калькуляторах и получал хорошие, правильные результаты.

— А вы уверены, — зададим ему вопрос, — что в процессе вычисления по программе калькулятор не получил ноль вместо малого числа, а затем что-то множил на него, и вы приняли ошибочный результат за верный? Да, это специально подобранные числа, но как их подобрать? И не «подберутся» ли они случайно в вашей программе? А если подберутся, то сумеете ли вы их отличить?

Рассмотрим пример: дано несколько кубиков с разными длинами ребра, по заданным длинам определить суммарный объем кубиков. Программа для калькулятора выглядит просто:

00.Сх 01↑ 02.С/П 03↓ 04.Г х² 05.х 06.+
07.БП 08.02

При каждой остановке программы мы видим на индикаторе накопленную сумму, затем вводим длину ребра очередного куба и нажимаем С/П. Посчитаем объем 10 кубиков с длинами ребер 1001, 203.298, 17.25, 7, 43.095, 12.03, 3.41, 5.2, 21.5, 96.0. Получим 1 012 387 100. А теперь посчитаем ту же программу с теми же данными, но вводить их будем в обратном порядке: сначала 96.0, затем 21.5, 5.2 и т. д. Получим 1 012 387 400. Разница довольно существенна: 3 единицы младшего разряда калькулятора. Можно попробовать и другой порядок суммирования: 2-е число, 4-е, 6-е, 8-е, 10-е, 1-е, 3-е и т. д. — получим промежуточный результат. А если добавить еще 3 числа, например еще раз 7, 43.095, 12.03, то разница достигнет 4 единиц. А с другими числами все может оказаться в порядке.

Попробуем теперь реализовать на калькуляторе простейший численный метод: вычисление интеграла по методу средних прямоугольников. И функцию возьмем простейшую, для которой результат должен получиться точным — $y=x$. Алгоритм вычисления интеграла от a до b с разбиением отрезка на M шагов имеет вид:

ввод a, b, M ;

$h := (b-a)/M$; инт:=0; $x := a \div h/2$;

повторять M раз инт:=инт+ $f(x)$;

$x := x+h$

вывод инт $\cdot h$;

Его реализация (в несколько оптимизированном виде) на калькуляторе.

Распределение регистров: a — P0, b — не хранится, M — P2, h — P3, x — P4, инт — P5.

Программа:

00.П 0 01.С/П 02.→03. — 04.С/П 05.П 2
06.÷07.П 3 08.2 09.÷10.ИП 0 11.÷12.
П 4 13.Сх 14.П 5 15.ИП 4 16.К НОП
17.ИП 5 18.÷ 19.П 5 20.ИП 4 21.К НОП
22.ИП 3 23.÷ 24.П 4 25.F L2 26.15
27.ИП 5 28.ИП 3 29.х 30.С/П

Назначение команд К НОП будет объяснено позже. Инструкция по использованию: набираем значение a , нажимаем С/П, после останова набираем b , С/П, M , С/П, после останова читаем результат.

Приведем результаты. Пусть $a=100$, $b=103$, $M=9$. После 29 с счета получаем 304.49997 (отличие от точного результата 304.5 на 0.00003). Из теории известно, что точность возрастает с увеличением числа разбиений. Увеличим M вдвое, при тех же a и b и $M=18$ после 58 с счета получаем 304.50007 (!?) — погрешность увеличилась (!) более чем в 2 раза. При $M=27$ получим 304.49998 — очень хорошо. А при $M=180$ калькулятор считает около 10 мин и ... результат 304.50095 — погрешность забралась чуть не в 3-й от конца разряд.

При $a=100000$, $b=100003$ получим: при $M=9$ — 300004.47, при $M=18$ — 300004.58, при $M=180$ — 300005.43. Заглянув после последнего расчета в P_4 , где хранится x , мы увидим там весьма странное значение 100003.61, которое явно больше, чем нужно.

Калькулятор не стоит нести в мастерскую. Он вполне исправен. Просто мы столкнулись с проблемой, знакомой любому программисту, независимо от того, работает он на простейшем калькуляторе или на мощнейшей ЭВМ. Эта проблема — особенности машинной арифметики.

Любая ЭВМ работает в ограниченном диапазоне и с ограниченной точностью представления чисел. Для калькулятора диапазон от 10^{-99} до 10^{99} (меньшие по модулю числа считаются нулем, а большие дают ЕГГОГ), а точность — 8 знаков (т. е. относительная погрешность не более $0.5 \cdot 10^{-7}$, а абсолютная зависит от величины числа). У ЕС ЭВМ диапазон от 10^{-75} до 10^{75} , а точность — всего 7 знаков, у БЭСМ-6 диапазон всего от 10^{-19} до 10^{19} , но точность большая —

12 знаков. Однако на ЭВМ выполняются серьезнейшие расчеты, значит — считать на них можно, и вопрос заголовка статьи следует сформулировать иначе: умеете ли вы считать на вашем микрокалькуляторе?

Сначала разберемся, что же происходило при вычислении интеграла. Вставим по адресу 21 команду С/П (вместо К НОП) — она нам будет при остановке показывать значение x , с которым идет счет на текущем шаге. Для более контрастного показа возьмем $a=1\ 000\ 000$, $b=1\ 000\ 003$. Последовательные значения x показаны в таблице (правую колонку пока не рассматривайте):

В отличие от обычной арифметики, где погрешность суммы равна сумме погрешностей слагаемых, в машинной арифметике меньшее по порядку число сначала приобретает погрешность того же порядка, что и большее по порядку число, а затем погрешности складываются. Как видим из таблицы, за 10 шагов накопилась погрешность, равная шагу. Именно это и происходило в наших расчетах, но так как a , b и h были более близки друг к другу по своему порядку, то для накопления погрешности понадобилось гораздо больше шагов.

Итак, нам надо научиться бороться с накоплением погрешностей. Не вдаваясь в глубины теории, сформулируем несколько достаточно простых правил.

Правило 1. Не жалейте лишней цифры. Вводите данные с максимально возможной точностью. 3.14 — это не л, 0.333 — это не $1/3$ и т. д. Объем шара радиуса 20 равен 33510.318, но если вместо л взять 3.14, то получим 33493.331. А сумеете ли вы на глаз отличить, что последнее зна-

Точное значение	Вычисление $x = x + h$		Вычисление $x := x_0 + ih$	
	результат	погрешность	результат	погрешность
1000000	1000000	0	1000000	0
1000000.333...	1000000.3	-0.0333...	1000000.3	-0.0333...
1000000.666...	1000000.6	-0.0666...	1000000.7	0.0333...
1000001	1000000.9	-0.1	1000001	0
1000001.333...	1000001.2	-0.1333...	1000001.3	-0.0333...
1000001.666...	1000001.5	-0.1666...	1000001.7	0.0333...
1000002	1000001.8	-0.2	1000002	0
1000002.333...	1000002.1	-0.2333...	1000002.3	-0.0333...
1000002.666...	1000002.4	-0.2666...	1000002.7	0.0333...
1000003	1000002.7	-0.3	1000003	0
1000003.333...	1000003	-0.3333...		

чение достаточно далеко от истины? Конечно, при наличии специальной клавиши для π никто не станет задавать его в виде 3.14, но в физике и в математике есть много констант, и надо помнить, что $\pi = 3.1415926$, а не 3.14, число $e = 2.7182818$, а не 2.72, что ускорение свободного падения g — это не 9.81 и т. д. Заметим, что цикл

$x := 0$; пока $x \leq 1$ цикл ... $x := x + 1/3$ кц
будет выполняться 3 раза, если одну треть представить в виде $3.3333333 \cdot 10^{-1}$, и 4 раза, если задать как 0.3333333 .

Правило 2. Держитесь середины диапазона. Организуйте вычисления так, чтобы промежуточные результаты имели минимальный по абсолютной величине порядок. Например, при перемножении нескольких величин надо чередовать большие и малые значения (вспомним пример с умножением $a \cdot b \cdot c$). При сложении это правило чуть-чуть меняется: сначала складывайте числа с наименьшими порядками, затем добавляйте слагаемые со все большими и большими порядками. Например, $1.0000000 + 0.0000003 \cdot 10^{-1} + 0.0000003 \cdot 10^{-1} = 1.0000000, 0.0000003 \cdot 10^{-1} + 0.0000003 \cdot 10^{-1} + 1.0000000 = 1.0000001$.

Правило 3. Применяйте масштабирование. Если промежуточные результаты близко подходят к границам диапазона или выходят за них, попробуйте использовать масштабирование. Если, например, из физического смысла задачи ясно, что при увеличении (уменьшении) исходных данных в 10 раз результат увеличится (уменьшится) в 100 раз, то при вводе данных поставьте запятые на несколько разрядов правее или левее, только не забудьте затем сдвинуть их в нужную сторону в результате. Например, пусть надо вычислить $\sqrt[6]{x^3 + y^3}$, x и y имеют порядок 40. И хотя результат будет иметь порядок 20, но при возведении в куб получим переполнение. Достаточно уменьшить их порядок на 10 единиц (т. е. перейти от, скажем, миллиграммов к десяткам тонн), и считать будет можно, а результат следует умножить на 10^5 или просто помнить, что он получается в соответствующих единицах. Иногда разные данные надо вводить с разным масштабом; например, при счете по формуле

$\sqrt[3]{x^3 + y^3 + C\sqrt{a^2 + b^2}}$, где $x, y \approx 10^{40}$, $a, b \approx 10^{-50}$, $C \approx 10^{90}$, можно взять для x, y масштаб 10^{-10} , т. е. вводить значения, меньшие истинных в 10^{10} раз, для a, b — масштаб 10^{10} , тогда вместо C надо взять значение $C \cdot 10^{-20}$ и помнить, что получившийся результат в 10^{10} раз меньше истинного.

Правило 4. Платите скоростью за точность. Для получения более точных результатов следует применять специальные алгоритмы, которые устойчивы к погрешностям, т. е. работают так, что погрешности не накапливаются или взаимно компенсируются. Такие алгоритмы не обязательно сложны, но обычно требуют больше времени на выполнение. Например, для задачи интегрирования вместо наращивания x на h на каждом шаге можно вычислять его по формуле $x := (a - h/2) + ih$, где $(a - h/2)$ вычисляется один раз перед циклом, а i (номер шага) можно взять из P2 (правда, тогда i будет меняться не от 1 до M , а от M до 1, но это не важно). Погрешности в этом варианте накопиться негде (см. правый столбец в таблице): на каждом шаге цикла x получается из исходных данных всего двумя операциями; а при последовательном добавлении шага последние значения x получаются путем многих сложений (на каждом шаге по одному), при которых и накапливается ошибка. Именно из таких соображений и надо выбирать алгоритм.

Правило 5. Сдвигайте аргумент. Из сказанного выше ясно, что, например, вычисление интеграла на отрезке [100, 103] с разбиением его на M шагов будет заведомо точнее, чем на отрезке (10000, 10003] с разбиением на столько же шагов. Но иногда с помощью некоторых преобразований можно свести один отрезок к другому. Простейший пример: если функция периодическая, то сведение тривиально. Если в нашей программе интегрирования заменить K НОП по адресу 16 на команду $\Gamma \sin$, то будет считаться интеграл от синуса. При $M = 20$ на отрезке [0, π] получим 2.0020581 (точное значение 2), а на отрезке [1000 π , 10001 π] получим 2.00333 — погрешность в 1.5 раза больше, хотя, казалось бы, не все ли равно, площадь какой дуги синусоиды вычислять.

Правило 6. Оценивайте погрешность. В любом случае надо хотя бы грубо прикинуть, какой погрешности можно ожидать и устраивает ли это автора расчета. Задавая вместо π число 3.14, мы «запускаем» в вычисления погрешность около 0.05 %, и трудно сказать, во что она превратится в конце. Но в примере

с расчетом радиуса шара можно точно сказать, что относительная погрешность результата будет тоже 0.05 %, а абсолютная — около 17. Но, может быть, получив результат 33510.318, мы все равно запишем «около 335 центнеров», т. е. нас такая погрешность устраивает и можно работать простейшими методами.

Ю. ПОДЪЯБЛОНСКИЙ С. КАЧАЕВ

Класс БК без ДВК

В статье «Тернистый путь БК в наш дом» (Радио. 1987. № 6) отмечалось, что в 1987 г. будет выпущено 40000 ПЭВМ «Электроника БК-0010» (БК) и основную их массу направят в учебные заведения. Уже сейчас многие КВТ оборудованы БК, однако не везде БК объединены в локальную сеть, что значительно снижает эффективность их использования.

Одна из школ Рязани обратилась с просьбой о помощи в организации локальной сети для БК в Рязанский техникум электронных приборов, где было разработано коммутационно-согласующее устройство (КСУ), позволяющее осуществить:

загрузку программ на все БК учащегося с одного магнитофона или с БК преподавателя;

запись программ на магнитофон или загрузку программ в БК преподавателя с любого БК учащегося по выбору преподавателя; перезапись программного обеспечения с магнитофона на магнитофон (или несколько магнитофонов), минуя фазу загрузки БК. Коммутация сети производится переключателями с РМП.

Если в вашем распоряжении имеется ДВК-2 и вы хотите использовать его в качестве РМП, то вам придется пригласить специалиста или самим доработать ДВК по схеме, предложенной в статье В. Н. Завилова, М. Ю. Константинова, М. В. Померанца «Программирование на языке Паскаль для микро-ЭВМ «Электроника БК-0010» (Микро-

процессорные средства и системы. 1987. № 1). После этого вы сможете:

создавать программы на Паскале для БК и загружать их на все РМУ;

хранить программы, написанные на Фокале и набранные на РМУ, на гибком магнитном диске (этот носитель лучше магнитофонной ленты, поскольку запись на него более надежна);

загружать с ДВК на РМУ все программы, записанные с РМУ.

Коммутационное устройство было опробовано в Рязанском техникуме электронных приборов при длине соединительных кабелей до 15 м. Использовалось 16 каналов связи, но при необходимости их число может быть увеличено в 2, 4, 8 и т. д. раз.

При всех достоинствах устройству свойственны некоторые ограничения:

пониженная (по сравнению с промышленной) скорость загрузки класса — 1200 бод (обусловлена скоростью работы драйвера магнитофона БК);

несовместимость программ, записанных по системе ОС КЛАСС, и программ, записанных по каналу магнитофона; этот недостаток можно обойти, переписав программы по системе ОС КЛАСС из ДВК в БК и затем с помощью программы MAG.SAV переписав из БК в ДВК.

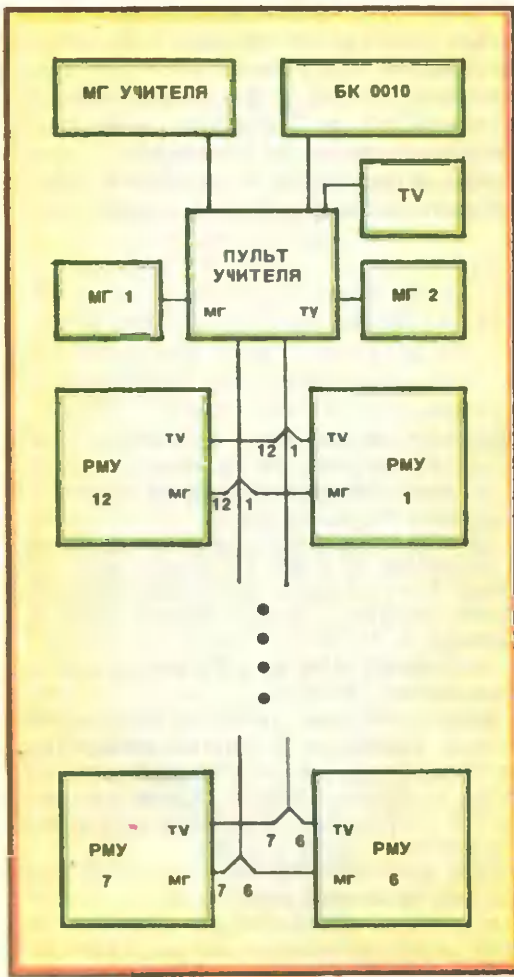
Всю информацию об устройстве можно получить по адресу: 390012, Рязань, ул. Циолковского, 19, РТЭП. Тел. 44-25-55.

С. ВАРАКС, С. ЯНОВСКИЙ

Объединение БК-0010 в сеть

Вариант объединения нескольких микро-ЭВМ БК-0010 в локальную сеть по магнитофонному каналу уже описывался в ИНФО.

Достоинства его очевидны: вдвое меньшая стоимость комплекса по сравнению с КУВТ-86, ненужность дефицитных дискет,

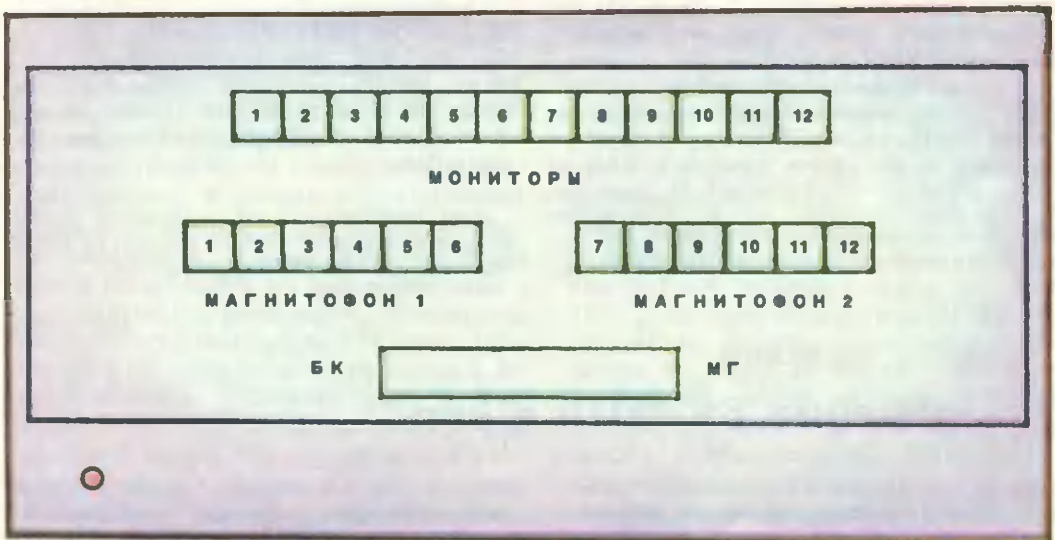


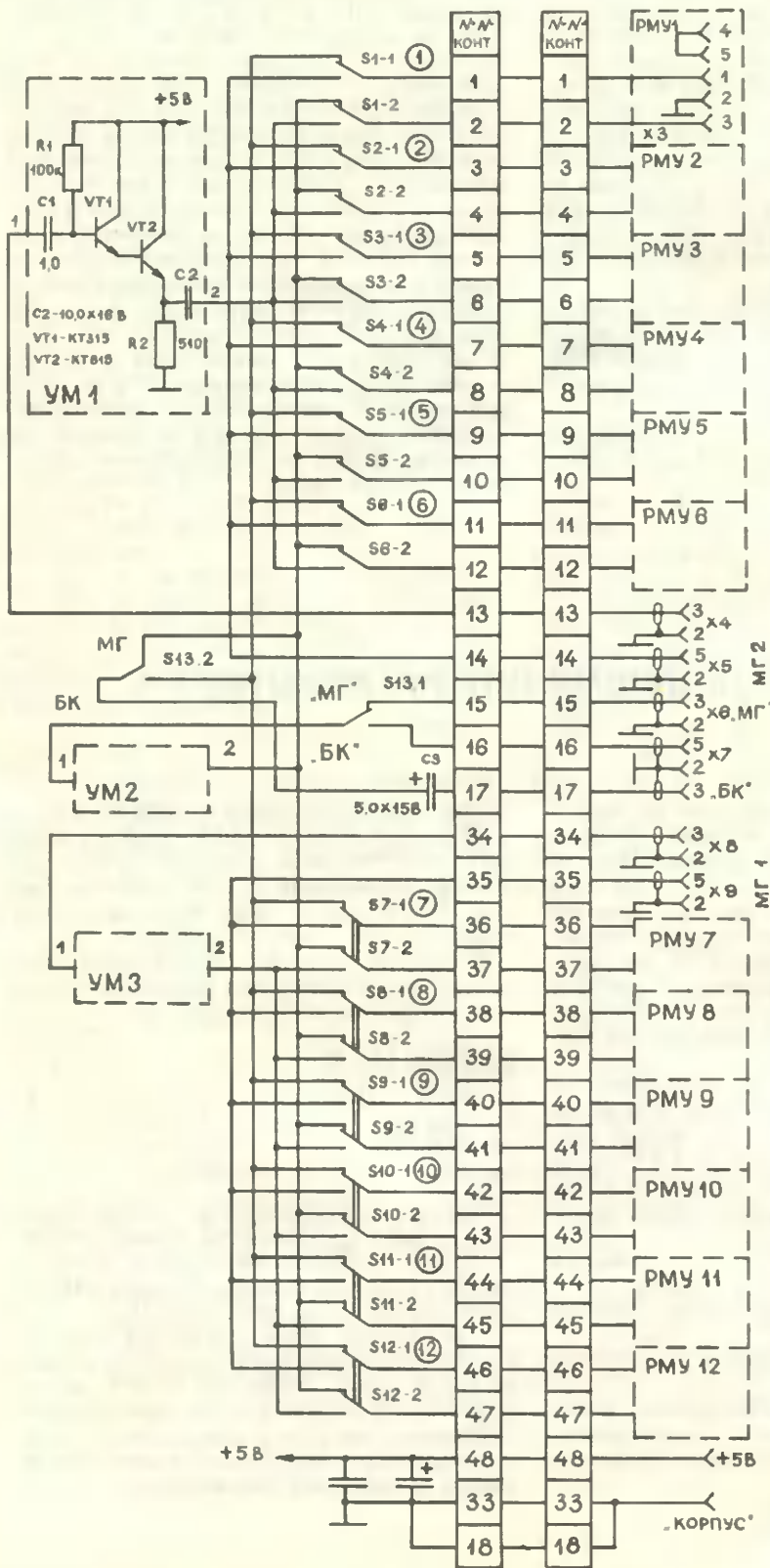
РП45-32 ГВФ X2

АДРЕС	№ КОНТ		
БК УЧИТЕЛЯ	1		
TV PMU 1	2	S14	1
TV PMU 2	3	S15	2
TV PMU 3	4	S16	3
TV PMU 4	5	S17	4
TV PMU 5	6	S18	5
TV PMU 6	7	S19	6
TV PMU 7	8	S20	7
TV PMU 8	9	S21	8
TV PMU 9	10	S22	9
TV PMU 10	22	S23	10
TV PMU 11	23	S24	11
TV PMU 12	24	S25	12
TV УЧИТЕЛЯ	25		
ЭКРАН	12		

TV

повышение надежности работы сети (за счет исключения ДВК-2М), существенно меньшая требовательность к профессиональной подготовке преподавателя по работе на ЭВМ. Дополнение описанного варианта сети рассылки





видеосетью снимает многие проблемы школьного урока и факультатива на сегодняшний день. Видеосеть позволяет преподавателю оперативно контролировать содержимое экрана РМУ на экране РМП, для чего достаточно нажать на пульте кнопку с номером требуемого РМУ, и копия экрана ученика возникает на экране преподавателя. Конструкция пульта очень проста и не содержит дефицитных деталей. Изготовление такой сети вполне доступно даже школьнику-радиолителю. Конструктивное исполнение может быть произвольным.

Возможности сети: загрузка с магнитофона РМП и (или) РМУ; загрузка непосредственно с РМП любых РМУ; выгрузка с любого РМУ на магнитофоны; просмотр содержимого экрана РМУ на экране РМП.

Недостаток: время загрузки увеличивается в 4 раза по сравнению с традиционным КУВТ-86.

58

О пересылке информации на принтер в КУВТ-86

В редакцию пришло сообщение из секции прикладной математики и вычислительной техники электростальского филиала Московского института стали и сплавов. Там была обнаружена возможность вывода на принтер или терминал РМП листингов программ и результатов их счета с любого РМУ без обращения к НГМД, не отраженная в документации на КУВТ-86 (речь идет о комплексе, включающем БК-0010Ш с вильнюсской версией Бейсика от 24.07.86).

При обращении к РМП может указываться не только спецификация файла, но и имена устройств. Если на системной дискете есть драйверы TT.SYS, MX.SYS и LP.SYS, то возможны следующие обращения к РМП:

«TT:XX» — файл XX записывается на системную дискету;

«TT:MX1:XX» — файл XX записывается на рабочую дискету нижнего кармана;

«TT:LP:» — файл выводится на печатающее устройство;

«TT:TT:» — файл выводится на терминал РМП.

Эти конструкции можно употреблять в операторах SAVE, OPEN и т. п. Например, команда SAVE «TT:LP:» выводит листинг программы на печать.

Учитывая существующий в настоящее время обмен ППС по КУВТ-86 на дискетах, было бы эффективно обеспечить место для просмотра и компоновки программ по типу методического кабинета, где необходимо установить ДВК-2М и БК-0010 со стандартной сетью обмена по ИРПС и магнитофоном, подключенным к БК. Любой школьный учитель района (города, области) мог бы просмотреть имеющийся фонд программ и сделать для себя выборку на магнитофонную кассету, получив консультации методиста. Программы, написанные преподавателями и учениками, могут, в свою очередь, пополнять фонд методкабинета. Созданием и ведением фонда занимается один человек в районе (квалифицированный методист), а не каждый учитель в отдельности, что, несомненно, высвобождает время учителя и не допустит на школьный урок неквалифицированно составленные программы.

При первом обращении к РМП по 3-му и 4-му форматам с системной дискеты считываются необходимые драйверы, при последующих обращениях это не требуется, так что НГМД можно даже выключить (при использовании монитора RT11SJ).

В качестве примера использования этих возможностей приводится программа вывода чисел и их квадратов на принтер:

```
10 OPEN'TT:PL:'
20 FOR I=11 TO 21
30 PRINT#'I=';I;' I*I=';I*I
40 NEXT I
50 CLOSE
60 END
```

Само собой разумеется, что одновременно должна работать программа связи ДВК-БК NET.SAV в режиме разрешения чтения и записи (в электростальском филиале МИСиС используется ее версия от 05.06.86).

К сожалению, вывод результатов счета таким способом имеет ряд ограничений. В частности, в списке оператора PRINT нельзя использовать запятые и встроенные функции, содержащие запятые в списке действительных параметров. Не найдена и возможность вывода графической информации.

Не всем известно, что БК-0010 — не единственный бытовой компьютер, имеющийся в продаже. Например, в московском магазине «Радиотехника» продают компьютер «Микроша». О нем рассказывает В. В. Дворцевой, активный пропагандист этой ЭВМ.

— «Микроша» — 8-разрядный компьютер на основе микропроцессора КР580ИК80А с быстродействием более 300 тыс. оп/с. Объем ОЗУ — 32К байт — легко увеличить до 48К, можно довести и до 64К байт. Объем ПЗУ — 16К байт. В качестве монитора используется любая телевизор — подключение осуществляется через антенное гнездо. Внешнее запоминающее устройство — магнитофон. В качестве печатающего устройства предусмотрен принтер EPSON (или CPA-80), но в комплект он не входит. Цена компьютера — 500 руб. Кроме Москвы он продается в Казани и Саратове.

— Какова его надежность?

— При испытании 30 машин (их не выключали месяц) вышел из строя лишь один светодиод. Характерный для БК-0010 с полноходовой клавиатурой дребезг клавиш у «Микроши» отсутствует.

— Можно ли порекомендовать эту машину начинающим?

— Да, конечно. Для человека с нулевым уровнем знаний по программированию «Микроша», пожалуй, даже лучше БК-0010: с ним проще «общаться», он подсказывает, чего

«хочет» в данный момент, проще и работа с клавиатурой...

— А что ожидает имеющих опыт программирования?

— Для них к «Микроше» есть хороший набор программ. Это ассемблеры, дизассемблеры, несколько интерпретаторов Бейсика (Бейсик+, Бейсик-best), графика Бейсика, эмулятор CP/M и др. Кстати, аналогов Бейсика+ и особенно Бейсика-best (IBM-овского) на компьютерах такого класса нет.

Имеются и прикладные программы. Текстовый редактор «Мастер-текст» В. Кузнецова позволяет одновременно держать в ОЗУ 15 страниц машинописного текста и предоставляет пользователю все удобства — аналогичных редакторов нет даже на более крупных ПЭВМ. Собственно, обо всех программах в коротком интервью не расскажешь — есть англо-русский и частотный словарь, программа КАЛЬКУЛЯТОР В. Дворцевого, выполняющая основные функции системы Supercalc (электронные таблицы, построение графиков), игровые программы...

59

Дорогие читатели! Если вас заинтересовал «Микроша», напишите нам, что еще вы хотели бы узнать о нем и его программном обеспечении. Если вы уже приобрели этот компьютер и у вас есть какие-то конкретные вопросы — можете звонить Владимиру Викторовичу ДВОРЦЕВОМУ по тел. 275-03-04.

ВНИМАНИЮ ЧИТАТЕЛЕЙ!

Центр информационных технологий и информационного обслуживания Высшего института народного хозяйства им. Д. Благоева в г. Варне (НРБ) приглашает заинтересованные организации принять участие в международном семинаре «Компьютеризация обучения в вузе», который будет проведен 7—10 мая 1989 г. в рамках сотрудничества стран-членов СЭВ в области применения ЭВТ при подготовке кадров.

Информацию с предложениями об участии в семинаре (аннотация доклада, научного сообщения, описание ППП, предлагаемого для демонстрации) высылать не позднее 15.12.88. Подтверждение о включении предложенных материалов в программу семинара — 15.02.89.

Срок высылки докладов — не позднее 15.04.89.

Адрес: Болгария 9002, Варна, ВИНХ, Центр информационных технологий и информационного обслуживания. Семинар «Компьютеризация обучения в вузе».

Л. ВИКЕНТЬЕВ,

доктор техн. наук

О. КОЗЛОВ,

канд. техн. наук, Пермь

Что дает учебный язык с русской лексикой

Проблемы школьной и вузовской информатики во многом схожи. Поэтому мы были приятно удивлены, прочитав три года тому назад в журнале «Микропроцессорные средства и системы» статью академика А. П. Ершова о предлагаемом им алгоритмическом языке для школьного курса «Основы информатики и вычислительной техники». Дело в том, что к этому времени мы уже имели свою разработку подобного языка, отличавшегося от школьного в деталях, был у нас и некоторый опыт его использования в учебном процессе. Мы назвали его алгоритмическим языком для представления алгоритмов решения инженерных задач (АЛРИЗ). Он реализован в виде претранслятора к языку ПЛ/1 на ЕС ЭВМ, сам претранслятор послужил основой для создания аналогичной версии школьного языка (статьи об этом были напечатаны в журнале «Информатика и образование» № 3 за 1986 г. и № 1, 2 за 1987 г.).

За три года эксплуатации несколько видоизменился сам язык, изменилось и наше представление о возможных формах его использования.

При создании и реализации языка мы исходили из предпосылки, что он должен помочь инженеру в решении его профессиональных задач, поэтому диагностические сообщения выдаются на русском языке. Основу АЛРИЗа составляют ключевые слова также на русском языке, «гибкий» синтаксис позволяет не

акцентировать внимание на наличии или отсутствии знаков препинания типа «;» в конце строки и правильно понимать смысл действия, даже если в операторе написано ЦЫКЛ или КАНЕЦ.

Сначала мы попробовали применить АЛРИЗ в «устоявшемся» вузовском курсе, где, дав за 20 ч понятие об алгоритмизации, мы в течение 40 ч буквально вдавливали в головы своих учеников тонкости языка ПЛ/1. Существенного эффекта это не давало. Требовалась новая методика.

Уже первый опыт показал, что АЛРИЗ может стать средством программной поддержки раздела «Алгоритмизация». На первых же занятиях можно сажать обучаемых к экранам дисплеев и «прокручивать» только что составленные у доски алгоритмы в виде псевдокодов. Интерес к предмету при этом повышается, но часов на начальный этап отводится мало, и дальше все заслоняется тонкостями производственного языка, о непригодности которого для использования в учебном процессе можно прочитать в каждом номере журнала «Информатика и образование».

Стало ясно, что пересматривать нужно всю систему начального обучения, в первую очередь — его содержание.


Это мы и пытались сделать.

Во-первых, чему учить? Самое главное — не какому-то конкретному языку, а принципам решения инженерных (профессиональных) задач с использованием

ЭВМ, т. е. на первый план выступает изучение идей алгоритмизации. В такой постановке конкретный производственный язык программирования выступает как одна из возможных форм представления алгоритма (но не единственная).

С первых же занятий нужно акцентировать внимание не на системах счисления и адресах переменных, а на технологиях программирования, на возможностях ЭВМ будущих поколений. Тогда учебный язык играет роль своего рода учебного самолета, освоение которого для будущих летчиков не является самоцелью, но в то же время облегчает переход к штатной технике.

Учебный язык можно применять уже на первых практических занятиях. С его помощью обучаемые решают первые задачи, начинают понимать специфику общения с компьютером. Этап же освоения производственного языка можно при такой постановке курса существенно сократить за счет формализации правил перехода от алгоритмов и учебного языка к производственному. Здесь особенно помогают опорные сигналы типа

Блок-схема	Псевдокод	ПЛ/1
	ЕСЛИ...	IF...

Такие «таблицы кодирования» существенно облегчают освоение производственного языка, и когда на последних занятиях мы обращаемся к аудитории с вопросом: «А что нужно было бы сделать, чтобы научить вас еще Бейсику и Паскалю?», — слушатели хором отвечают: «Отчеркнуть еще две колонки!»

Но такая схема хорошо работает лишь при наличии программного обеспечения и адекватной организации работ в ВЦ. Как можно раньше нужно начинать решать задачи в дисплейном классе, и не произвольные задачи, а по профилю подготовки.

В этом случае преподаватели смежных дисциплин, даже не владеющие программированием, могут консультировать обучаемых.

Опыт авторов показывает, что нали-

чие русского в своей основе алгоритмического языка, реализация которого имеет развитую диагностику также на русском языке, снимает для многих обучаемых (в том числе и на различных курсах повышения квалификации) психологический барьер боязни ЭВМ, который часто возникает при решении первых задач. Его порождают в первую очередь «заорганизованность» обслуживания пользователя в больших ВЦ и англоязычность программного обеспечения, диагностических сообщений и средств диалога с ЭВМ. Поэтому в первоначальный период самостоятельного решения задач преподавателей буквально осаждают обучаемые, в руках которых, как хорошо сказано в одной из популярных книг, «... чушь несусветная, английскими буквами на бумаге напечатанная». При этом главной потерей является не потеря времени, хотя его затраты на преодоление таких трудностей у преподавателей и у обучаемых весьма велики, а потеря уверенности в себе. При использовании же на этом этапе языка с русскими ключевыми словами и диагностическими сообщениями только в редких случаях у обучаемых возникает необходимость обращаться к преподавателям с вопросами по смыслу допущенных ошибок. В результате появляется уверенность в своих силах, резко сокращается календарное время решения, что позволяет с самого начала обращаться к задачам достаточной степени сложности, особое внимание уделяя профессиональной ориентации. Опыт показал, что на начальном этапе обучения нецелесообразно решать чисто «программистские» задачи типа перестановки столбцов матриц и т. п.

Для проверки справедливости выдвинутых педагогических гипотез был проведен эксперимент. Занятия проводились в одинаковой сетке часов для первокурсников двух групп электротехнического профиля. Контрольная группа обучалась по обычной схеме, экспериментальная — с упором на алгоритмизацию с использованием АЛРИЗ. На экзамене обе группы проверялись на умение составить алгоритм, прочитать и написать программу на языке ПЛ/1.

Прежде скажем о качественном отли-

и порядок выполнения операций по переработке информации. В соответствии с правилами структурного программирования во входном языке выделяют три типа операторов: линейные, разветвления и циклические.

Линейные операторы. Для изменения значений переменных в процессе решения задачи служит оператор присваивания

`|X=A |X:=A;`

где A — арифметическое выражение, X — переменная, элемент массива или имя массива.

Обмен информацией осуществляется операторами

`ВВОД (<список ввода>);`

`ВЫВОД (<список элементов вывода и форматов>);`

`ВЫВОД ТЕКСТА (<список вывода>);`

`ВЫВОД СТРОКИ (<список вывода>);`

`ВЫВОД ПОИМЕНОВАННЫЙ (<список переменных>);`

Стандартные и собственные процедуры могут быть вызваны оператором

`ВЫЗОВ_(имя процедуры) (<список фактических значений>);`

Разветвления. Изменить естественный порядок выполнения операций можно с помощью операторов

`|ПЕРЕЙТИ_|{K|NA}|ИДТИ{K|NA}|<обозначение оператора>;`

`ЕСЛИ_(<условие>){|_|, ПЕРЕЙТИ {K|NA} <обозначение оператора>;`

`ЕСЛИ_(<условие>){|_|, _ТО:|_|; <операторы ветви да>`

`ИНАЧЕ{|_|:|_|; <операторы ветви нет>`

`КОНЕЦ — ЕСЛИ;`

Циклы. Циклические вычислительные процессы реализуются с помощью конструкций `<МЕТКА> <ЗАГОЛОВОК ЦИКЛА>;`

`|ТЕЛО ЦИКЛА|`

`<МЕТКА> КОНЕЦ — ЦИКЛА — <МЕТКА ЗАГОЛОВКА>;`

Заголовок цикла может иметь вид

`ЦИКЛ ПОКА (<условие>);`

`ЦИКЛ ДО (<условие>);`

`ЦИКЛ ПО <переменная> ОТ <начальное значение> ДО <конечное значение> ШАГ <величина изменения>;`

Тело цикла может быть группой операторов любой степени сложности.

Неэлементарные операторы. Включение неэлементарных операторов в состав входного языка позволяет непосредственно в процессе разработки программы осуществлять последовательную детализацию в соответствии с принципами структурного программирования. Приведем пример использования неэлементарных операторов и их последующей детализации.

операции по вычислению и печати максимального элемента массива X
318 КОНЕЦ 154;

Процедуры. Для процедур, разрабатываемых в программе, необходимо составлять их описание, которое имеет вид `|ПРОЦЕДУРА|ПОДПРОГРАММА | ПРОГРАММА|ПП|_(<имя>) (<список формальных параметров>);`
`КОНЕЦ_|ПРОЦЕДУРЫ | ПОДПРОГРАММЫ|ПРОГРАММЫ|ПП|_(<имя>);`

В теле процедуры для завершения ее работы могут стоять один или несколько операторов `|ВЫХОД|;`

Руководство пользователю. Претранслятор входного языка оформлен в виде каталогизированной процедуры ОС ЕС ЭВМ. Для вызова претранслятора перед текстом программы и после него необходимо записать управляющие операторы, начинающиеся с 1-й позиции стандартного бланка.

`//<имя>_|JOB_(<учетная информация>), <фамилия автора>`
`_|EXEC_|ALRIZ`
`//ALRIZ.SYSIN DD_*`
`<текст программы на входном языке>`
`/*`
`//`

Если претранслятор обнаружит ошибку в исходном тексте программы (в логике работы операторов, в синтаксисе ключевых слов), то после печати текста программы и диагностических сообщений на русском языке дальнейшая обработка задачи будет прекращена. Об отсутствии таких ошибок сообщит фраза **ЗАВЕРШЕНО ФОРМИРОВАНИЕ РАБОЧЕЙ ПРОГРАММЫ**, после чего будет продолжена обычная обработка сформированной ПЛ-программы.

Отсутствие ошибок, обнаруженных претранслятором, не может служить гарантией отсутствия ошибок вообще. При их обнаружении будет работать вся диагностика ПЛ/1, а после текста программы на входном языке отпечатается результирующая программа на ПЛ/1 и сообщения об ошибках в ней.

Использование претранслятора на 30—50 % увеличивает общее время трансляции, однако количество выходов на машину для отладки задачи, как правило, уменьшается.

Возможные сокращения ключевых слов.
`AL|ГОРИТМ| PR|ОГРАММА| НАЧ`
`[АЛО] СЛ|ОВАРЬ| П|ЕРЕМ|ЕННЫЕ|`
`СП|ИСОК|_|ПЕРЕМЕННЫХ| K|ОНЕЦ|`

63

ИС[ХОДНЫЕ ДАННЫЕ]

*** [А...А] ВЫВ[ОД] Т[ЕКСТА]
[С]Т[Р[ОКИ]] Т[АБ[ЛИЦЫ]] [ПО]
И[МЕНОВАННЫЙ] П[ЕРЕ]Й[ТИ] [К]
НА] ИД[ТИ] [К] [НА] ЕСЛ[И] ИН[АЧЕ]
Ц[ИКЛ] ПОК[А] ПР[ОЦЕДУРА]
П[ОД]П[РОГРАММА]
ВЫХ[ОД] ВЕЩ[ЕСТВЕННЫЕ] Ц[ЕЛЫЕ]
СИ[МВОЛЬНЫЕ] Ш[АГ]
ДЕ[СЯТИЧНЫЕ] КОМП[ЛЕКСНЫЕ]
УП[РАВЛЯЮЩИЕ] МЕТОЧ[НЫЕ]
ЛО[ГИЧЕСКИЕ] БУЛ[ЕВЫЕ] ЛИ[ТЕР-
НЫЕ] БУК[ВЕННЫЕ].

Ниже приведен текст программы для построения таблицы значений функции $\sin^2 x$. Читатель может убедиться, что для прочтения программы на языке АЛРИЗ не нужно никакой квалификации.

НАЧАЛО

ВЫВОД ТЕКСТА ('ДЕМОНСТРАЦИОННЫЙ ПРИМЕР')

ВЫВОД ТЕКСТА ('АРГУМЕНТ' ; 'ФУНКЦИЯ')

10 ЦИКЛ ПО X ОТ 0 ДО 3.1415/2
ШАГ 3.14/2

F:=SIN(X)**2;

ВЫВ СТРОКИ (X, F);

20 КОНЕЦ — ЦИКЛА ;

30 ВЫВ Т ((48) '=')

КОНЕЦ ПРОГРАММЫ

Оператор 30 печатает черту из 48 знаков «равно».

Заинтересованных лиц мы можем проконсультировать дома по тел. 44-63-94 (ВИКЕНТЬЕВ Леонид Федорович) и 33-33-53 (КОЗЛОВ Олег Александрович) с учетом разницы во времени в Перми с Москвой в 2 ч.

От редакции. За время подготовки статьи к печати у авторов появилась новая, на наш взгляд весьма интересная, информация, которую мы и предлагаем вниманию читателей.

В школах г. Перми и области широко используется предлагаемый авторами подход, основанный на разработке средств программного обеспечения алгоритмического языка школьного курса для вычислительной техники крупных вычислительных центров (ВЦ) и решений задач школьников в пакетном режиме.

Первая версия программного обеспечения языка персональных вычислений (ПВ), являющегося расширением алгоритмической нотации школьного курса,

была составлена на базе пробного учебного пособия для IX класса.

К концу 1987 г. в Пермском политехническом институте была полностью завершена разработка и экспериментальная эксплуатация программного обеспечения для второй версии языка ПВ, в которой предусматривается использование в полном объеме всех конструкций, приведенных в пробном учебном пособии для IX и X классов. Кроме того, во второй версии снимается существующее в первой версии ограничение на применение русских букв в именах объектов программы и предусматривается автоматическое исправление более широкого класса ошибок, допускаемых разработчиками ПВ-программы.

Использование мощностей существующих ВЦ, оснащенных ЕС ЭВМ, несомненно позволяет уже сегодня, без приобретения и установки новой техники, предоставить школьникам (в том числе и сельским) возможность решать на ЭВМ все задачи курса ОИВТ по составлению алгоритмов и программ в пакетном режиме, пусть даже и заочно, по почте.

Для обеспечения такой работы в Пермской области пришлось решить ряд организационных вопросов — какую школу закрепить за каким ВЦ, каким образом и за счет каких средств компенсировать материальные и трудовые затраты ВЦ, как подготовить к такой работе учителей. Потребовались совместные усилия облоно, областного ИУУ, ВЦ Пермского политехнического института, кафедры информатики Пермского педагогического института. Были прочитаны лекции учителям; помогли 1—2-й номера «Информатики и образования» за 1987 г., а сейчас вышло и учебное пособие, дополняющее существующие учебники практическими рекомендациями (Викентьев Л. Ф., Козлов О. А. Язык персональных вычислений: Учебно-методическое пособие. Пермь, 1988. 96 с.).

Справки о приобретении программного обеспечения для второй версии языка ПВ можно получить у директора ВЦ Пермского политехнического института Кострова Алексея Владими-

ровича по тел. 39-14-27 или по адресу: Пермь, 614000, Комсомольский проспект, 29-А, Политехнический институт, вычислительный центр.

Прошедший учебный год позволил нам сделать некоторые выводы о том, что дало для вузовского курса «Основы устройства и применение вычислительной техники» (раздел «Алгоритмизация и программирование») введение в школе курса ОИВТ, который изучала большая часть первокурсников. Существенного улучшения усвоения вузовского курса, к сожалению, не произошло. Некоторые любопытные замечания позволяют сделать следующие статистические данные.

На рассматриваемом курсе из 152 человек 125 (82 %) учили в школе информатику.

40 человек имели в школе «5» по ОИВТ, 67 % из них подтвердили эту оценку, 26 % получили «4» и 7 % — только «3» на вузовском экзамене. Это говорит о том, что в целом отличной оценке школьных учителей можно верить.

В. ВЕЙЦМАН

Старт в информатику

Экспериментальный курс ОИВТ, предложенный вниманию читателей, делится на 2 части, но это не просто механическое разделение материала между IX и X классами. Первую часть (материал IX класса) мы рассматриваем как теоретические основы информатики. Во второй части акцент сделан на практическое применение полученных знаний. Конечно, возможности доступа к вычислительной технике у школ могут быть различными. Поэтому и «обкатка» нашего курса проводилась в различных условиях. На основании этого опыта мы предлагаем три уровня работы с курсом.

Продолжение. Начало в № 4 за 1988 г.

69 человек имели «4» по ОИВТ, 47 % подтвердили эту оценку, 17 % улучшили и 34 % ухудшили ее на экзамене. Так что школьное «хорошо» очень «размыто».

Интересна картина среди имевших по ОИВТ «3», а их было 26 человек. Только 26 % из них не сумели повысить эту оценку, 31 % получили «4» и 43 % — «5»! Что можно сказать об этом? Либо за три месяца резко переменилось отношение к учебе, либо их в школе плохо учили.

Из 27 не учивших в школе информатику 37 % получили «3», 37 % — «4» и 16 % — «5».

Что ж, школьный курс пока только «встает на ноги»; и учителям, и ученикам многому еще предстоит научиться, но один момент настораживает. Изучавшие машинный вариант хорошо ориентируются в Бейсике и клавиатуре, но вот азы алгоритмизации они не усвоили.

Мы сделали для себя вывод, сделайте их и вы, школьные учителя.

65

I уровень (68 ч, наличие класса ПЭВМ, имеющих графику, музыку и т. д., или регулярный доступ к такому классу).

Первое полугодие X класса, примерно 34 ч, — логическое продолжение курса IX класса. После знакомства с общими принципами компьютерной обработки информации и первых проб в составлении алгоритмов для ЭВМ учащиеся получают курсовую работу сроком на полугодие. Содержание этой работы — придумать и реализовать на ЭВМ диалоговую программу обучающего характера или типа «экзаменатор» по какому-либо предмету. Выбор предмета остается за учащимися.

Курсовая работа разбивается на несколько этапов:

постановка задачи («чего мы хотим добиться?»);

дизайн программы (разработка сценария программы, выбор текста, оформление кадров программы);

разработка алгоритма (здесь могут успешно применяться блок-схемы) и его запись на алгоритмическом языке программирования, т. е. составление программы; отладка программы.

Разумеется, в одиночку даже десятикласснику не просто справиться с такой работой. Поэтому курсовую работу обычно выполняет бригада из 2—3 учеников. С одной стороны, старшеклассники приучаются к коллективному труду. С другой — для каждого члена бригады появляется возможность сосредоточиться на том, что ему ближе (например, один специализируется на сценарии программы, другой на разработке алгоритмов, третий — на графике или музыке). Не исключен, конечно, эффект «лебедя, рака и щуки», поэтому для учителя очень важно быть в курсе работы бригад.

В своей практике автор ориентируется на КУВТ «Ямаха». Отсюда и критерии оценок. Работающая диалоговая программа, содержащая элементы графики (и, возможно, музыки), заслуживает отличной отметки. Диалог без использования графических и музыкальных элементов оценивается четверкой, если он дружественный (т. е. любому новичку-пользователю ясно, как действовать). Наконец, недружественная, примитивная по исполнению, но работающая программа оценивается удовлетворительной отметкой.

Столь большое внимание к «сервису» программы, удобству работы с ней приучало школьников к мысли, что программный продукт они создают не для себя, а для других. Это, естественно, повышает интерес, уважение и ответственность ученика по отношению к собственному труду. Мы часто предлагаем ребятам привести своих младших братьев и сестер или просто знакомых поработать с их программой. Тут оценка, как говорится, налицо.

Из-за ориентации на КУВТ «Ямаха» возникла необходимость изучить язык Бейсик. На его изучение вполне хватает одной четверти. Материал по Бейсику подготовлен как приложение к учебному пособию. Он построен точно так же, как и материал по языку КАЯ из курса IX класса. Это позволяет на уроке уделять внимание главным образом особенностям Бейсика, оставляя большую часть материала для самостоятельного изучения. Переходя с КАЯ на Бейсик, ученики сравнивают их конструкции и, находя аналогии, легко осваивают новый язык. Правда, при изучении Бейсика упор делается не столько на примеры обработки данных (в частности, файлов, как в КАЯ), сколько на диалог, графику и другие элементы языка, которыми можно «украсить» курсовую работу. Таким образом, освоение Бейсика идет параллельно с первыми двумя этапами курсовой работы.

Так как работу над постановкой задачи и дизайном программы учащиеся ведут в основном самостоятельно, во внеклассное время, работа в классе (без ЭВМ) сводится к проверке выполнения курсовых работ бригадами, обсуждению спорных вопросов и изложению порции нового материала по Бейсику для его последующего закрепления на практических занятиях в компьютерном классе (2-й час ОИВТ в неделю).

Во второй четверти все внимание сосредоточено на курсовой работе. Урок в классе посвящается оценке текущей работы бригад, разбору типичных ошибок и простейшим приемам программирования. Урок в компьютерном классе — это практически отладка курсовых работ. Кстати, отладкой совсем не обязательно заниматься всей бригаде. Иногда это делает лишь тот, кто отвечает за данный этап работы.

Практика показала, что у учащихся лучше получаются программы типа «экзаменатор», нежели обучающие программы. В этом нет ничего удивительного. Увы, ориентация на поверхностное зазубривание преобладает и в большинстве школьных учебников, и в объяснениях учителей. Краткое, емкое

и красочное объяснение — редкость в современной школьной практике. Откуда же у учеников могут развиваться качества, необходимые для создания обучающих программ?

Теперь немного об опыте подобных курсовых работ. Конечно, их уровень далеко не одинаков. Большинство учеников сразу же пытаются начать с самого «красивого» — графики, мультипликации, мигающих надписей и т. д. Усилия, затрачиваемые на достижение этих эффектов, зачастую идут в ущерб всей работе. У некоторых опускаются руки, едва возникают трудности. В этом случае необходимо убедить ребят заново продумать алгоритм, может быть, опуститься «на ступеньку ниже» по сложности, чтобы потом вновь попытаться достичь желаемого. Но большинство успешно справляется с заданиями. Именно из таких курсовых работ и родился комплекс обучающих и контролируемых программ к главе I курса. Часть из них мы демонстрировали этой весной в Ленинграде, в дни Всесоюзной конференции «Школьная информатика-88». Несмотря на то что их авторы — самые обыкновенные школьники, программы выгодно отличаются от некоторых «профессиональных» изданий (вроде новосибирской программы «Обучение графике Бейсика»), которыми многие все еще вынуждены пользоваться. Они веселые, красивые и, самое главное, дружественные по отношению к начинающему пользователю. Над первой версией работали ученики двух таллинских школ — 51 и 58, и уже сейчас девятые классы с успехом используют эти программы на начальном этапе обучения.

Для школ, имеющих более или менее регулярный доступ к многопользовательским ЭВМ (типа ЕС ЭВМ или СМ ЭВМ), больше подходит II уровень нашего курса, так сказать, промежуточный (количество часов колеблется в интервале от 34 до 68). В этом случае также возможна организация и проведение курсовых работ. Однако в зависимости от используемой техники следует изменить задания. Скажем, вместо обучающих программ можно заняться разработкой различных задач из

школьной жизни (фрагменты расписания уроков и т. д.). Изменятся и критерии оценки, поскольку с той же графикой на ЕС ЭВМ или СМ ЭВМ возникают сложности. В качестве языка для реализации алгоритмов этих задач можно использовать любой из трех предлагаемых в нашем курсе языков: КАЯ, Бейсик или Фортран-77 (последние два предлагаются в качестве приложений). Выбор языка зависит главным образом от содержания задачи и от типа ЭВМ. При реализации курсовых работ, например, на ЕС ЭВМ нет необходимости учить дополнительно какой-либо язык программирования, поскольку в этом случае КАЯ, безусловно, предпочтительнее всех остальных языков. Именно для этих целей в главе 2 курса IX класса предложен материал, посвященный его дополнительным возможностям.

67

Что же касается школ, обреченных на безмашинный вариант информатики, то им вряд ли есть смысл проводить курсовые работы, а также изучать дополнительный язык. Обычно в таких школах информатика X класса ведется в объеме 34 ч, в нашем курсе этому соответствует III уровень (программинимум), по которому, минуя курсовую работу, можно переходить сразу к материалу главы 3.

После завершения знакомства с логикой компьютера, с программированием следующий шаг — знакомство непосредственно с вычислительной техникой. Глава 3. Архитектура ЭВМ.

3.1. Конфигурация ЭВМ (2 ч)

3.2. Внешние устройства ЭВМ (4 ч)

3.3. Классификация ЭВМ (2 ч)

Название и материал этой главы выбирались исходя из существующего представления: «архитектура ЭВМ — это устройство ЭВМ, как его представляет себе пользователь». Подчеркиваю — именно пользователь. Специалисты в области вычислительной техники трактуют понятие «архитектура ЭВМ» в несколько ином смысле, примерно так, как это сделано в действующем учебнике для X класса.

На наш взгляд, эта тема — одна

и, важнейших. Поэтому материал для нее подбирался тщательно, по принципу «как можно меньше лишнего, как можно больше полезного». Сегодняшний школьник — это будущий пользователь ЭВМ. В рамках школьного курса ему необходимо дать представление обо всем, что может способствовать наиболее эффективному использованию компьютера. Но не более того! Поэтому мы сознательно пошли на значительные упрощения и решительно исключили такие вопросы, как, например, основной алгоритм работы процессора. Это — хлеб программиста, да и то не всякого. Пользователю не до таких тонкостей! Ему необходимо иметь представление о конфигурации машины и о назначении ЭВМ различных классов (микро-, мини-, больших универсальных ЭВМ) для того, чтобы в будущем оценить, какая ЭВМ и в каком качестве может стать оптимальной в его деятельности. И не повторять промахи малосведущих людей, использующих большие ЭВМ в тех случаях, когда можно ограничиться персональным компьютером и, наоборот, «вешающих» на мини-ЭВМ задачи, для решения которых необходима большая универсальная ЭВМ.

Нам кажется совершенно необходимым познакомить будущих пользователей с элементами конфигурации ЭВМ, ее внешними устройствами, в том числе и самыми современными, такими, как видеодиск, диджитайзер, фотосканер. При этом мы стремимся свести к минимуму технические подробности, акцентируя внимание учащихся на вопросах применения этих устройств. Кроме того, нелишне дать им некоторые количественные представления, например, какой объем основной памяти компьютера считается большим, а какой малым.

При изложении материала главы 3 мы старались избежать ориентации на какой-либо конкретный класс ЭВМ и рассматривать то, что характерно для любого современного компьютера.

Задачи, предлагаемые в этой главе, носят преимущественно «менеджерский» характер. Например, ребятам предлагается попробовать себя в роли

директора школы, компьютеризирующего свое учебное заведение. Такие задачи поначалу вызывают робость учащихся, поскольку в них как бы отсутствует единственно верный ответ. Однако скванность быстро проходит, а опыт их решения способствует не только закреплению материала, но и общему развитию (повышению уровня интеллекта, логике и самосознания) ученика, помогает ему раскрывать взаимосвязи в окружающем мире и принимать обоснованные решения.

Глава 4. Программное обеспечение ЭВМ.

- | | |
|--|-------|
| 4.1. Назначение и классификация программного обеспечения | (1 ч) |
| 4.2. Операционная система ЭВМ | (3 ч) |
| 4.3. Системные сервисные программы | (3 ч) |
| 4.4. Трансляторы языков программирования | (3 ч) |
| 4.5. Прикладное программное обеспечение | (4 ч) |

Знания, полученные в IX классе (и по программе I или II уровня в X), позволяют представить в общих чертах «анатомию» программного обеспечения. Знакомство с архитектурой ЭВМ сформировало представление о том, «откуда что берется». Поэтому к изучению программного обеспечения ученик приступает достаточно подготовленным.

На сегодняшний день практически не существует четкой системы отбора учебного материала по этой теме, во всяком случае если судить по нынешнему учебнику X класса. Более того, сама терминология, используемая в сфере программного обеспечения, достаточно путаная и неустоявшаяся. В этом смысле мы чувствовали себя достаточно свободно и уверенно. Разумеется, не все соглашались с предложенными нами понятиями и терминологией, однако, еще раз повторяю, мы сознательно упрощаем многие вещи, предназначая свой материал прежде всего ученикам и начинающим преподавателям этого предмета.

Понимая под термином «программное обеспечение электронно-вычислительной машины» все программы, кото-

рые могут работать на данной машине, мы постарались дать упрощенную классификацию всего программного обеспечения исходя из следующего принципа. По назначению программное обеспечение ЭВМ можно разделить на системное («сервис» компьютера) и прикладное (программы для решения конкретных задач). По характеру работы — на пакетное (минимальное взаимодействие с пользователем или полное отсутствие такового) и диалоговое (постоянное взаимодействие с пользователем).

Знакомство с операционной системой обусловлено ее ведущей ролью в программном обеспечении современных ЭВМ. Любой пользователь так или иначе вступает с ней в контакт. Однако изложение такого сложного понятия, как операционная система, мы постарались дать в форме максимально доступной. Школьнику достаточно знать, что «операционная система — это набор программ, организующих взаимодействие пользователя с аппаратурой ЭВМ».

Различные редакторы (программные, текстовые, графические и т. д.) и мониторы (программы управления задачами, «надстройки» над ОС, экранное меню) объединены в теме «Системные сервисные программы». Они непосредственно связаны с операционной системой и дополняют ее «сервис» (отсюда и название). Но, поскольку они не являются непременным атрибутом ОС, мы рассматриваем их отдельно, как тему «Трансляторы языков программирования», содержащую сведения о типах трансляторов и классификации языков программирования на машинно-независимые и машинно-ориентированные.

Последняя тема главы 4 посвящена прикладному программному обеспечению. На материале этой темы стоит остановиться более подробно. Во-первых, мы предлагаем деление задач, решаемых с помощью ЭВМ, на три большие группы: инженерные, коммерческие и задачи искусственного интеллекта. Рассматриваются и задачи смешанного типа (АСУ, САПР, АРМ, интегрированные системы). Учащиеся получают

представление о том, какими средствами создаются программы; какие языки программирования используются для реализации различных типов задач. Наконец, рассматриваются вопросы, связанные с пакетами прикладных программ и их применением.

Задачи и контрольные работы по темам главы 4 аналогичны задачам к главе 3. Это все те же «деловые игры», т. е. задачи на принятие решений, но более высокого уровня, комплексные. Ориентируясь прежде всего на старшеклассников средних школ, мы считаем их, в большинстве своем, потенциальными студентами вузов и техникумов, и, следовательно, в дальнейшем им так или иначе придется принимать решения. Поэтому, на наш взгляд, именно уроки информатики нужно использовать для начальной подготовки к такой деятельности. Тем более что принять правильное решение на основании полученной информации — это тоже элемент информационной культуры, основы которой мы стремимся заложить (см. контрольную работу).

В какой мере необходима вычислительная техника при изучении глав 3 и 4? Конечно, в идеале следовало бы каждого ученика обеспечить автоматизированным рабочим местом на базе самой современной ЭВМ, снабженной различными периферийными устройствами и соответствующим программным обеспечением. Но такая картина может стать реальностью лишь в далеком будущем. Пока же можно обойтись и обычным комплектом учебно-вычислительной техники. Любой КУВТ содержит почти все, о чем рассказывается в главах 3 и 4, за исключением некоторых периферийных устройств. Если же нет и такого класса (весьма распространенная ситуация), мы советуем учителям, не суетуя на обстоятельства, найти способ организовать несколько экскурсий в вычислительные центры. На любом из них есть почти все, о чем шла речь (кроме, может быть, джойстиков, «мышек», сканеров и прочей периферии). На уроках в классе можно использовать и фотографии, которые сейчас в изобилии встречаются в различных журналах, особенно рекламных.

Глава 5. Мир компьютеров.

5.1. ЭВМ вокруг нас (3 ч)

5.2. История вычислительной техники (2 ч)

70 Главу 5 можно считать заключительным аккордом курса информатики. Она содержит сведения преимущественно познавательного характера и направлена на повышение эрудиции и расширение кругозора учащихся в вопросах вычислительной техники. В теме «ЭВМ вокруг нас» приводятся сведения о наиболее распространенных на сегодняшний день в СССР моделях ЭВМ и сферах их применения. Учащиеся знакомятся с такими понятиями, как совместимость ЭВМ, сети ЭВМ и мультипроцессорные комплексы. Завершает наш курс ОИВТ раздел, посвященный истории вычислительной техники.

Читатели, наверное, обратят внимание на отсутствие отдельной темы, посвященной применению ЭВМ. Этот вопрос в той или иной форме затрагивается в течение всего курса, начиная с простых, но вполне «жизненных» задач из разделов IX класса и кончая вопросами, связанными с программным обеспечением ЭВМ. Мы убеждены, что отдельно рассматривать эту тему нет необходимости. Это показали и сочинения, написанные десятиклассниками (мы используем и такую форму работы, благо темы для сочинений по информатике в наши дни — буквально «под ногами»). Владение темой, кругозор и суждения учащихся были вполне удовлетворительными.

Вместо послесловия. Наш экспериментальный курс успешно применяется во многих таллинских школах. Опыт его использования показал, что практически любой учитель, даже без специальной подготовки, может преподавать этот предмет, используя наш «учебник». Курс легко усваивается учениками, позволяя снять с предмета ярлык «темной науки». Просьбы предоставить наш учебный материал идут со всех концов страны — из школ, учебных комбинатов, педагогических институтов и институтов усовершенствования учителей Калининграда, Тбилиси,

Йошкар-Олы, Ярославля, Красноярска, Караганды, Кронштадта.

В связи с этим мы оказались в довольно сложном положении. С одной стороны, мы рады бы помочь всем, кого сегодняшняя обстановка с предметом обрела на невыносимые условия работы. С другой — мы не можем и, строго говоря, не имеем права этого делать. Наш курс никем не утвержден и нигде не издается. У людей, от которых зависит и то и другое, мы не встречали ничего, кроме раздражения и равнодушия. Минпрос ЭССР с его лабораторией информатики и республиканский институт усовершенствования учителей с его кабинетом ОИВТ вот уже третий год работают практически вхолостую. Так что пока нам приходится заниматься, по сути дела, самодеятельностью, чтобы как-то удовлетворить просьбы людей.

В то же время результаты конкурса на учебник ОИВТ также не вселяют оптимизма. На наш взгляд, одна из главных причин его неудачи состоит в том, что уважаемые авторские коллективы (в частности, те, чьи работы упомянул журнал «Информатика и образование») очень далеки от школы. Отдельные специализированные учебные заведения Москвы, Новосибирска или других центров едва ли могут служить полигоном при создании массового учебника. По уровню подготовки учащихся, уровню преподавания, по технической базе, наконец, они не сравнимы с сегодняшней «средней» школой. Есть ли смысл определять общее направление на их основе? И не в стенах ли подобных школ родилась идея о переносе информатики в средние классы? Идея, которая уже воплощается в учебной программе и которая не может принести обычным школам ничего, кроме новых мучений! В самом деле, сейчас, преодолевая трудности, информатика все же оправдывает себя, поскольку готовит (пусть даже пока только морально) старшеклассника к встрече с вычислительной техникой в вузе или на производстве, где она внедряется уже сегодня. Но перейдя в средние классы, она станет просто балластом. Прекрасная идея ранней компьютерной гра-

мотности при нынешних темпах компьютеризации школ превращается в формальность и полностью дискредитируется. Не нужно спешить. Еще не скоро дисплей станет таким же символом нашей школы, как, скажем, классная доска. Информатика в старших классах — это самое разумное решение на сегодняшний день, да и на завтрашний тоже.

Работая втроем над курсом ОИВТ, мы обеими ногами стоим на «грешной земле», поскольку преподаем в обычных школах. Один из нас — учитель физики, он первым опробовал «на себе» материал курса. Двое остальных — инженеры-программисты, работающие в школах по совместительству. Один из них (а именно автор этой статьи, непосредственно писавший учебное пособие) совмещает две профессии с третьей — журналистикой. Такой состав, безусловно, сыграл свою роль, и сейчас мы имеем достаточно оснований считать свой труд не напрасным. Когда видишь улыбки и радостные лица учеников, входящих в класс на урок информатики, чувствуешь их интерес и живое участие, приходит уверенность в правильности выбранного пути.

Контрольная работа по курсу информатики

Условие. Ты — директор Таллинского универмага. Для использования компьютеров в работе Дома торговли выделено 150 тыс. рублей и 50 тыс. долларов. В пределах этой суммы можно производить закупки техники по данному прейскуранту.

Задание. 1. Выбрать конфигурацию вычислительных средств и обосновать выбор.

2. Указать, какие элементы программного обеспечения, по твоему мнению, необходимы.

3. Описать интерфейс покупателя с вычислительной системой (сценарий работы программы).

Общая сумма затраченных средств не должна превышать выделенную!

Конфигурация	Марка	Цена
Отечественные персональные компьютеры		
Дисплей текстовый	ДВК-2М	10 000 руб.
2 дисководов под гибкие	ЕС-1840	12 000 руб.
дискеты емкостью по 256К	ИСКРА-1030	10 000 руб.
ОЗУ 128К	Электроника-79	12 000 руб.
Импортные персональные компьютеры		
Дисплей графический, монохромный	ROBOT-RON 1715	5500 руб.
ОЗУ 256К	IBM PC	1400 долл.
2 флоппи-дисководов (256К диски)	ИЗОТ 1036	1200 долл.
Отечественные мини-ЭВМ		
4 текстовых дисплея	СМ-4	100 000 руб.
2 флоппи-дисководов (256К — диски)		
ОЗУ 256К		
4 дисководов емкостью по 5МБ	СМ-1420	100 000 руб.
4 текстовых дисплея		
2 флоппи-дисководов (256К — диски)		
ОЗУ 2МБ		
2 дисководов емкостью по 14МБ		
Примечание: Для мини-ЭВМ необходима еще и система кондиционирования воздуха стоимостью 30 000 руб.		

71

Таблица 2
Дополнительные внешние устройства

Устройства	Цены	
	отечественные (руб.)	импортные (долл.)
Гибкий (флоппи) диск	—	3
Дисковод для флоппи-диска	—	250
Дисплей графический, монохром	1000	160
Дисплей графический, цветной	—	400
Принтер	6500	500
Устройства для объединения ЭВМ в сеть	—	15 000

Цены в таблицах условные.

Пакет программ «LEARN ENGLISH»

Описываемый пакет состоит из пяти программ-тренажеров для учеников и одной программы, с которой работает учитель. Программа-тренажер оперирует со словами, которые выбираются случайным образом из определенного для данной программы набора и предъявляются ученику. Для каждого слова ученик должен выполнить задание.

Организация пакета предполагает: возможность смены данных, с которыми идет работа;

выработку адекватной формы ответа; наличие механизма настройки на индивидуальные возможности учащегося; подсказку и выработку стратегии ее применения;

обучающий и контролирующий режимы для разного вида работ;

возможность наглядного отражения хода выполнения работы;

наличие программы учителя для различных диспетчерских функций.

Выбор решения по каждому из этих положений определялся как учебно-педагогическими целями, так и имеющимся оборудованием.

Пакет «LEARN ENGLISH» реализован на Бейсике для ПЭВМ «Ямаха». Сеть состоит из машины учителя с дисководом и 15 ученических машин без дисковода. С машины ученика невозможен доступ к диску. Учитель может переслать на машину ученика только программу. Следовательно, данные нужно представить как элементы программы. Их можно задать в операторах DATA и «подклеивать» к программе перед рассылкой по сети.

Выбор формы ответа учащегося является существенным, если не решающим элементом учебных программ. Широкое распространение получают программы, в которых нужно выбрать один ответ из нескольких возможных. Но этот способ, как правило, малоэффективен, здесь просто вреден: постоянно предъявляя учащемуся варианты ошибочного написания слова, мы способствуем их запоминанию. Поэтому

в пакете принят другой способ ввода ответа: нужное слово должно быть набрано на клавиатуре. Иногда высказывается опасение, что работа будет затруднена плохим знанием клавиатуры. По этому поводу можно привести следующие цифры: количество слов, обрабатываемых в течение 5-минутного сеанса, возрастает от 10 до 30 за два месяца при работе раз в неделю. Дальнейший рост сдерживается возможностями не ученика, а программы, так как при обработке каждого ответа используются графические и музыкальные средства. Программа обрабатывает правильные и неправильные ответы: из правильных выделяет быстрые и медленные; из неправильных — ситуации «ответ неверен» и «не успел». Действия ученика оцениваются числом очков.

Настройка программы на индивидуальные возможности учащегося происходит по двум направлениям.

Во-первых, предусмотрено изменение частоты выбора слов, в которых допущены ошибки. Начальный набор состоит из N слов; до первой ошибки слова из него предъявляются с равной вероятностью $1/N$. После того как для некоторого слова ученик допускает ошибку, это слово «дописывается» к набору. В нем становится $N+1$ слово, а вероятность выбора ошибочного слова равна $2/(N+1)$, т. е. увеличивается практически вдвое. После второй ошибки в этом слове оно «дописывается» еще раз; в наборе теперь $N+2$ слова, а вероятность выбора ошибочного становится $3/(N+2)$, т. е. почти втрое больше первоначальной.

Второе направление, по которому ведется настройка, — скорость работы программы. Для ввода ответа ученику планируется определенное время, которое всегда пропорционально длине правильного ответа. Коэффициент пропорциональности меняется во время работы программы. Скорость работы программы регулируется отрезком времени, выделяемым для ответа.

В пакете предусмотрена работа ученика в одном из трех режимов. Первоначально ученик сам выбирает подходящий ему режим, определяя тем самым свою начальную скорость работы. При выборе первого, самого медленного режима предусмотрено дальнейшее замедление скорости после трех неправильных ответов подряд. После пяти правильных ответов подряд ученик переводится на следующий режим (уровень), при этом увеличивается скорость и меняются некоторые другие параметры программы. После первой же ошибки ученик возвращается на предыдущий уровень. Таким образом большую часть времени он работает в достаточно трудных для себя и оптимальных для процесса обучения условиях.

Программы-тренажеры допускают использование подсказки. В пакете подсказка организуется по запросу нажатием функциональной клавиши F1). После подсказки ученик должен сам ввести правильный ответ в машину.

В ряде программ предусмотрено получение дополнительной информации. Так, в программах «Неправильные глаголы» и «Имена прилагательные» можно запросить перевод предъявленного глагола или имени прилагательного. Это не является подсказкой, так как в этих программах преследуется другая цель: нужно ввести недостающую форму неправильного глагола или сравнительную и превосходную степень указанного прилагательного. Поэтому в отличие от подсказки такие запросы не наказываются снятием очков.

Для каждой программы введены два режима работы: обучающий и контролирующий. Все, что говорилось до сих пор о характере выбора слов и об использовании подсказки, относится к обучающему режиму работы. В контролирующем режиме слова выбираются случайно, но только по одному разу. Сеанс заканчивается либо через 5 мин, либо раньше, если выбраны все слова. Использование подсказки на данном этапе не допускается.

При работе с программами на экран выводится дополнительная информация: текущая сумма набранных учеником очков, оставшееся для ответа время,

уровень, на котором он работает.

Информация, позволяющая следить за тем, сколько времени осталось для ответа, представлена в графической форме. При предъявлении на экране очередного слова начинается движение по экрану некоторого объекта (корабля, мяча, зверя и т. п.); ответ должен поступить раньше, чем объект достигнет границы экрана. При переходе на другой режим работы меняется, например, тип корабля, вид зверя.

Необходимость настройки программ на слова конкретного урока, сбор и анализ результатов работы учащихся потребовали создания программы для учителя, выполняющей диспетчерские функции: формирование набора слов для урока, сбор результатов по сети, анализ результатов, демонстрация программ. Последняя функция предназначена для работы с пакетом на этапе знакомства и по существу представляет собой меню для запуска программ пакета с некоторым демонстрационным набором слов.

Для работы на уроке учитель должен позаботиться о нужном наборе слов. Их можно выбрать из подготовленных на диске данных, пользуясь функцией «Формирование набора слов для урока». Данные для программы «Перевод слов» включают в себя словарный запас учебников под редакцией Уайзера для V—X классов и Старкова для IV—VIII классов. Слова разбиты по урокам. Можно выбрать слова для одного или нескольких уроков, часть слов одного урока. Выбранные слова можно «подклеить» к программе и разослать по сети немедленно, а можно сохранить на диске, если подготовка ведется не непосредственно перед уроком.

В конце пятиминутного сеанса работы с тренажером ученику сообщаются результаты работы: количество введенных слов и процент правильных ответов. Эту информацию можно сохранить на диске для дальнейшего анализа. Для этого ученик должен положительно ответить на вопрос: «Нужно сохранить результаты?» При этом информация о сеансе (имя программы, фамилия ученика, количество слов, процент правильных ответов, описание набора слов) по-

сылается в сеть; программа переводится в состояние ожидания до получения с машины учителя подтверждения об успешной пересылке. Прием результатов, запись их на диск и вывод сообщения об успешной пересылке выполняет функция «Сбор результатов по сети» программы учителя. Поскольку при работе с сетью нередко возникают ошибки, предусмотрена реакция на ошибки и возможность повторной пересылки. Пользуясь функцией «Анализ результатов», учитель может выбрать и просмотреть на экране результаты по каждой из программ, как общие, так и относящиеся только к одному ученику, к одной дате, к одному уроку.

Возможен выбор результатов в любой комбинации этих признаков, например можно просмотреть результаты одного ученика по одному уроку; они будут отражать динамику усвоения учеником новых слов. Можно просмотреть результаты одного урока за один день (например, если в этот день проводился контрольный урок). По желанию учите-

ля просмотренные результаты могут быть распечатаны.

Программы пакета использовались на уроках английского языка в Пушкинской экспериментальной школе (ПЭСШ) и средней школе № 3 г. Пушкино. Кроме того, в течение 1986/87 учебного года пакет регулярно использовался на занятиях кружков по информатике в VI—VIII классах ПЭСШ.

Автор выражает благодарность А. И. Хибнику за предложения по использованию разных режимов работы и настройке программ, а также А. Ю. Огурцову за реализацию быстрой пересылки по сети.

Программы пакета можно получить по адресу: 142292, г. Пушкино Московской области, НИВЦ АН СССР, Фонд алгоритмов и программ. Подробное описание пакета содержится в работе [1].

Литература

1. Лунина Н. Л. Пакет учебных программ-тренажеров по английскому языку: Материалы по математическому обеспечению ЭВМ: Пушкино: ОНТИ НЦБИ АН СССР, 1987.

И. ТОКАРСКАЯ,

канд. пед. наук, Узбекский республиканский педагогический институт русского языка и литературы

Информатика и повышение квалификации

Слушатели факультетов повышения квалификации должны овладевать новыми, современными знаниями, относящимися к их трудовой деятельности. Необходимо им и знакомство с компьютерной техникой.

На ФПК в Узбекском педагогическом институте русского языка и литературы с 1985 г. читается курс «Современная вычислительная техника и ее применение». Задачи курса предусматривают ознакомление слушателей с функциональными характеристиками ЭВМ, особенностями общения человека и машины, с составлением лингвистических алгоритмов для автоматизированной обучающей системы, с использованием компьютеров в преподавании языковедческих дисциплин. Следует отметить, что знакомство с этими вопросами важно не только в плане достижения всеобщей компьютерной грамотности, но и с

чисто практической точки зрения: в вузах и училищах Узбекистана уже имеется немало мини-ЭВМ, для работы на них нужно составлять обучающие программы и по русскому языку.

Чтение курса по информатике и вычислительной технике связано с целым рядом трудностей: новизной материала, необходимостью в предельно сжатые сроки (26 ч) дать слушателям представление о составлении алгоритмов и их записи, познакомить с работой компьютера, подготовкой обучающих программ и многим другим.

Наряду с этими общими трудностями (их испытывает практически каждый, взявшийся за чтение нового курса) имеются и специфические, обусловленные составом слушателей: преподавателей русского языка и литературы педвузов и педучилищ, получивших фило-

логическое образование и ведущих преподавание лингвистических и литературоведческих дисциплин, порядком подзабывших математику, пугают встречи с вычислительной техникой. Негативную роль сыграл и первый, неудачный опыт начавшегося повсеместно компьютерного всеобуча: собранным в одну аудиторию биологам, лингвистам, географам и других специалистов, читали курс «Информатика и вычислительная техника» с математическими выкладками, графиками, формулами. Такие занятия, без учета конкретной специальности обучающихся приводили слушателей к мысли, что вычислительная техника им не под силу, а главное — не может быть применена как средство обучения.

Преодоление психологического барьера между слушателями и лектором представляло нам большую трудность в работе. Разрядка наступала лишь в конце курса, когда слушатели знакомы с составленной нами обучающей программой («Гласные в приставках пре- и при-»), выполняли ее на компьютере «Ямаха», т. е. получали возможность конкретно представить себе учебную ситуацию: вычислительная техника на занятии по русскому языку.

Все это наводило на мысль изменить порядок изучения материала: первое же занятие начинать с выполнения на компьютере обучающей программы по специальности. Приведем начало составленной нами обучающей программы, которая рассчитана на работу со студентами национальных групп педучилищ или первых курсов пединститутов, а также с учащимися IX классов национальной школы с углубленным изучением русского языка. Цель ее — закрепить у учащихся навыки правописания гласных в приставках *пре-* и *при-* в процессе выполнения самостоятельной работы на компьютере.

На занятии в дисплейном классе «Ямаха» слушатели довольно быстро, в течение 15—20 мин, освоились с ролью пользователей ЭВМ — большинство из них знакомо с клавиатурой и техникой работы на пишущей машинке, да и выполнять приходилось самые простые операции: нажимать на клавиши с буквами Е или И в паре с приставками *пре-* или *при-*, печатать краткие ответы Да или Нет и один раз — свое имя.

Это была первая встреча преподавателей с ЭВМ.

Им нравилось все: возможность вести диалог с машиной, ее вежливое обращение к каждому по имени, а главное — ответные реплики-разъяснения и реплики-поощрения, высвечивавшиеся на экране дисплея, разнообразный характер в зависимости от ответа и трудности примера, уважительный тон и др.

Все было ново, вызывало живой интерес. И здесь же, в дисплейном классе, было задано много вопросов: «Как составляется программа?», «Как программа вводится в ЭВМ?», «Сколько программ может запомнить машина?» и многое-многое другое.

Психологический барьер был сломлен. Материал всех последующих занятий воспринимался как ответы лектора на интересующие слушателей вопросы. Особое внимание привлекали темы, непосредственно связанные с практической деятельностью преподавателей-русистов: «Лингвистический алгоритм»* и «Обучающая программа по русскому языку». Кратко остановимся на рассмотрении этих двух тем, что позволит осветить некоторые вопросы применения вычислительной техники в преподавании русского языка.

Тема: Гласные в приставках *пре-* и *при-*. Внимательно прочитайте правило о правописании гласных в приставках *пре-* и *при-*.

75

В русском языке есть две приставки — *пре-* и *при-*, в которых гласные в безударном положении произносятся одинаково. Чтобы правильно писать эти приставки, нужно знать их значение.

Рассмотрите таблицу «Значение приставок *пре-* и *при-*»

Слова с приставкой	Значение приставки
<i>при-</i> прицепить вагон прибежать к реке пришкольный участок приоткрыть дверь	присоединение приближение близость
Слова с приставкой <i>пре-</i> предобрый человек прекислое яблоко преувеличивать преграда	неполнота действия Значение приставки сходно со значением слова «очень» сходно со значением приставки пере- (перегородить)

— Хотите поупражняться в написании слов с приставками *пре-* и *при-*?

— Да.	— Нет.
— Как ваше имя?	— Очень жаль! Хотелось бы с Вами поработать. Выключите, пожалуйста, машину
— Лола.	
— Очень приятно.	

Лола, давайте выполним упражнение.
Порядок работы:

* Термин «лингвистический алгоритм» в статье используется как алгоритм решения орфографических и пунктуационных задач.

- 1) прочитать предложение,
- 2) найти слово с приставкой *пре-* или *при-*,
- 3) определить значе- ние приставки в слове,
- 4) вставить вместо пропуска гласную «е» или «и» (нажмите клавишу с буквой «е» или «и»).

Желаю тебе, Лола, успеха!

I. Не меняя позы, медленно, Алексей приоткрыл глаза (Б. Полевой).

— И

— Хорошо начали! Правильно: Алексей не вполне открыл глаза, только приоткрыл. Переходите ко второму заданию.

— Е

— Вы допустили ошибку: Алексей не- много открыл глаза, дей- ствие было неполным. Следовало вставить гласную И. Вдумайтесь в содержание!

76

II. Поезд пр бывал (Ю. Яковлев).

— И

— Вы ответили пра- вильно: поезд прибы- вал — значит, прибли- жался

— Е

— Ваш ответ невер- ный! А пример нетруд- ный, следовало внима- тельно прочитать пред- ложение. В слове «при- бывал» приставка имеет значение «приближе- ние», нужно было вста- вить гласную И.

III. Артиллеристы, как и всегда, вели себя пр - восходно (Л. Толстой).

— Е

— Вы внимательно разобрались в этом при- мере: приставка имеет значение «очень».

— И

— Ошибка! Работай- те внимательнее, аду- мывайтесь в содер- жание:

артиллеристы вели себя очень хорошо, превос- ходно. Следовало вста- вить гласную Е.

Лингвистический алгоритм.

В преподавании русского языка алго- ритм — определенный порядок рассужде- ний для решения орфографических и пунк- туационных задач — используется давно. Так, в конце XIX в. в работе известного методиста Д. И. Тихомирова «Чему и как учить на уроках родного языка в начальной школе?» (1890) имеются очень удачные алго- ритмические построения рассуждений о пра- вильности безударных гласных в корне слова и о других орфографических правилах. Воп- росы использования алгоритмов в системе преподавания русского языка освещались в работах Г. Г. Граник (1964), А. И. Власенко- ва (1965), Л. П. Федоренко (1984) и других. Определенный интерес представляет пособие «Использование алгоритмов на уроках рус-



ского языка в азербайджанской школе (ме- тодические рекомендации)» (1984), в кото- ром представлено около 30 алгоритмов и ме- тодические рекомендации по их использо- ванию.

Таким образом, слушателям ФПК алго- ритм известен как дидактический прием. На занятиях работа с алгоритмами велась в но- вом аспекте:

алгоритм как основное звено в подготовке обучающей программы;

запись алгоритма на алгоритмическом языке;

типы алгоритмов (линейные, разветвля- ющиеся, циклические).

Разветвляющийся тип алгоритма — самый распространенный лингвистический алгоритм (линейные и циклические применяются ред- ко). Слушатели ФПК легко справлялись с рассуждениями на базе орфографических и пунктуационных правил, но испытывали за- труднения в построении алгоритмов и их

записи на алгоритмическом языке. Приходилось вначале составлять и записывать план рассуждения, разбирать записи готовых алгоритмов. Большую помощь в этой работе оказывали подготовленные нами кодопозитивы, позволяющие проецировать на экран готовые алгоритмы. Работа над алгоритмами проводилась с постепенным нарастанием трудности, но начинали с самых простых типа:

алг правописание приставок «раз(с)-» и «роз(с)-»

нач

если ударение не падает на приставку
то писать приставку «раз(с)-»
иначе писать приставку «роз(с)-»
все

кон

На практических занятиях слушатели работали самостоятельно: выбирали в учебнике по русскому языку орфографическое или пунктуационное правило, изображали его вначале в виде привычной схемы (1), а затем на ее основе составляли алгоритм, записывали его на алгоритмическом языке (2), изображали в виде блок-схемы (3). В качестве примера можно привести работу над правилом о правописании суффиксов имен существительных -ец и -иц-.

1.

СУФФИКСЫ -ЕЦ, -ИЦ-

1. Мужской ли это род? ----- Пиши -ЕЦ
2. Женский ли это род? ----- Пиши -ИЦ(а)
3. Средний род:
падает ли ударение на окончание?

	ДА
НЕТ	
┌──────────────────────────┐ └──────────────────────────┘	
Пиши -ИЦ(е)	Пиши -ЕЦ(о)

2.

алг правописание суффиксов «-ЕЦ» и «-ИЦ-»

нач

если имя существительное мужского
рода
то писать суффикс «-ЕЦ»
иначе
если имя существительное женского
рода
то писать суффикс «-ИЦ(а)»
иначе
если имя существительное среднего
рода с ударными окончаниями
то писать суффикс «-ЕЦ(о)»
иначе
если существительное сред-

него рода с безударным окончанием

то писать суффикс «-ИЦ(е)»

все

все

все

все

кон

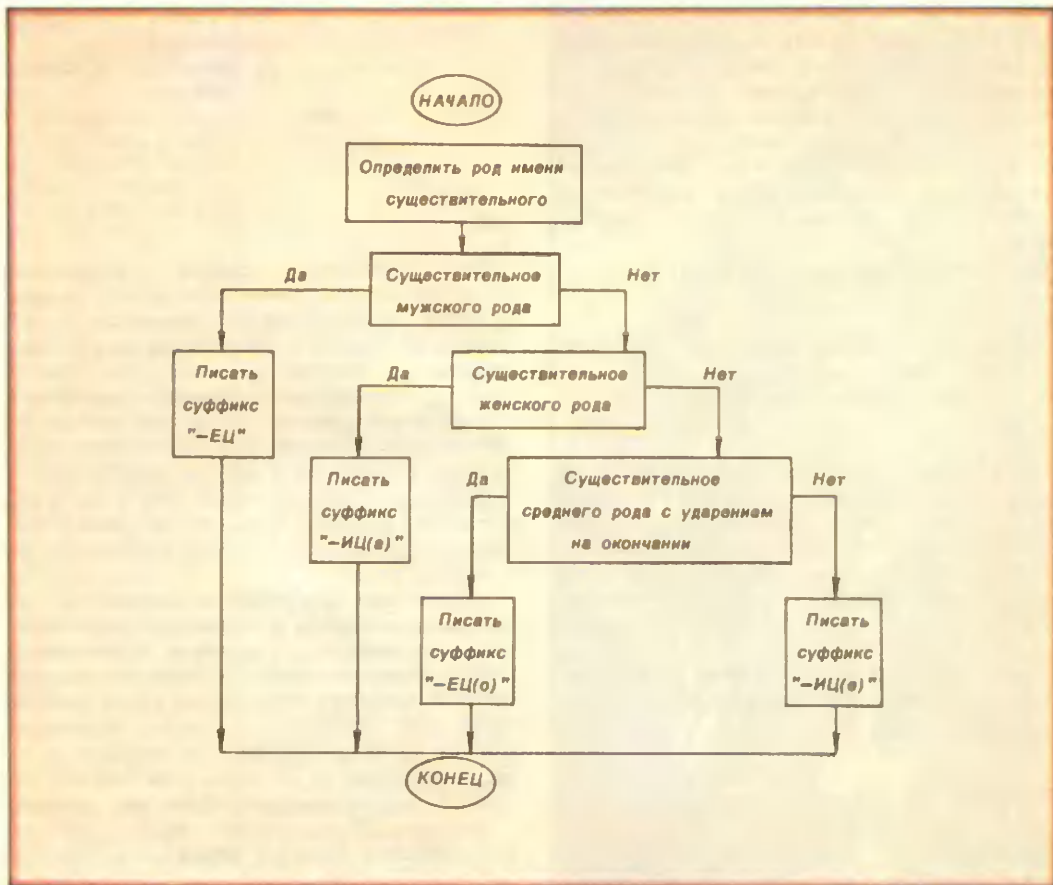
3.

Этот материал (схемы, алгоритмы, блок-схемы) аккуратно записывался на прозрачной пленке, демонстрировался с помощью кодоскопа и обсуждался присутствующими на занятии — происходила своеобразная защита подготовленного материала.

Работа эта увлекла слушателей, воспринималась как подготовка к составлению обучающей программы и как создание дидактического материала, который может быть использован без ЭВМ. Силами слушателей подготовлен небольшой сборник алгоритмов по русскому языку.

Обучающая программа — центральная тема курса, имеющая и теоретическое, и практическое значение: в вузах и педучилищах Узбекистана уже имеется немало компьютеров, а программ по русскому языку практически нет. Нужны составители программ. Лекция (2 ч) и практические занятия (6 ч) были нацелены на решение этой важной задачи. В центре внимания были два вопроса: значение использования ЭВМ в системе преподавания русского языка;





составление обучающих программ по русскому языку для студентов и учащихся национальных групп.

Раскрыть роль ЭВМ в интенсификации преподавания было необходимо — только убежденные преподаватели могут взяться за такое новое и трудное дело, как составление обучающих программ.

Практическая работа в дисплейном классе, как уже описывалось в статье, произвела на слушателей большое впечатление. Это надо было подкрепить теоретически, показать, что работа с компьютером создает условия для реализации резервов, интенсификации преподавания русского языка, дает возможность:

усилить индивидуализацию в процессе обучения (при выполнении программы, фрагменты которой приводились в настоящей статье, хорошо подготовленным ученикам можно предложить вариант с более краткими комментариями или без комментариев при правильном ответе, а слабоуспевающим — облегченные примеры);

увеличить объем коммуникативных упражнений за счет сокращения с помощью компьютера затраты времени на другие виды работы;

предъявлять различного рода информацию на дисплее;

интенсифицировать выполнение упражнений, сделать их более результативными: психологами давно отмечено, что в процессе работы над заданием ученик должен немедленно получать информацию (подкрепление) о правильности или ошибочности принятого им решения. Это является одной из важных предпосылок формирования прочных навыков, в том числе и языковых. Компьютер моментально реагирует на вводимый ответ, что и способствует результативности тренировочной работы.

На практических занятиях отмечались общие требования к обучающим программам: соответствие учебной программе и учебнику, обеспечение условий для самостоятельного выполнения заданий, доступность материала, учет возрастных особенностей учащихся и др.

Основное же внимание было сосредоточено на лингводидактическом обосновании, определении лингводидактических требований к каждому этапу работы.

Выбор темы — главный этап в подготовке обучающей программы. Необходимо исходить из таких факторов, как значимость отбираемого материала для обучения русскому языку учащихса национальной школы, национальных групп вузов, частотность его использования в устной и письменной речи, трудность усвоения материала, возможность использовать полученные знания, сформированные навыки и умения в реализации коммуникативных задач.

Отбор материала для обучающей программы должен проводиться с учетом специфики усвоения русского языка национальных школ, во взаимосвязи с их родным языком и уровнем языковой подготовки.

Составление комментариев, «реплик» компьютера, оценивающих результаты выполнения каждого задания, потребует от автора программ немалога напряжения и даже творческой изобретательности. Реплики-подкрепления должны быть понятными, построенными на знакомой лексике, небольшими по объему, выразительными, содержащими умеренную похвалу и порицания.

Написание сценария обучающей программы, компоновка всего материала также требуют большого внимания: задания должны быть расположены в строгом соответствии с логикой учебного материала, с постепенным нарастанием трудности, таблицы и схемы — компактными, с выделенными значимыми элементами.

Запись на языке программирования, экспериментальная проверка — два очень важных этапа подготовки обучающей программы — рассматривались в самом общем, ознакомительном плане из-за недостатка времени.

Аудиторные занятия по составлению обучающих программ проводились очень уплот-

ненно. Слушатели приносили, в соответствии с выбранной темой, дополнительный материал и самостоятельно по этапам отработывали столь новый и необычный тип пособия, получали в нужный момент консультации преподавателя. В результате были составлены фрагменты программ, а отдельными преподавателями — завершенные сценарии. К числу последних следует отнести: «Использование мини-ЭВМ в организации самостоятельной работы студентов в усвоении лексики по специальности» Л. Россинской, «Род имен прилагательных» Н. Паршкова, «Глагол. Категория залога» М. Стручалина и другие.

Сейчас с уверенностью можно сказать, что курс «Современная вычислительная техника и ее применение» играет значительную роль в деле повышения квалификации преподавателей-русистов.

Литература

Проект ЦК КПСС. Основные направления перестройки высшего и среднего специального образования в стране // Правда. 1986. 1 июня.

Власенков А. И. Материалы к исследованию по теме «Использование алгоритмов в обучении орфографии». М.: Просвещение, 1965. 95 с.

Граник Г. Г. Приемы работы по различению частиц НЕ и НИ // Русский язык в школе. 1964. № 1. С. 51—55.

Звенигородский Г. А. Первые уроки программирования. М.: Наука, 1985. С. 16.

Использование алгоритмов на уроках русского языка в азербайджанской школе (методические рекомендации). Баку: Минпрос АзССР, 1984. 20 с.

Садчикова П. В. Вычислительная техника и ее применение: Для ФПК учителей русского языка и начальных классов национальной школы при пединститутах и университетах. М.: Минпрос СССР, 1986. С. 13.

Тихомиров Д. И. Чему и как учить на уроках родного языка в начальной школе? М., 1890. с. 327.

Федоренко Л. П. Закономерности усвоения родной речи. М.: Просвещение, 1984. С. 123—124.

Карманная память

Быстродействующая, надежная внешняя память небольших размеров разработана фирмой ИТТ Саппо. Это устройство на интегральных схемах размером не более стандартной кредитной карточки в отличие от флоппи-дисков не боится влаги, пыли,

легко переносит механические нагрузки и допускает не менее 10 тыс. циклов использования. В том же плоском корпусе находится схема управления, поз-

воляющая с высокой скоростью записывать и находить необходимые данные.

Сфера применения такой памяти практически не ограничена: от лингвистических процессоров до систем управления реального времени.

ЧТО?
МОЖЕТ
ЭВМ

О компьютерном всеобуче в Чечено-Ингушской АССР

80

Первые попытки использовать электронную вычислительную технику в школах ЧИАССР относятся к середине 70-х гг. В средних школах № 1, 2, 3 г. Грозного учителями-интузиастами на кружковых и факультативных занятиях, на уроках математики, физики и химии применялись микрокалькуляторы. Их использование носило хаотичный характер, так как отсутствовала продуманная методика, да и сами МК были разнотипными, поэтому методистом кабинета физики республиканского ИУУ В. А. Кудерским была организована проблемная группа для тех учителей, которые хотели бы применять МК на своих уроках. Опыт работы этой группы был обобщен, и в 1983 г. вышла брошюра «Работа с микрокалькулятором „Электроника МК-18А“». В ней излагались методические приемы освоения клавиатуры, составления и реализации на МК вычислительных алгоритмов и, что особенно важно, освещался опыт использования МК при изучении конкретных тем по математике и физике.

В 1982 г. вышло методическое письмо МП СССР «Об использовании микрокалькуляторов в учебном процессе», однако практика их применения в учебном процессе в школах республики не изменилась. Даже после принятия документов о реформе общеобразовательной и профессиональной школы, когда задача компьютеризации была сформулирована как одна из главных, автору этих строк пришлось вступать в острые, хотя и доброжелательные, дебаты по поводу целесообразности использования в школе вычислительной техники с директором республиканского ИУУ.

Видимо, результатом подобных взглядов руководства можно объяснить тот факт, что поступавшие в 1982—1984 гг. в базисные магазины МК не находили потребителя, залеживались. Резкое усиление пропаганды ВТ в печати с конца 1984 г. способствовало тому, что в настоящее время около 50 % из более 500 школ республики имеют в школах МК типа МКШ-2. Но нет ни одной школы, оснащенной программируемыми микрокалькуляторами, а практика использования имеющихся оставляет желать лучшего, частью из-за технических характеристик.

Преподавание ОИВТ во всех школах республики началось в безмашинном варианте. Только в 1986/87 учебном году учащиеся школы № 3, 21 г. Грозного получили возмож-

ность работы на машинах, в школе № 21 по инициативе руководителей вычислительного центра предприятия «Грознефтеоргсинтез». Программа, составленная В. А. Ковалевым, состоит из двух частей — для IX и X классов. Для каждого года обучения программой предусмотрено 35 ч теоретических и 105 ч практических занятий. В первый год изучается подготовка информации на машинных носителях, во второй — работа с ЭВМ. Программа ориентирована на технологию обработки информации и на технические средства КИВЦ «Грознефтеоргсинтез». Вместе с аттестатом учащиеся получают и специальность оператора ЭВМ.

В сравнительно молодом Чечено-Ингушском госпединституте создан дисплейный класс, оборудованный КУВТ «Ямаха», кабинет для обучения работе на МК. В ноябре 1986 г. образована кафедра «Информатика и вычислительная техника». С участием специалистов ЧИГПИ совместно с республиканским ИУУ дважды проводилась подготовка учителей. В пединституте проводится работа по организации и проведению экскурсий в дисплейном классе для учащихся школ и ПТУ республики.

Организируются экскурсии и в других вузах республики — в Чечено-Ингушском госуниверситете и Грозненском нефтяном институте, а также на ВЦ ведущих предприятий НПО «Промавтоматика», завод «Красный молот» и некоторых других.

Однако нельзя сказать, что сегодня во всех школах полностью выполняется практическая часть программы для безмашинного варианта. Не говоря уже о том, что не все экскурсии достигают целей, не всегда их удается провести из-за отношения руководителей ВЦ. Особенно болезненно это отражается на учащихся сельских школ.

С помощью НПО «Промавтоматика» в СШ № 9 г. Грозного в 1987 г. оборудован дисплейный класс КУВТ-86, оснащается ПЭВМ «Агат», ДВК-1, ДВК-2, созданный в Ленинском районе специализированный УПК.

Поступившие в 1987 г. два класса КУВТ-86 Минпрос ЧИАССР распределил в УПК Старопромысловского района и среднюю школу № 5 с. Урус-Мартан. Таким образом, в настоящее время есть одна городская и одна сельская школа, обеспеченные компьютерными классами. Это позволяет уже сейчас про-

водить исследовательскую работу по учету особенностей сельской и городской школ.

Необходимо отметить, что учителя, не дожидаясь, пока централизованно будет поставлено программное обеспечение, сами начали составлять и использовать программы как на уроках информатики, так и на других уроках. Важно и то, что при составлении программ они используют собственные методические приемы.

Имеющаяся в школах техника нужна и самой школе. На диски заносятся данные об учащихся, учителях, составляются бланки, справки, экзаменационные протоколы и т. д., что позволяет частично автоматизировать рутинную работу и высвободить время учителя, а также совершенствовать информационную технологию школьной жизни. С 1987 г. для участников республиканских олимпиад проводится практический тур, на котором выполнение заданий требует использования ВТ.

В школах и УПК, где уже имеются дисплейные классы, налажено техническое обслуживание на базе местных предприятий г. Грозного. Однако пока это дорогое удовольствие (75 рублей в месяц). Не всегда у мастеров по техническому обслуживанию имеются детали для замены. Предприятие в 1988 г. приобрело дисплейный класс «Корвет», который планируется использовать для создания в Грозном республиканского центра по информатике и вычислительной технике.

На базе республиканского НТО на общественных началах создан Комитет по информатике и вычислительной технике, который проводит работу по пропаганде знаний о компьютерной грамотности, организует семинары и конференции. В его состав вошли ведущие специалисты по ВТ вузов и предприятий республики. Был проведен семинар по проблемам компьютерной грамотности для руководящего состава обкома КПСС (а в настоящее время это немаловажно). В плане мероприятий комитета обобщение опыта при-

менения компьютерной техники, в том числе и в школе. С этой целью организован сектор по внедрению компьютерной техники в учебный процесс. Однако проводимая комитетом работа по координации практических усилий, созданию определенной базы, в том числе и методической, для повсеместного перехода к использованию компьютерной техники еще недостаточно эффективна.

Нынешнее положение в сфере компьютерной грамотности нельзя считать удовлетворительным. Как и везде, остро ощущается нехватка персональных компьютеров, а качество имеющихся оставляет желать лучшего. Но опасно другое. Ряд исполкомов, например Урус-Мартановский, не планирует и в обозримой перспективе выделение средств для эффективного решения задач компьютерной грамотности. Настойчивые просьбы включить в план социально-экономического развития района в двенадцатой пятилетке строительство здания для специализированного УПК исполком райсовета оставил без внимания, хотя председатель райисполкома и оказал неоценимую помощь в открытии первого дисплейного класса в районе.

В заключение считаю необходимым высказать свое мнение по одному аспекту компьютеризации в настоящий период. Представляется нецелесообразным в условиях действующих жестких санитарно-гигиенических норм работы учащихся за дисплеем оснащение всех школ дисплейными классами. Техника в настоящее время еще слишком дорогая да и оплата за техобслуживание существенная, а если при этом учесть фактическое время работы дисплейного класса с его нынешними техническими характеристиками, то экономическая неэффективность такой стратегии очевидна.

Экономически выгодным представляется создание передвижных дисплейных классов, которые могут сыграть большую роль для обеспечения доступа большого количества учащихся к ВТ с меньшими затратами.

Н. ПАХОМОВА, Б. ШРЕЙБЕР

База данных школы

Во многих публикациях, посвященных компьютеризации школьного образования, важное место занимает проблема информационного обслуживания школ с помощью ЭВМ. Рассмотрим возможные пути ее решения, исходя из опыта работы с учащимися школы № 444 Москвы.

Работы по сбору, хранению, обновлению, накоплению разнообразной информации об учащихся, их семьях, о проведении учебно-воспитательных мероприятий и т. п. составляют существенную и трудоемкую часть школьного производственного процесса. В настоящее время эта работа выполняется вручную работниками школы. Несмотря на то что школьная реформа призвана упразднить формализм, процентоманию, упростить отчетность, снизить бумажный поток, обработка и хранение информации всегда будет иметь место.

Сведения об учащихся. Едва ли педагог признается, что он счастлив от того, что каждый учебный год ему приходится начинать с составления списка своего класса в 5—6 экземплярах (без единого исправления) с указанием данных о каждом учащемся и его родителях и отвечать за достоверность информации в этих списках в течение всего учебного года.

Ежечетвертная отчетность о количестве проведенных уроков, использованных ТСО: цифры, проценты, бумажные простыни информации!

Бесплатные учебники: учет их выдачи и сдачи, оценка состояния на начало и конец учебного года.

Денежные расчеты за горячее питание школьников: сбор денег, учет количества питавшихся ежедневно, учет колебаний ежедневной стоимости завтраков, выведение ито-

говых цифр по возврату неизрасходованных денег родителям — весьма трудоемкая работа, требующая более тысячи операций (а при ручном исполнении продлевается 2—3 раза). Она отнимает много времени и сил и не оплачивается.

Если говорить об объеме работы с разного рода информацией директора и завуча школы, перечисление займет не одну страницу.

Таким образом, в школе производится большое количество информационных операций, автоматизация которых высвободила бы силы и время педагогов для педагогической деятельности, позволила бы администрации иметь оперативный документ для принятия решений.

Если подняться на ступеньку выше по иерархической лестнице управления школами, то обнаруживается, что количество информационных операций возрастает так же, как и требование к достоверности информации и ответственности за принятие решений на ее основе.

На промышленных предприятиях задачи информационного обслуживания успешно решаются с помощью системы управления базами данных (СУБД). Уже разработано довольно много СУБД, что позволяет выбрать подходящую с учетом специфики обслуживаемого предприятия [1].

При выборе типа СУБД нужно учитывать воспитательные и образовательные цели, которые ставит перед собой школа. Уборку кабинетов, мелкий и средний ремонт мебели, школьного здания, библиотечного фонда стараются сделать с привлечением ребячьих рук для того, чтобы научить их работать, привить чувство хозяина. С тех же позиций можно настаивать на подобном самообслуживании и в информационном обеспечении школы с

помощью ЭВМ. А значит, при выборе СУБД нужно исходить из того, что работать с ней будут школьники, получившие основы знаний по предмету ОИВТ. Пролодав ряд практических учебных работ на ЭВМ, учащиеся, проявившие интерес к предмету, смогут расширить свои знания на факультативе, где познакомятся с СУБД и ее использованием, примут участие в решении настоящей производственной задачи — задачи ведения базы данных школы (БДШ). Опыт школы № 444 Москвы, где преподавание программирования введено впервые в стране в 1959 г., показывает, что учащиеся VIII и даже VII классов легко осваивают работу с СУБД реляционного типа.

Реляционный подход в решении информационных задач заключается в следующем. Хранимые данные представляются в виде отношений, каждое из которых можно рассматривать как таблицу. Так же как в теории множеств определены операции над множествами (пересечение, объединение и др.), так и в теории реляционных систем существуют операции над отношениями (таблицами): выборка, проекция, соединение.

Как показывает наш опыт, реляционные операции достаточно просты и доступны пониманию школьника. Решение отдельной информационной задачи сводится к написанию весьма простой инструкции (программы) на языке реляционной алгебры. При работе с СУБД УНИСОН (Универсальной информационной системы оперативного назначения) [2] инструкции выглядят как предложения на русском языке, поскольку язык взаимодействия с УНИСОН является подмножеством русского языка. Это и определило наш выбор.

С большим интересом, чувством ответственности и сознанием важности порученного дела школьники изучали элементы реляционной алгебры и средства реляционной СУБД УНИСОН, выполнили работы по созданию основы БДШ, решили ряд информационных задач школы под руководством сотрудников ВЦ базового предприятия. УНИСОН реализована на ЕС ЭВМ, может использоваться в пакетном и интерактивном режимах. В настоящее время СУБД реляционного типа имеются и на СМ ЭВМ, и на ПЭВМ.

Итак, что такое база данных школы?

Почти все информационные операции в школе тем или иным образом связаны со списком учащихся. Такой список с разбивкой по классам, как уже говорилось, составляется на начало учебного года в нескольких экземплярах и содержит разнообразную информацию об учащихся: фамилию, имя, отчество, дату рождения, пол, домашний адрес,

телефон, данные о родителях. Это первичные утверждаемые дирекцией списки, на основе которых затем школьный врач, библиотекарь и другие составляют свои вторичные списки, добавляя информацию об учащихся, такую, как «группа здоровья», диагноз или название выданных учебников с баллами оценок их состояния и пр.

Следовательно, БДШ содержит информацию, которую можно при необходимости наращивать в зависимости от различных информационных задач. Например, в БДШ может помещаться информация о текущей и итоговой успеваемости учащихся, о выполняемой общественной работе, об интересах и внешкольных занятиях, о профессиональной склонности и профпригодности.

В БДШ может находиться информация о кадрах педагогического и технического состава школы. На ее основе можно решать информационные задачи по учету кадров, распределению учебной нагрузки преподавателей и составлению расписания уроков и факультативных занятий, распределению учебных групп по кабинетам и др.

БДШ может быть использована при создании АОС, так как индивидуальный подход в обучении учащихся может осуществляться лишь при наличии информации об успехах каждого в процессе обучения.

Задача по ведению БДШ прежде всего заключается в поддержании достоверности информации о списочном составе школьников. Такая работа с успехом выполняется активом учащихся под руководством преподавателя — администратора БДШ.

Информация обязательно кодируется, доступ к ней ограничен и строго контролируется администратором БДШ. Эти и ряд других мер полностью исключают возможность бесконтрольного распространения сведений об учащихся.

Следует заметить, что в средних школах ряда стран уже введены автоматизированные информационно-управляющие системы, с помощью которых решаются задачи, подобные описанным выше. В этих системах также уделяется большое внимание вопросам защиты данных. В США данные об учащихся стали вторым — после данных о кредитных операциях — объектом правового регулирования.

Успешная работа с СУБД и БДШ активизирует творческую инициативу учащихся. Например, ими предложена и осуществляется идея создания банка задач. В память ЭВМ вводятся тексты задач по математике и физике с указанием тематики и степени сложности. Банк облегчит проблему подбора задач по заданной тематике или по степени трудности. На очереди еще не одна интересная

работа, сознание важности и необходимости которой делает мальчишек и девочек взрослей, серьезней и собранней, учит видеть и решать задачи реальной действительности, смело применяя школьный багаж знаний.

Наш опыт показывает, что привлечение учащихся к созданию и ведению базы данных школы с помощью современных средств обработки данных, к решению школьных информационных задач — не только реальное и нужное дело, но и необходимое звено компьютеризации школы.

Авторы охотно откликнутся на любое предложение об обмене опытом подобной работы, ответят на возникшие при прочтении статьи

вопросы, примут с благодарностью критические замечания.

Адрес для справок: 105504, Москва, ул. Нижняя Первомайская, д. 14, школа № 444.

Л и т е р а т у р а

1. Системы управления базами данных для ЕС ЭВМ: Справочник / Под ред. В. М. Савинкова. М.: Финансы и статистика, 1984.

2. Шрейбер Б. Т. Теоретические и практические вопросы обеспечения информационного комфорта в человеко-машинной системе // Вопросы информационной теории и практики. 1981. № 45. С. 66—79.

В. КАТКОВ

Студенческие олимпиады

84

В течение прошедших пяти лет Белорусский государственный университет им. В. И. Ленина был базовым вузом по проведению заключительного тура всесоюзных студенческих олимпиад по ЭВМ и программированию. Цели олимпиад состояли «в повышении качества изучения студентами фундаментальных наук и специальных дисциплин, развитии у них навыков самостоятельной работы и потребности к творческому овладению знаниями».

В олимпиадах принимали участие сборные команды союзных республик, Москвы и Ленинграда. Задачи готовились представителями команд и после тщательного обсуждения отбирались жюри в день проведения очередного соревнования.

Соревнования заключительного тура проходили обычно в течение нескольких дней и состояли в решении 3—4 задач, из которых не менее двух задач пропускатся на машине для получения числового результата.

Обычно предлагались задачи следующих типов.

Алгоритмы и программы. Требовалось разработать алгоритм и программы на языке высокого уровня (Алгол-60, Алгол-68, Фортран, ПЛ/1 или Паскаль) для решения задачи, опирающейся на знания студентов по структурам данных, алгоритмам сортировки и поиска информации, организации рекурсивных программ, построению эффективных алгоритмов.

Задача оценивалась пятью баллами; дополнительные баллы (не более двух) назначались за оригинальный, эффективный алгоритм и минимальное число запусков про-

граммы для отладки. Ошибки перфорации не учитывались.

Тестирование. Цель задания — выявить у студентов знания методов и навыков тестирования программ.

Студенту давались словесное описание задачи, подлежащей решению, и соответствующая программа на языке высокого уровня. Требовалось построить по возможности наиболее представительную систему тестов, проверяющую решение задачи. Студент должен был представить словесное описание входных данных теста и реакцию программы на них: правильный (неправильный) результат, заикливание, прерывание и т. д.

Задача оценивалась пятью баллами.

Автокод. Цель задания — выявить у студентов знания по ЕС ЭВМ, приемам и методам разработки программ на автокоде с использованием, если необходимо, макросредств. Обычно предлагалась задача из области текстовой обработки или вычислительной математики.

Задача оценивалась пятью баллами; дополнительные баллы назначались за наиболее простой алгоритм, наиболее понятную для чтения и эффективную программу, за минимальное число запусков программы для отладки.

Оптимизация. Требовалось в готовой программе (дается текст программы и словесное описание задачи) обнаружить ошибки без выхода на ЭВМ и провести ее оптимизацию программистскими средствами: чистка циклов, экономия выражений, уменьшение силы операций и т. п.

Решение оценивалось тремя баллами и должно было состоять из словесного

описания предлагаемых изменений и исправленного текста программы.

Приведем примеры заданий.

1. Вычислить максимальную сумму попарных произведений элементов двух 10-мерных векторов при условии, что каждый из элементов участвует в вычислении суммы не более одного раза.

Очевидное решение состоит в переборе всевозможных попарных произведений. Некоторое улучшение алгоритма дает построение по векторам А и В матрицы с элементами $x_{ij}=a_i b_j$, если $a_i b_j > 0$, и $x_{ij}=0$ — в противном случае. В этой матрице необходимо отыскать элементы, по одному в каждой строке и в каждом столбце, такие, чтобы их сумма была максимальной. Если же сумма окажется нулевой, то дополнительно необходимо убедиться, что в векторах А или В есть элементы, равные нулю. Если нулевых элементов нет, то следует просто найти максимум из чисел $\{a_i b_j\}$ с учетом их знаков.

Лучшее решение состоит в упорядочении векторов А и В по неубыванию и суммировании положительных слагаемых $a_i b_i$. Если же все $a_i b_i < 0$, то в качестве решения берется $\max(a_1 b_{10}, a_{10} b_1)$.

2. Для приводимой ниже процедуры составить возможно более полную систему тестов. Процедура-функция отыскивает элемент х в массиве А(1), ..., А(К) методом деления пополам.

```
IND: PROC (A,K,X) RETURNS (DEC(4));
DCL (X,A(K)) DEC(5);
DCL (K,I,J,L,M) DEC(4);
```

```
SPUR: PROC OPTIONS (MAIN);
DCL A(50:50) FLOAT, (X(N),S,S1,S2,I,J,N,K) FIXED;
GET LIST(N); GET EDIT(X) (F(2,1));
DO I=1 TO N; /*Построение матрицы*/
DO J=1 TO N;
IF I<J<=N THEN
S=0;
DO K=1 TO N;
S1=S1+SIN(18)*(I+J)*(SIN(X(I))*
COS(X(J))+0.5/(SIN(X(J))+COS(X(I)))*2+K;
END;
A(I,J)=S1;
ELSE
S2=0;
DO K=1 TO N;
S2=S2+COS(72)*(-1)**K*2**K/2**K**2;
END;
I=I+1;
END;
DO I=1 TO 50; /*Вычисление следа матрицы*/
S=0;
IF I=J THEN
S=S+A(I,J);
END;
END;
PUT EDIT ('СЛЕД='S) (SKIP,A,F(6));
END SPUR;
```

```
I=1; J=K;
DO WHILE (I<J);
L=(I+J)/2;
IF A(L)>X THEN
J=L-1;
ELSE I=L;
END;
IF A(I)=X THEN
RETURN(I);
ELSE
RETURN(999);
END IND;
```

Студенты должны были сформулировать какой-нибудь критерий полноты тестирования и в соответствии с ним подготовить набор тестов. Жюри обращало внимание на наличие не только «правильных» тестов, но и тестов, проверяющих неверные входные данные ($K \leq 0$), и тестов для тонких ошибок ($K > 999$, искомого элемента нет в А).

3. Автокод. Сосчитать количество чисел из $(2^n, 2^{n+1})$, кратных трем, у которых количество единиц в двоичном представлении нечетно; n вводится с перфокарты.

4. Приводимая ниже программа вычисляет след матрицы $n \times n$ с элементами

$$a_{ij} = \begin{cases} \sin 18^\circ \sum_{k=1}^{i+j} (i+j) \frac{\sin x_i \cos x_j + 0,5}{(\sin x_i + \cos x_j)^2} k & \text{при } i \leq j \leq n; \\ \cos 72^\circ \sum_{k=1}^n (-1)^k \frac{2^k}{2k^2} & \text{при } 1 \leq j \leq i. \end{cases}$$

Необходимо: а) указать построчно все синтаксические и смысловые ошибки в приводимой ниже программе; б) написать оптимизированный вариант.

Ошибки в программе очевидны, точно так же, как и фрагменты, подлежащие оптимизации «в лоб», без размышлений. Лучшее решение состоит в получении формулы для a_{ii} и ее программировании:

$$\sum_{i=1}^n a_{ii} = \frac{n^2(n+1)^2}{2} \sin 18^\circ.$$

Итоги олимпиад показывают, что наибольшие трудности у студентов вызывают задачи на тестирование и оптимизацию программы за столом. По-видимому, это обусловлено недостаточным вниманием к этому материалу на лекциях и особенно на практических занятиях.

Победителями в командном зачете были в разные годы команды Москвы, Белоруссии, Украины, РСФСР.

В будущем полезно было бы, на наш взгляд, изменить формулу проведения олимпиад, взяв за основу не индивидуальную форму соревнования, а коллективную, поскольку именно она наиболее типична для работы программистов. Задание должно выдаваться одно на всю команду. Члены команды в соответствии со своей специализацией (алгоритмист, кодировщик, тестовик) готовят свою часть задания и вместе создают и отлаживают программу. При таком подходе можно усложнить задания и не ограничивать студентов количеством выходов на ЭВМ — важен результат, полученный за ограниченный отрезок времени. Если при этом студенты имеют в своем распоряжении персональные ЭВМ, то отпадает необходимость в перфорации и контроле за числом исправлений, вносимых в программу при ее отладке.

Эргономика рассудит

«Хозяйке на заметку» — подобные рубрики есть во многих газетах. Как перешить немодное платье, что делать с больным ребенком до прихода врача, где хранить лыжи летом... А вот в газете «Вашингтон пост» начали печатать полезные советы по части компьютеризации.

В частности, газета рекомендует при работе за дисплеем сидеть в максимально свободной и удобной позе, пользоваться подвижным стулом. Экран должен быть расположен в 35—50 см от глаз (имеется в виду экран специального монитора, а не заменяющего его телевизора) и — внимание! — ниже их уровня: взгляд должен быть направлен вниз под углом 20—30 градусов. Все необходимые справочные материалы следует размещать рядом с экраном, желательно на таком же расстоянии, чтобы, переводя на них взгляд, не утомлять зрачок необ-

ЧТО? МОЖЕТ ЭВМ

ходимостью изменения фокусировки.

Конфигурация достаточно новая, даже неожиданная. Не придется ли теперь менять эргономику КУВТов? Гигиенисты, не затягивайте с ответом!

Компьютер «выпадает в осадок»

Это вовсе не означает, что компьютер чем-то изумлен, а журнал переходит на молодежный жаргон. Речь идет о том, что можно смешать некие жидкости, потрясти пробирку, и на ее дно осядет... пусть не компьютер, но его важнейшие компоненты.

В американском журнале «Про-теин инжиниринг» помещена статья сотрудников университета штата Нью-Йорк, в которой предлагается подход к созданию логических электронных элементов из отдельных молекул. В качестве арматуры, полагают авторы, можно использовать спираль ДНК — она обладает достаточной жесткостью. Есть у нее и другие полезные свойства, позволяющие, в частности, организовывать самосборку коротких отрезков спирали в разветвляющуюся пространственную конструкцию, напоминающую план города с улицами и перекрестками.

К некоторым из этих коротких фрагментов можно заранее химически «пришить» молекулы полимера, способные проводить электрический ток; к другим — «цифровые элементы» из молекул, имеющих два устойчивых состояния, разделенных энергетическим барьером. А раз есть проводники и бистабильные элементы, значит, можно создавать сложные электронные схемы.

Что-то будет завтра?!

Журнал в журнале
для школьников, студентов,
учащихся СПТУ и техникумов
Издается при участии ЦК ВЛКСМ

МОЛОДЕЖНАЯ ИНИЦИАТИВА

5

В НОМЕРЕ:

ПРАЗДНИК В МОСКОВСКОЙ ШКОЛЕ

**НОВАЯ РУБРИКА «КЛУБ ПОЛЬЗОВАТЕЛЕЙ БК» ПРЕДСТАВЛЯЕТ
СЕРИЮ МАТЕРИАЛОВ, НАПИСАННЫХ ЛЮБИТЕЛЯМИ И ПРОФЕС-
СИОНАЛАМИ**

ИНФО-88

СТЕРЕОИЗОБРАЖЕНИЕ НА ДИСПЛЕЕ

Праздник информатики состоялся

В специальной математической школе № 444 проходит праздник Юношеского научного общества...

Торжественно убран актовый зал. По стенам развешены плакаты и фотомонтажи. До начала праздничного действия в зале проходит выставка картин Егора Трофимова.

Пока собираются зрители, трое мальчишек суетятся вокруг дисплея, раскинувшего по сценическим подмосткам свои провода.

88

У секции ЮНО сегодня юбилей — ей исполнился год.

Школьники здесь обучаются принципам и технике программирования, сами разрабатывают программы. И небезуспешно: грамота МГУ, диплом физтеха.

Без компьютера в наше время — никуда. Человек должен уметь плавать в море информации.

Школа № 444 — математическая. Выпускники, заканчивая учебу, обязательно пишут и защищают дипломную работу. Это не игра в информатику. «Технология соз-

дания и ведения базы данных на основе 4-го энергоблока Белоярской АЭС» А. Розанова и М. Царькова — одна из лучших работ. Секция ЮНО — поставщик светлых голов.

Какой же праздник без праздничного представления? Но спектакль спектаклю — рознь. Эта постановка — и реклама, реклама ЮНО, и небольшой экскурс в историю информатики.

Затем на школьном празднике был показан спектакль-сказка про Иванушку-дурачка и его супругу — ЭВМ-Царевну-лягушку...

И мы несколько отвлекаемся и тоже расскажем коротенькую сказку, только свою.

В общем, так. Жили-были трое братьев. Двое, как все, а третий Иванушка. Ученик 9-го класса одной московской школы. Долго ли, коротко ли, пустили братья по



стреле. И оженлись. И Иван тож. А поженили Иванушку-то на ЭВМ. Всем хороша оказалась жена. Да вот условия не ахти: жилплощади молодой семье не выделили — ютятся все вместе в одной комнатухе; язык для советского программирования никак не найдут; ни времени тебе друг для друга нет, ни дискет. Им бы книжек почитать по психологии такой семейной жизни — да нету книжек...

Так вот пока и живут...

К чему сказка эта? Да все к тому же: мало машин, не хватает площадей, нет над-

лежащего программного обеспечения...

Есть ЮНО. И, конечно, есть энтузиасты своего дела — такие, например, как руководитель школьной секции ЮНО Н. Пахомова. Но энтузиазма мало, нужна мат-

база и еще...

И еще, как сказал нам М. Донской, автор знаменитой «Кайссы», нужно, чтобы «компьютер был прозрачным». Попросту говоря, компьютерное образование должно стать «подкладкой»

тех фундаментальных навыков, знаний и умений, которые ребята получают в школе.

**А. СУББОТИН,
В. БУЛДАКОВ,
студенты МГУ**

КЛУБ ПОЛЬЗОВАТЕЛЕЙ БК

Московский клуб БК

Когда три года назад в продаже наконец появились первые бытовые компьютеры «Электроника БК-0010», никто из покупателей еще не представлял, с какими трудностями придется столкнуться, осваивая эту машину. Надо сказать, что возможности БК казались тогда ограниченными настолько, что никто просто не мог поверить, что для этой машины можно создать, например, такую программу, как «Шахматы» (С. Шмытов, Москва).

Действуя по известной поговорке «ум хорошо, а два — лучше», немногочисленные счастливые обладатели БК из Москвы стали встречаться друг с другом; образовались небольшие группы по интересам. Постепенно встречи стали регулярными, хотя и на разных «площадках»: то в здании издательства «Мо-

лодая гвардия», то в Политехническом музее.

Основой этих встреч был обмен программами (создавался общедоступный фонд) и всевозможной информацией, касающейся компьютеров вообще и БК в частности.

Сегодня «Московский городской клуб пользователей бытовых компьютеров» — это крупнейшее в стране любительское объединение пользователей «Электроники БК». Клуб решает много задач, главные — оказание помощи начинающим пользователям компьютеров, создание общедоступного фонда программного обеспечения. Клуб объединяет уже более 500 человек, поддерживает связи с другими компьютерными клубами страны.

Адрес одной из площадок клуба — Москва, ул. Павловская, 6, ДК ЗВИ, к. 218.

«Электроника БК-0010» работает с магнитофоном

Пока у БК нет дисковода, единственный способ сохранить программу — это записать ее на магнитофон. Магнитофон может быть любым: отечественным или зарубежным, новым или ста-

рым, кассетным или катушечным. Единственное условие: он должен надежно воспроизводить последовательности импульсов — компьютерные программы. Надежность зависит от многих

причин, но есть несколько основных параметров, которые чаще всего определяют, насколько удобно работать с домашним вычислительным комплексом. Во-первых, это уровень сигнала, — обычно от 0,25 до 0,8 В. Оптимум здесь находится примерно посередине. Второе условие — верное положение головки магнитофона. Бывает так: программы, которые вы составляете самостоятельно, хорошо записываются и считываются с магнитофонной ленты, а кассету, полученную из другого города, прочесть не удастся. Дело в том, что магнитные головки во всех магнитофонах должны находиться строго на одной стандартной высоте — только в этом случае запись, сделанная на одном магнитофоне, подойдет к другому аппарату. К сожалению, так случается далеко не всегда, особенно когда речь идет о портативных кассетных магнитофонах. В то же время в большинстве случаев положение можно исправить. Прежде всего разыщите кассету с эталонной записью (есть специальные кассеты для измерений, но подойдет любая кассета, записанная на заведомо отрегулированном магнитофоне). Слегка поворачивая регулировочный винт, нужно выставить стандартную высоту головки. Ориентироваться при этом следует прежде всего на качество воспроизведения

высоких частот. Отметив эталонное положение винта краской, можно попытаться прочесть программы, записанные на «неправильных» магнитофонах, также ориентируясь на то положение головки, когда лучше всего воспроизводятся высокие частоты. Однако настроить маг-

нитофон, руководствуясь исключительно слухом, довольно сложно. В то же время БК-0010 вполне может оказать серьезную помощь в этом деле, если воспользоваться программой С. Гуторенко из г. Железнодорожного (Московская обл.).

В этом случае управление вновь передается на п. 2.

Наконец анализ закончен и начинается обработка результатов. Увеличивается на единицу элемент массива с номером, соответствующим длительности периода. При этом компьютер следит за тем, чтобы график не вышел за пределы экрана. По горизонтальной оси в верхней части экрана откладывается длительность периода, по вертикальной — количество периодов соответствующей длительности. Как только график построен, выполнение программы повторяется с начала.

Способ С. Гуторенко хорош, однако его можно дополнить еще несколькими приемами, позволяющими выбраться почти из любой затруднительной ситуации с магнитофоном. Если магнитофон работает с компьютером хорошо, то можно воспользоваться приемом, ускоряющим ввод программ в компьютер. Для этого программу записывают на кассетный магнитофон со скоростью 2,38 см/с, а воспроизводят со скоростью 4,76 см/с. Для катушечного магнитофона это 9,53 см/с и 19,05 см/с или 4,76 см/с и 9,53 см/с. В тех моделях, где переключателя скорости нет, его можно сделать. Для этого последовательно с подстроечным резистором R в блоке управления скоростью устанавливают цепочку из выключателя и резистора, включенных параллельно (рис. 2). Разомкнутое положение переключателя соответствует режиму «Запись с БК», замкнутое — «Загрузка в БК». Если считывание с магнитофона идет плохо, попробуйте обратный прием: запись на большей скорости, воспроизведение на меньшей. Если слишком мала ампли-

БК считает периоды

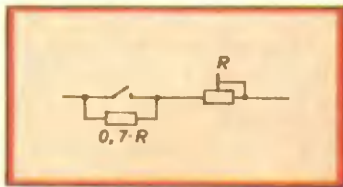
После того как программа (рис. 1) введена в компьютер, ее запускают с адреса 1000. На экране ничего не произойдет до тех пор, пока мы не поставим на магнитофон качественную запись и не включим воспроизведение. Через 2 с на экране телевизора появится график, который будет постоянно обновляться. Если запись действительно качественная и головка установлена верно, на экране будут два четких максимума различной высоты, а между ними — четкий минимум, доходящий до нуля. Правый максимум должен быть меньше левого. Если вы увидите лишь один максимум, это значит, что сейчас воспроизводится настроенная последовательность или файл, содержащий один нуль. Но если поставить кассету, которую компьютер не считывает, то вид графика тотчас же изменится: пропадет нулевой минимум, максимумы расплывутся. Глядя на картинку, можно попытаться подстроить магнитофон.

Вводят программу из отладочного режима начиная с адреса 1000. Длина программы 134 байта, контрольная сумма 31 571. Запускают программу командой 1000G в режиме «64 символа в строке». Алгоритм программы несложен.

1. Запустив двигатель магнитофона, компьютер очищает массив, в котором будет накапливаться статистика, полученная при обработке сигналов магнитофона.

2. Компьютер запоминает состояние сигнала на выходе магнитофона (0 или 1) в текущий момент времени, очищает счетчик и начинает цикл анализа сигнала: содержимое счетчика увеличивается на единицу и, если возникает переполнение, управление возвращается на п. 2. Если с магнитофона продолжает поступать тот же сигнал, то управление возвращается к началу цикла, в противном случае цикл завершается. Каждый сигнал, длительность периода которого оказывается больше, чем 512 циклов анализа, БК считает ошибоч-

12705	177716	42715	200
12700	1134	12701	400
5020	77102	11501	5002
5202	102774	21501	1774
20227	1000	2367	105262
1134	126227	1134	357
103761	12700	14	104016
5000	5001	5002	104030
5200	5003	116302	1134
42702	177400	104032	5203
62701	40	20127	1000
2765	722		



туда сигналов на линейном выходе магнитофона, можно попытаться воспользоваться гнездом для подключения наушников. Амплитуду импульсов на выходе — 0,5 В устанавливают ручкой громкости (для этого лучше всего воспользоваться осциллографом). Манипулируя ручками тембра, нужно отыскать такое положение, когда форма импульсов будет наиболее близка к прямоугольной.

И наконец, еще несколько советов, как хранить программы. Прежде всего, кассет должно быть несколько, даже если у вас пока что всего около десятка программ. Одну из кассет выделяют под запись программ в машинных кодах, другую — для программ на Бейсике-MSX, третью — для Форт-программ или для программы на Т-языке. Одну из кассет выделяют под разработку новых программ. Можно завести специальную архивную кассету, куда записывать по одному дублю каждой программы, чтобы исключить возможность ее утраты. Чтобы реже регулировать положение головки, имеет смысл завести специальную обменную кассету для записи программ на других магнитофонах, с которой затем переписывать новые программы на свои кассеты в стандартном положении головки.

Каждую программу нужно записывать, как минимум, дважды. Возможны два варианта, впрочем, оба они не лишены недостатков. Можно записывать дубли подряд, но тогда трудно будет находить нужную программу на ленте, если магнитофон

не имеет счетчика. Другой способ — записывать один дубль с каждой стороны кассеты и, тотчас же перевернув кассету, — второй дубль. В этом случае программы располагаются компактно, но механическое повреждение ленты почти наверняка повредит оба дубля. Можно предложить еще немало приемов: например, записы-

вать после каждой программы серию тональных сигналов или проговаривать в микрофон название программы, чтобы легче было находить нужную. Впрочем, отыскать эти или свои собственные приемы совсем несложно — все зависит от вашей фантазии и настойчивости.

А. БОЙКО,
Москва

Программная орбита

9. МИРАЖ SZ/01—88 (С. Зильберштейн, г. Киров). Машинные коды. Это мини-операционная система низкого уровня, состоящая из экранного ассемблера, дис-ассемблера, средств отладки и средств работы с файлами. Система предназначена для разработки небольших программ на ассемблере, модификации существующих кодовых программ, обучения программированию на уровне машинных команд БК-0010. Система «МИРАЖ» не занимает места в ОЗУ программ, поскольку она размещается в восьми нижних строках экрана.

10. PRINTFOC (АСП, Москва). Машинные коды. Вывод из Фокала БК текстовой информации на принтер.

11. GRAFFOC (АСП, Москва). Машинные коды. Вывод из Фокала БК графической информации на принтер.

12. DRIVER (М. Сикорский, г. Сыктывкар). Машинные коды. Вывод из БК текстовой информации на принтер ROBOTRON CM 6329.01-М.

13. TRF-88 (А. Ромашевский, Минск). Демонстрационный текстовый редактор, позволяющий не только набирать и редактировать текстовую информацию, но и производить некоторые нестандартные операции, такие, как кодирование текста и его дешифровку.

14. КЛАСС-ЭКЗАМЕНАТОР (А. Тихонов, г. Петропавловск-Камчатский). Операционная система, рассчитанная для работы в КУВТ-86, предназначенная для проверки знаний по курсу информатики с ориентацией на язык программирования Фокал БК-0010. Содержит 39 файлов на одном диске.

15. GARDEN (АСП, Москва). Динамическая игра в машинных кодах. На фруктовый сад, в котором вырос богатый урожай, напали прожорливые чудовища. Необходимо как можно быстрее собрать все фрукты.

16. LODERUNNER (А. Марков, Ленинград). Динамическая игра в машинных кодах. Цель игры — пройти несколько лабиринтов, собирая клады и стараясь избежать опасных встреч с хранителями кладов. Надо уметь

91



расставлять ловушки, а за каждые десять кладов добавляется одна «жизнь». В отличие от большинства игровых программ предусмотрено редактирование лабиринтов и создание новых маршрутов, причем их можно записывать на магнитофон. Программа с музыкальными эффектами.

17. MONEY (М. Бураков, Ленинград) — на Фокале.

Пакет из четырех программ для ведения семейного бюджета. Программы наглядно показывают динамику, структуру и другие показатели экономики семьи.

18. SLV-3 (М. Сикорский, г. Сыктывкар) — на Бейсике-MSX. Пакет из трех программ для решения системы линейных уравнений обычным методом, методом Гаусса и методом вращения.

компьютер сравнивает его со входным напряжением до тех пор, пока они не станут равными. Число от 0 до 255, записанное к этому моменту в порт вывода, соответствует измеряемому напряжению. Программное обеспечение АЦП — одна строка на Фокале (рис. 2).

Результат работы программы — число V , соответствующее входному напряжению. Алгоритм измерений напоминает алгоритм решения уравнения методом половинного деления. Все восемь циклов, необходимых для получения результата, БК выполняет за полсекунды. Если такое быстрое действие окажется недостаточным, можно воспользоваться аналогичной программой в машинных кодах (рис. 3). Результат измерений записывается в регистр R0. БК с такой программой может работать как осциллограф, если только частота исследуемого сигнала не превышает 100 Гц. Изображение сигнала в этом случае выводится на экран телевизора в любом удобном масштабе. Для измере-

Не только цифровые...

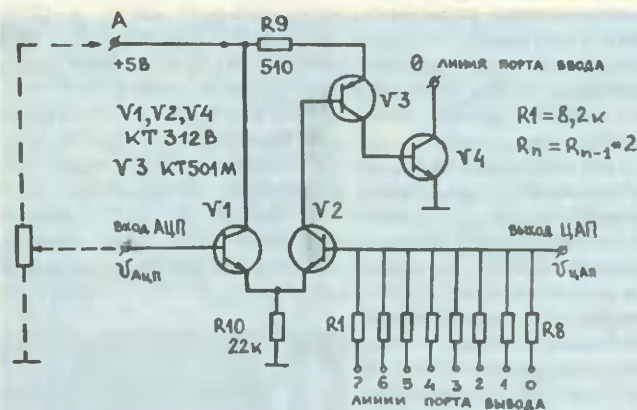
92

Известно, что «Электроника БК-0010» способна управлять различными электрическими устройствами не только в быту, но и на производстве. Если устройства цифровые, то никаких сложностей не возникает. Другое дело, когда работать предстоит с аналоговыми сигналами; тут не обойтись без цифро-аналогового и аналого-цифрового преобразователя. Сделать эти устройства не так уж сложно, воспользовавшись опытом С. Камнева.

Простейший ЦАП для БК — это восемь резисторов (рис. 1). Напряжение на выходе ЦАП будет пропорционально числу, записанному в порт вывода. Например, если записать в порт число 128, которому соответствует двоичное число 10000000, то на линиях 0—6 появится напряжение +5 В, а на линии 7 напряжение будет отсутствовать. Суммарное напряжение на выходе схемы окажется равным 2,5 В. Засылая в порт вывода числа от 0 до 255, можно регулировать напряжение на выходе ЦАП от 5 до 0 В. Качество работы преобразователя определяется тем, насколько точно подобраны резисторы R2—R8; особенно важно, чтобы выполнялось соотношение

$$R_n = 2 \cdot R_{n-1}$$

Аналого-цифровой преобразователь — это тот же ЦАП, дополненный ячейкой сравнения: транзисторы V1—V4, резисторы R9 и R10 (рис. 1). Вход А подключают к любому контакту разъема, где есть напряжение +5 В. Принцип работы схемы несложен: если напряжение на входе АЦП меньше, чем на выходе ЦАП, то транзисторы V2—V4 открываются и в порт ввода поступает «1», в противном случае — «0». Изменяя напряжение на выходе ЦАП,



$$1.1 \quad S U = 128; S A = U; F K = 7; X F X(1, 177714, U); S A = A/2; S U = U + 2 \cdot A \cdot F P(, 1) - A$$

10246	12700	200	10002	110037	177714	6202	1410
32737	1	177714	1402	60200	766	160200	764
32737	1	177714	1002	5300	12602		

ний такой осциллограф вряд ли пригодится — величина V изменяется от 25 до 235, что соответствует напряжению от 5 до 0 В. Впрочем, для измерения больших напряжений схему можно дополнить резисторным делителем, причем переключение диапазонов — задача, доступная для компьютера.

Есть у АЦП — ЦАП еще одно интересное применение:

подключив к АЦП переменный резистор R_p , получим превосходную «ручку управления», или «круль», для различных игровых программ. В отличие от обычного джойстика таким АЦП-джойстиком можно задавать не только направление, но и скорость перемещения.

С. КАМНЕВ,
Москва

Изолятором между контактами зачастую служит воздушное пространство, образовавшееся из-за того, что в верхней пластине выдавили некоторое подобие клавиш. Нажимая на такую клавишу, мы деформируем пластик и замыкаем цепь. Упругие свойства пластика таковы, что без каких-либо пружин он принимает исходное положение и может выдерживать сотни тысяч и более нажатий.

Пленочные переключатели

Пожалуй, ни одно электронное устройство не обходится без каких-либо кнопок или переключателей.

Стоит взглянуть на них, чтобы убедиться, что подавляющее большинство — это довольно сложные конструкции, состоящие из нескольких отдельных элементов, а значит, кроме изготовления этих элементов нужно предусмотреть еще и сборку. Каждый элемент и каждая операция таят в себе какую-либо причину будущих отказов. И, как это часто бывает, из-за поломки какой-то одной клавиши нельзя в нужный момент воспользоваться, например, тем же компьютером. В то же время в компьютере может быть 40, а то и 60 одинаковых по конструкции клавиш.

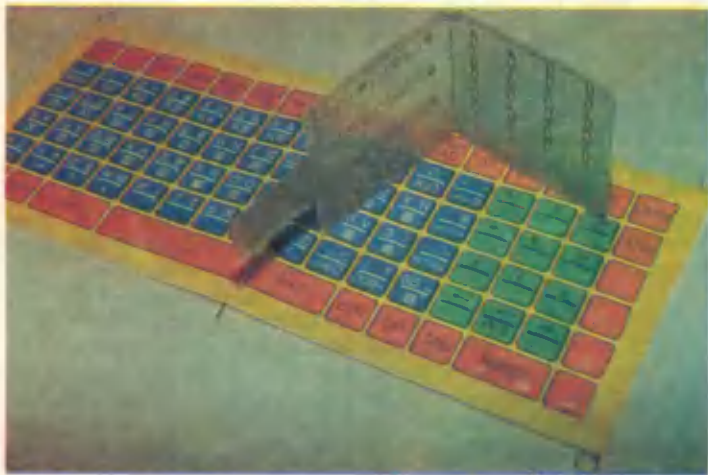
Разработчики новых клавиатур руководствовались, по-видимому, известной в микроэлектронике интегральной технологией, когда при одновременном изготовлении устройства из типовых элементов надежность всего устройства практически не отличается от надежности одного элемента. Так появились пленочные переключатели, успешно работающие в пультах управления различными устройствами на производстве и в быту.

Небольшая западногер-

манская фирма «Клемм» — один из крупнейших производителей печатного оборудования, на котором изготавливаются разнообразные пленочные переключатели. Все операции осуществляются на линии из нескольких станков, с которой сходят уже полностью готовые к эксплуатации клавиатуры, например такие, как те, что изображены на фото. Можно печатать клавиатуры двух типов: многослойные и однослойные. Что же такое пленочные переключатели? Это напыленные на пластик контактные площадки и дорожки соединительных проводников. Дорожки покрывают лаком, чтобы исключить неправильные соединения.

Иногда клавиатуру делают плоской; тогда, чтобы обеспечить изолирующий зазор, между контактными пластинами прокладывают пластиковую пластину с отверстиями. В этом случае можно обойтись даже без изолирующего лака. Кстати, именно по такому принципу сделана клавиатура для БК, показанная на фото. В отличие от той, что используется в БК сейчас, новая клавиатура не содержит ни одного паяного соединения, ее подключают к компьютеру с помощью специального разъема. Кстати, точно такая клавиатура стоит на детской электронной игрушке «Луноход» — сломать ее очень трудно.

Можно спорить, будут ли по такому принципу делать все компьютерные клавиатуры.



туры; многие считают, что для ЭВМ необходима клавиатура с большим ходом клавиш, как на пишущей машинке. Но в калькуляторах, телефонных аппаратах и в промышленной электронике, где нужно заботиться,

например, о том, чтобы защитить устройство от грязи, пыли, влаги (а пленочную клавиатуру можно мыть), такая клавиатура вскоре, несомненно, «приживется».

А. БОРИСОВ,
Москва

которая более не нужна, легко удалить из памяти командой L E, ИМЯ, причем остальные программы сдвинутся так, чтобы передать освободившееся место свободной части ОЗУ. С той же легкостью включаются в программу и более сложные блоки, например вычисления интегрального синуса. $Y=1 + \text{FINT}(0,2,1E-8, «YX, = \text{FS}(X)/X», /J, E); ZT$

Число 0 и 2 — пределы интегрирования, $1E-8$ — требуемая погрешность вычисления интеграла. Подынтегральная функция оказалась простой, поэтому она записана среди параметров подпрограммы (выделена кавычками). Значение синуса вычисляет функция Фонда FS(X), которая отличается от аналогичной функции Фокала только лишь большей точностью. Результат вычисления хранится в памяти, как значение переменной J, а E — вычисленная оценка погрешности вычислений. Столь короткое обращение позволяет вставлять процедуру интегрирования в любое арифметическое выражение.

Если подобных подпрограмм будет достаточно много, то на долю медленного Фокала останется лишь роль обрамления — своеобразного «связующего раствора» для «кирпичиков» — подпрограмм. С помощью Фонда можно, например, реализовать любые операторы любой версии Бейсика. Но в отличие от настоящего Бейсика такой суперФокал не занимает память постоянно, необходимые подпрограммы загружают лишь тогда, когда они будут использованы в данной конкретной программе.

Впрочем, потенциальные возможности Фонда гораздо шире, чем у любого штатного Бейсика. Интересы, например, подпрограммы, позволяющие включать в Фокал-

Супер-Фокал, или Фонд БК

Создавать программы для БК было бы гораздо проще, если бы в Фокале или Бейсике существовали подпрограммы типа фортрановских процедур. Программа Фонд, о которой пойдет рассказ, реализует эту идею. Загруженный в память БК, Фонд существенно расширяет возможности Фокала: позволяет увеличить скорость и одновременно повысить точность арифметических вычислений, с его помощью можно обращаться к подпрограммам, составленным на Фокале или в машинных кодах. Фонд занимает в памяти 4,5 К, но этот расход многократно окупается теми возможностями, которые получает программист. Но для того, чтобы Фонд стал по-настоящему мощной системой программирования, необходимы общие усилия всех программистов, работающих с БК. Программу Фонд и подпрограммы, которые уже собраны, можно получить у членов московского клуба БК (см. заметку о нем в этом номере), там же предполагается накапливать новые подпрограммы, созданные всеми программистами, которые бы хотели принять участие в этой интересной и полезной работе.

Разговор о Фонде лучше всего начать с небольшой разминки — решить квадратное уравнение, классический пример пособия по программированию. Разумеется,

нам понадобится подпрограмма решения такого уравнения, она существует, и ее можно считать с магнитофона оператором L CALL FSQRT. Нетрудно догадаться, что FSQRT — подпрограмма, CALL — новый оператор Фокала для вызова подпрограмм (можно просто С). Подпрограмма автоматически записывается на свободное место в памяти, а ее данные направляются в таблицу подпрограмм. Теперь к ней можно обратиться еще одним новым оператором JSR (можно просто J). Чтобы решить квадратное уравнение $Ax^2 + Bx + C = 0$, достаточно теперь выполнить один оператор J FSQRT (A, B, C, /X1, X2). Переменные X1 и X2 при этом примут значения корней, вычисленные с точностью до девяти знаков. Программы, составленные на основе возможностей, представленных Фондом, легко читаются.

1.10 J FSQRT (V, 5.5, C—4, /P, PP)

1.20 Y=P; ZT; V=PP; ZT

Первая строка решает квадратное уравнение $Vx^2 + 5,5x + C = 4$, во второй встречаются еще два оператора Фонда. Это Y— оператор вычисления арифметического выражения и ZT — оператор вывода результата вычислений на экран. Эти операторы аналогичны операторам X и T Фокала, но работают с числами, содержащими девять десятичных разрядов. Подпрограмму,

программу одноголосые мелодии или эффекты, имитирующие вой сирены или хлопок выстрела. Подпрограмма FTITR (X, Y, MX, M, Y, «ТЕКСТ») выведет на экран произвольный текст, но буквы будут увеличены по горизонтали и вертикали в MX и MY раз. Созданные в графическом редакторе кадры мультипликации также можно включить в программу на Фокале с помощью подпрограммы Фонда. А рисунки, в свою очередь, нетрудно оформить, как новые подпрограммы Фонда: шрифты, математические символы, знаки дорожного движения,

элементы радиосхем.

Хотя простейшие подпрограммы создаются непосредственно на Фокале, эффективными будут подпрограммы в кодах. И, конечно, не хотелось бы лишней раз изобретать велосипед — хорошо бы использовать для создания Фонда множество алгоритмов, накопленных, например, для Фортрана. Кстати, в Фонде уже сейчас предусмотрены некоторые средства, облегчающие ручной перевод процедур Фортрана на машинный язык БК-0010.

Г. ПРИС,
Москва

сказал автору С. И. Следнев, известный среди пользователей БК-0010 энтузиаст-любитель. Будем надеяться, что ей воспользуются не только обладатели БК, но и завод-изготовитель.

Суть предлагаемого решения в том, что следует внести небольшое изменение в схему БК-0010.01, обеспечивающее снижение быстродействия клавиатуры: поставить вместо R3 и R4 (см. рис.) сопротивления по 180 кОм, емкость конденсатора C3 увеличить до 0,33 мкФ, а емкость C4 — до 0,47 мкФ.

Так как эта работа связана со вскрытием корпуса компьютера, то будет лучше, если ее выполнит специалист ремонтного предприятия в порядке технического обслуживания; впрочем, она по силам любому, имеющему навык работы с паяльником.

Нетрудно устранить и неприятные щелчки и заедания клавиатуры. Для этого рекомендуется сделать следующее:

снять клавиши с клавиатуры, что легко сделать, поддев каждую из них снизу отверткой (не снимается только клавиша «пробел»);

положить в пространство между клавишами подушечки из тонкого поролона;

смазать внутренности клавиш графитной или иной тугоплавкой смазкой, после чего установить клавиши на место.

Если осуществить все вышеописанное, то «дребезг клавиш» исчезнет навсегда.

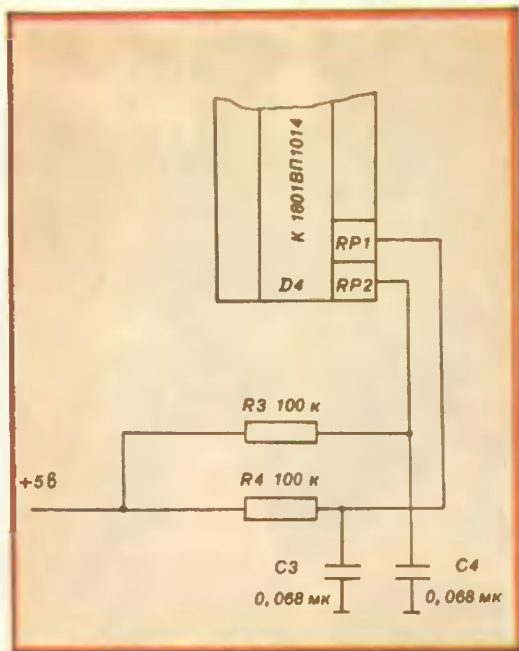
Автор желает заинтересованным читателям успешно справиться с описанными переделками и тем самым обеспечить себе нормальную работу на БК-0010.01.

А. Гриценко,
зав. кабинетом ОИВТ
Алтайского КИУУ

Устранение «дребезга клавиш» у БК

Всякому пользователю БК-0010.01 с полноходовой клавиатурой хорошо известно такое неприятное явление, как «дребезг клавиш»: при нажатии на клавишу вме-

сто одного символа на экране дисплея появляются два и больше, или вместо нужного символа появляется совсем другой. Идею устранения этого недостатка под-



Стереоизображение на дисплее

96

Работая в школьном КВТ, мы заинтересовались применением так называемого метода цветных анаглифов. Его суть — создание стереоэффекта при рассмотрении двух изображений различных цветов, размещенных с некоторым горизонтальным сдвигом одно относительно другого через очки с подобранными разноцветными светофильтрами. Одно из изображений (свое для каждого глаза) гасится соответствующим светофильтром, и создается эффект объемности.

ЭВМ, оснащенная цветным дисплеем, предоставляет

пользователю интересные возможности в этой области. Программы с использованием метода цветных анаглифов могут найти широкое применение — от обучения до игры. Наглядность расположения предметов в трехмерном пространстве может улучшить усвоение учащимися таких дисциплин, как стереометрия, стереохимия, генетика, многие разделы физики. Если реализовать движение стереопары с изменением линейных размеров фрагментов и сдвига между ними, то получится модель движения в трехмерном пространстве.

В нашем школьном творческом объединении (ШТО) составлены пробные программы на языке QASIC-2 (см.: Микропроцессорные средства и системы. 1987. № 2) для ДВК-2 с подключенным телевизором «Электроника Ц-432». Полученные результаты обнадеживают, хотя качество изображения телевизора оставляет желать лучшего. Исследования продолжают.

За справками можно обращаться по адресу: 454080, г. Челябинск, ул. Володарского, 18, ФМШ № 31, ШТО.

А. МАЛЫШЕВ,
г. Челябинск, IX класс

ИНФО-88

29 мая. Второй день проведения конференции — выставки — ярмарки ИНФО-88. Уже схлынула волна журналистов и телеоператоров, и в огромном зале нового Дворца молодежи в Москве на Комсомольском проспекте стало просторнее.

Посетители переходят от стенда к стенду, сажаются за компьютеры: пробуют технику, что называется, на зубок.

Участников очень много, более пятисот. Никаких ограничений для участия в ИНФО-88 не было. Наоборот, организаторы старались привлечь как можно больше народу. Ведь подобное мероприятие проводится впервые. Много интересных разработок, которые пользуются большим спросом. Дей-





ствительно, многие стенды уже опустели — техника и программы закуплены.

В рамках ИНФО-88 прошла научно-практическая конференция по вычислительной технике. В программе три десятка докладов — по технологии программирования, машинной графике, прикладному программному обеспечению...

В демонстрационном зале — квалифицированные специалисты. Каждый заинтересованный посетитель незамедлительно получает необходимую информацию. На столах реклама, буклеты, распечатки.

Размах широкий. И необходимый. Надеемся, что следующая выставка-ярмарка не заставит себя долго ждать.

А для тех читателей нашего журнала, кому не удалось побывать на ИНФО-88, мы предлагаем информацию о некоторых участниках.

Московская инициативная группа (МИГ) — творческое объединение научных и ин-

женерных работников, специализирующихся в области информатики и вычислительной техники. Оно входит в

вания и ведет исследования и разработки в области программного обеспечения для мини- и микро-ЭВМ.

Научно-производственная группа «ОФИС» — результат сотрудничества Центрального экономико-математического института АН СССР (отдел АРМ) и экономического факультета МГУ (кафедра математических методов анализа экономики, студенческий вычислительный центр). «ОФИС» выполняет работы по компьютеризации управления с ориентацией на несколько типов персональных ЭВМ и осуществляет гарантийный авторский надзор всех разработок.

И еще один разработчик — Владислав Викторович Шевкунов (группа ALV Software из Москвы). На ИНФО-88



состав «Фонда молодежной инициативы», объединения, начавшего свою деятельность при Балашихинском ГК ВЛКСМ. Объединение работает на принципах хозяйственного расчета и самофинансиро-

ими была представлена разработка СУБД ЭВРИКА для создания и ведения баз данных картотечного типа («Агат»). Система рассчитана на неподготовленного пользователя.



ИНТЕРКВАДРО

**Центр обучения «ИНТЕРКВАДРО»
к вашим услугам**

**ЦЕНТР
ОБУЧЕНИЯ**

Персональный компьютер сможет использовать каждый.

Обучение программированию, использованию микро-ЭВМ и их техническому обслуживанию.

Предлагаются следующие курсы:

программирование и эксплуатация программных средств в различных операционных системах (ЮНИКС, MS DOS);

автоматизация административно-управленческой деятельности;

машинная статистика;

обработка изображений на персональной ЭВМ;

автоматизация издательской деятельности;

программное и техническое обеспечение вычислительных сетей;

организация и ведение баз данных и экспертных систем;

техническое обслуживание микро-ЭВМ типа ИБМ XT/AT и др.;

разработка педагогических программных средств;

языки программирования (Си, Паскаль, Пролог, Рефал и др.);

медицинская информатика;

обучение русскому языку с использованием компьютерной технологии.

Программы курсов адаптируются к требованиям заказчиков и уровню подготовки слушателей.

Эффективность обучения обеспечивается современными методиками, индивидуальным подходом к каждому слушателю, ежедневными практическими занятиями на ЭВМ, предоставлением каждому слушателю уникальной технической документации в личное пользование.

Занятия проводят высококвалифицированные преподаватели и специалисты практики.

Продолжительность обучения от одной до четырех недель.

Иногородние слушатели обеспечиваются гостиницей с размещением в гостиничном комплексе «Измайлово».

Стоимость обучения:

525 рублей при обучении на русском языке;

1250 рублей при обучении на английском, французском, немецком или итальянском языках.

Для преподавателей учебных заведений всех типов предоставляется скидка в 25 %.

До встречи в ЦЕНТРЕ ОБУЧЕНИЯ совместного советско-франко-итальянского предприятия «ИНТЕРКВАДРО».

Дирекция «ИНТЕРКВАДРО»

адрес: 125130, Москва, 2-й Новоподмосковный пер., д. 4

телефон 159-01-47, телекс 412560, факс 943-00-59

Генеральный директор Л. И. Вайнберг.

Центр обучения

адрес: Москва, ул. Усиевича, д. 31а (ст. метро «Сокол»)

телефон 151-29-42

Руководитель Центра обучения А. Н. Маслов.

А. ВАСИЛЬЧЕНКО, В. ФЕДОТОВ,

канд. физ.-мат. наук

Об учебной ЭВМ

Мало кто из работающих на школьной вычислительной технике удовлетворен ее качеством и техническими характеристиками. А что ждет нас в будущем? Пытаться ответить на этот вопрос, не учитывая мнения педагогов, нельзя.

В трех статьях, публикуемых ниже, предлагаются разные подходы к определению параметров ПЭВМ для школы; мы надеемся, что активный отклик читателей поможет решению этой проблемы. Чем шире дискуссия, тем выше вероятность найти оптимальный вариант.

Уже первый опыт использования ЭВМ в школе выявил множество присущих существующим моделям эксплуатационных недостатков, несовершенство архитектуры аппаратных средств и функциональную ограниченность программного обеспечения. Поэтому целесообразно обсудить вопросы конфигурации аппаратных средств, функциональную приемлемость программного обеспечения, основные технические характеристики учебного компьютера массового применения, чтобы прийти к общему соглашению о том, какой должна быть учебная ПЭВМ, удовлетворяющая требованиям широкой компьютеризации учебно-воспитательного процесса.

Аппаратные средства учебного компьютера

Совокупность узлов и блоков, составляющих аппаратные средства поставляемых сегодня в учебные заведения персональных ЭВМ, по большинству технических характеристик не удовлетворяет потребностям

ни сегодняшнего, ни тем более завтрашнего учебного процесса. 99

8- и даже 16-разрядные ПЭВМ при сроке службы, равном десяти годам, в условиях стремительно развивающихся элементной базы и архитектуры построения компьютеров безнадежно устареют морально, не успев оправдать вложенные в них затраты общественного труда.

Не улучшает ситуацию и принятый 1 июля 1987 г. ГОСТ 27201-87 на ПЭВМ [1], ориентированный скорее на возможности промышленности, чем на реальные нужды учебного процесса. Вспомним, например, рекламные фотографии производства учебных «Агатов» («Информатика и образование». 1987. № 6), вызывающие лишь грустную улыбку: несомненно, не на таком технологическом уровне должны быть организованы сборка и испытания этих учебных вычислительных машин.

Себестоимость массового производства изделий электронной промышленности во всем мире неуклонно снижается, для ПЭВМ стоимость системного блока часто оказывается ниже стоимости периферийных устройств. Поэтому целесообразно оснащать учебные заведения специальными учебными ПЭВМ, существенно уступающими по своим возможностям тем, с которыми в последующей своей деятельности учащийся будет иметь дело. Системный блок учебной ПЭВМ должен быть многопроцессорным и включать 32-разрядный микропроцессор RISC-архитектуры и ряд сопроцессоров, возможно меньшей разрядности (в первую очередь арифметический, далее символично-графический и звуковой сопроцессоры).

Хотелось бы иметь в учебном компьютере ОЗУ объемом не менее 256К байт

с возможностью дальнейшего наращивания памяти путем простого подключения дополнительной платы ОЗУ или по крайней мере заменой микросхем памяти. Архитектура и набор команд должны давать возможность эффективного распараллеливания и конвейеризации вычислений. В составе системного блока необходимы контроллеры устройств типа «мышь», «джойстик», «птица» [2] и др.

По нашему мнению, в учебной ПЭВМ обязательно наличие минимум одного встроенного НГМД на 3,5-дюймовых дискетах — наиболее надежных, портативных и достаточно емких носителях информации. Должна быть предусмотрена возможность подключения дополнительного НГМД (возможно, оптического или ЦМД ЗУ), принтера, кассетного магнитофона и усилителя низкой частоты с акустическими системами.

100

Для организации сетевой работы учебных компьютеров и связи с внешним миром необходимы адаптер кольцевой сети с маркерным доступом и модем.

Очень привлекательна идея транслирования учебных программ по телевизионным каналам, следовательно, надо предусмотреть и эту возможность в массовом учебном компьютере.

Недопустимо комплектовать учебную ПЭВМ массового применения бытовым телевизором в качестве монитора. Необходимы монохроматические и цветные мониторы, удовлетворяющие санитарно-гигиеническим нормам [3], основанным на психометрических и инструментальных измерениях качества изображения [4].

Учебная ПЭВМ не должна содержать встроенных вентиляторов (они шумят) и выносных блоков питания, увеличивающих количество внешних кабелей, разъемов да и просто неуместных на рабочем месте учащегося.

Большие эксплуатационные неудобства создает совмещенная с системным блоком клавиатура. Очевидно, что для учащегося эргономичнее работать на выносной плоской полноходовой клавиатуре.

Особые требования в учебной ПЭВМ массового применения предъявляются к надежности работы всех узлов и блоков. Среднюю наработку на отказ в 10 тыс. ч надо признать минимальной нижней границей, а отнюдь не желаемым пределом. Конечно, для достижения даже такой надежности необходимо организовать автоматизированный контроль элементной базы на стадии производства, жесткую отбраковку узлов, блоков и всего компьютера на стадии приемки изделия.

Однако большой психологический вред учащемуся наносят не отказы в работе ПЭВМ, а частые сбои. Для сведения их к минимуму желательно иметь повышенную избыточность каналов обмена информацией или пойти по пути, предложенному в [5].

Программные средства

Оснащать учебную ПЭВМ одной резидентной операционной системой, записанной в ПЗУ, — значит и сужать потенциальные возможности компьютера, и обеднять процесс обучения. Не лучше ли отказаться от такой архитектуры, предоставив развитие этой концепции проектировщикам микроЭВМ, встроенных в разнообразное оборудование (металлорежущие станки, измерительные и медицинские приборы и т. д.)?

Практические занятия на учебных компьютерах, с нашей точки зрения, необходимо больше ориентировать на грамотное и эффективное применение типовых функциональных пакетов прикладных программ, приобретения навыков интенсификации труда с помощью ПЭВМ, а не на приобретение начальных навыков программирования. В конечном итоге написание, отладка, верификация программных продуктов широкого применения есть дело более профессиональное, чем всеобщее, массовое [6]. Может быть, разумнее изучать программирование на факультативных курсах, а приобретать навыки компьютеризированного труда — на основных.

Неуспех части программного обеспечения, используемого при изучении общеобразовательных дисциплин в учебных заведениях, часто вызван не низкой квалификацией писавших его программистов или недостатками выбранного алгоритмического языка, а отсутствием специально ориентированной на учебный процесс программной среды. Такую среду могут предоставить и учащемуся, и преподавателю только типовые экспертные системы (ЭС), разработанные специально для учебного процесса [7—8]. Они, несомненно, являются идеальным средством для организации непрерывного и дифференцированного контроля усвоения знаний учащимся, могут выступать активным средством стимуляции его индивидуальной творческой активности.

Широкое внедрение ЭС в сферу образования потребует использования различных инструментальных средств внесения изменений в базу знаний обучающей ЭС, а также создания собственных оригинальных вариантов ЭС, основанных на знаниях и личном опыте преподавателя. В перспективе потребуется интеграция ЭС с широко

применяемыми функциональными пакетами прикладных программ.

Литература

1. Стандарт на ПЭВМ // Информатика и образование. 1988. № 1.
2. Курдюмов Г. Л. Системы ручного управления непрерывно меняющимися сигналами // Микропроцессорные средства и системы. 1986. № 2.
3. Маслов А., Таиров О., Труш В. Физиолого-гигиенические аспекты использования персональных ЭВМ в учебном процессе // Информатика и образование. 1987. № 4.
4. Козловский С. Узловые проблемы оценки ка-

чества изображения на экране дисплея // Информатика и образование. 1988. № 1.

5. Значительное улучшение технико-экономических показателей компьютера фирмы Tandem // Электроника. 1986. № 8.

6. Д. Ван Тассел. Стиль, разработка, эффективность, отладка и испытание программ. М.: Мир. 1985.

7. Экспертные системы. Принципы работы и примеры / Под ред. Р. Форсайта. М.: Радио и связь. 1987.

8. Блишун А., Симонов М., Шапиро Г. Инструментальные программные средства формализации профессиональных знаний учителя // Информатика и образование. 1987. № 6.

С. ХРИСТОЧЕВСКИЙ

Перспективный компьютер для сферы образования

В последнее время многие специалисты ведут разговоры о том, каким должен быть школьный компьютер. Большинство оперируют сразу конкретными характеристиками ПЭВМ, никак их не обосновывая или утверждая, что для сферы образования все должно быть самым лучшим. Думается, что такой подход порочен в своей основе. Необходимо сначала определить, какие именно характеристики ПЭВМ необходимы для использования в учебном процессе, во сколько обойдется компьютер с такими параметрами для страны в целом, т. е. настолько ли мы богаты, чтобы иметь во всех образовательных учреждениях самые лучшие компьютеры, или все-таки целесообразно воспользоваться критериями функционально-стоимостного анализа.

К сожалению, о таком всестороннем анализе у нас в печати не сообщалось. Автор предлагает для обсуждения свои предложения, подчеркивая, что он не является специалистом по методике преподавания. Поскольку первые школьные ПЭВМ («Корвет», УКНЦ) уже запущены в производство, целесообразно обсудить перспективные ПЭВМ, которые придут на смену существующим.

А нужен ли специальный компьютер для сферы образования? Следует признать, что невозможно, а скорее нецелесообразно разрабатывать для нее универсальный компьютер, способный работать с базами данных, текстом, электронными таблицами, управлять лабораторным (тренажерным) оборудованием, входить в состав тренажеров и т. д., хотя и такая точка зрения существует и ее можно аргументировать снижением затрат

на эксплуатацию установленных систем (унификация установки, обслуживания, ремонта и т. п.). Но все же, учитывая массовость производства (сотни тысяч, даже миллионы), целесообразно провести некоторое разграничение и выделить соответствующие наиболее массовые применения, что позволит сэкономить значительные средства при изготовлении компьютеров, отвечающих указанным применениям.

Сейчас в школах устанавливаются КУВТ с одним преподавательским местом и 10—15 ученическими. Как будет меняться положение в будущем?

КУВТы в основном используются для овладения компьютерной грамотностью. В то же время в недалеком будущем (мнение академика А. П. Ершова) до 30 % контактного времени, выделяемого на отдельного учащегося, будет тратиться на орудийные применения.

При наличии в школе двух девятых и двух десятых классов один КУВТ будет загружен только по основам информатики до 408 ч в год, или около 10 ч в неделю. Учтем необходимость подготовки преподавателей и факультативов — и для остальных предметов останется около 2 ч в неделю на каждый (для всех классов), что, конечно, недостаточно для систематической работы по каждому предмету. А ведь в дальнейшем обязательно будет осуществлен переход к схеме один ученик — один компьютер, так как работа учеников парами нецелесообразна, особенно с медицинской точки зрения.

Видимо, в дальнейшем в школах появится

второй КУВТ, который будет использоваться на уроках физики, химии, профессиональной ориентации и т. д. Его основными функциями будет управление работой лабораторного оборудования, простые вычисления (естественные предметы). Такие же КУВТы найдут применение в учебно-производственных комбинатах, ПТУ, для профессиональной подготовки студентов. Для входящих в них ПЭВМ характерны такие свойства, как управление лабораторным оборудованием и наличие динамической графики для моделирования процессов. По всей видимости, именно эти ПЭВМ будут использоваться и в тренажерных системах профессиональной подготовки, но для этого потребуется более высокое быстродействие.

Практически одновременно должны появиться отдельные ПЭВМ (2—5), используемые в качестве «электронных досок», с помощью которых преподаватель будет показывать определенные модели с изменяемыми параметрами, иллюстрировать тематику урока, управлять оборудованием класса (видеомагнитофон, диа- и кинопроекторы, шторы и т. д.).

Кроме того, в каждом учебном заведении появятся интерактивные видеосистемы (типа интерактивных энциклопедий) на основе профессиональных ПЭВМ. Они будут использоваться для информационно-справочной, исследовательской работы школьников, студентов и педагогов при подготовке отдельных тем, требующих углубленного знания первоисточников и документов.

Используемые компьютеры должны составлять единое семейство. На нижнем уровне (орудийное применение) — несложная модель, предназначенная для работы с текстами, контроля, практической работы в простейших случаях и т. д. Второй уровень (компьютерная грамотность) обеспечивается моделями, обладающими большой емкостью ОЗУ, внешними устройствами, электронными дисками, дополнительными устройствами ввода-вывода («мышь», трекболл, устройство ввода графической информации и т. д.). Третий уровень (управляющие и моделирующие, тренажерные ЭВМ) — возможность подключения дополнительных устройств, аналого-цифровых преобразователей, учебных роботов и т. д. Четвертый уровень (интерактивные видеосистемы, «электронные доски») — мощные профессиональные ПЭВМ.

В представленной схеме предпочтение отдается персональным ЭВМ. Возможны ли другие варианты? Да. Ранее в образовании использовалась одна центральная ЭВМ, к которой подсоединялись выносные дисплеи. Однако такая модель оказалась дорогой из-за больших накладных расходов на операцион-

ную систему, необходимости специального обслуживания ЭВМ, низкой надежности (при выходе из строя любого элемента полностью останавливается работа), да и не совсем удобно работать с ней преподавателю. В настоящее время для целей образования во всем мире отдается предпочтение простым ПЭВМ или ПЭВМ, объединенным в КУВТ, а не более мощным многотерминальным, но в будущем потребуются провести дополнительный анализ.

Широкое использование в школе мощных ПЭВМ (более дорогих, с большими быстродействием, ОЗУ и т. д.) возможно. Обоснованием этому служит намечаемая во многих странах, а кое-где уже идущая замена 8-разрядных машин на 16-разрядные, с большими возможностями (типа IBM PC). Однако в таком случае (по экономическим причинам) придется сохранить систему межшкольных кабинетов еще надолго. В то же время возможности этих ПЭВМ представляются избыточными для некоторых целей обучения. Так, большинство обучающих программ, используемых в настоящее время в СССР, требуют сравнительно малого объема ОЗУ для работы (до 64К).

Суммируя, перечислим основные подклассы применений:

- простые контролирующие системы (применение для проверки знаний, по схеме АОС, на уроках, самостоятельных занятиях и т. д.);
- управление лабораторным оборудованием; тренажеры, моделирующие системы, развитие профессиональных навыков;
- внедрение компьютерной грамотности на низшем уровне (работа с «активной» информацией, преобразование данных);
- режим «электронной доски» (моделирование процессов, наглядное представление);
- применение в преподавании гуманитарных и других предметов (ИПС, базы данных, базы знаний, системы с элементами искусственного интеллекта);

- разработка программных средств для сферы образования (инструментальные комплексы).

На первый план выходит задача создания такой архитектуры ПЭВМ, которая соответствует концепции непрерывного образования, способна обеспечить преемственность в цепочке *школа — ПТУ — ВУЗ — профессиональная деятельность*. Число моделей компьютеров, применяемых в сфере образования, необходимо свести к одной-двум, чтобы не расплывать силы и средства и улучшить возможности для обмена опытом.

Школьный компьютер должен удовлетворять некоторым специфическим требованиям, определяемым задачами применения. Кроме того, необходимо учитывать требова-

ния функционально-стоимостного анализа, а именно как то или иное свойство будет отражаться на стоимости компьютера (с учетом изготовления миллионными тиражами).

Каковы же эти требования?

1. Надежность (не менее 10—20 тыс. ч).
2. Простота использования.
3. Простота и доступность интерактивного взаимодействия (средства манипулирования текстовой и графической информацией).
4. Возможность моделирования процессов (динамическая графика, быстрая смена графических планов, закрашка экрана, различных фигур).
5. Возможность подключения демонстрационных устройств с большим экраном.
6. Возможность подключения лабораторного оборудования и управления им с использованием таймера.
7. Наличие цветных мониторов на РМП.
8. Наличие звукогенераторов и индивидуальных средств прослушивания на каждом рабочем месте.
9. Быстродействие ПЭВМ должно быть достаточно для обеспечения эффективной работы в классе (точных оценок нет, но можно ориентироваться на ПЭВМ, применяемые в сфере образования во всем мире, — около 500 тыс. — 1 млн. операций в секунду). Аналогичные требования к ВЗУ, локальной сети и т. д.
10. Все РМУ должны предусматривать подключение гибких дисков или содержать электронные диски.
11. На РМП необходимо наличие жестких дисков.
12. Целесообразно предусмотреть ряд ПЗУ для типовых применений ПЭВМ в учебном процессе.
13. Печатающие устройства на РМП должны обеспечивать быстрый и бесшумный вывод информации, в том числе и графической.

14. Должны быть развитые средства архивизации, т. е. устройства для долговременного хранения (типа магнитных лент).

15. Структура РМП и РМУ должна соответствовать эргономическим требованиям и обеспечивать для пользователей комфортную среду.

16. Локальная сеть должна обеспечивать весь набор функциональных требований к локальной сети и обеспечивать организацию индивидуальной и коллективной деятельности.

17. Все рабочие места должны быть взаимно независимы и не нарушать работоспособность друг друга.

18. РМП выполняет функции сетевого процессора только в фоновом режиме, предоставляя учителю возможность работать независимо.

19. Должны быть предусмотрены средства подключения к внешней информационной сети.

Следовательно, необходима ПЭВМ блочной структуры (или семейство ПЭВМ), которая бы учитывала спектр от простейших применений до мощных инструментальных комплексов для разработки прикладного программного обеспечения и позволяла бы при необходимости наращивать уже установленную ПЭВМ до требуемой конфигурации в зависимости от применения.

Именно с этой точки зрения следует подготовить технико-экономическое обоснование для разработки перспективных компьютеров, учитывающее их стоимость, предполагаемый тираж, эффективность в сфере образования. Неотъемлемой его частью должны стать обоснованные психолого-педагогические требования к компьютерам и социально-экономический анализ последствий их использования.

А. МАТЮШКИН-ГЕРКЕ

Каким же быть школьному компьютеру?

В последнее время стали появляться статьи, авторы которых критикуют Минрадиопром и Минэлектронпром за неполадки с выпуском школьных ЭВМ «Корвет» и УКНЦ. Не собираюсь защищать изготавителей вычислительной техники. Грехов у них пока что еще многовато, и в долгу они не только перед школой. Но этот случай требует особого

разговора, ибо далеко не безупречны тут и сами критики, а выход следует искать совсем не там, где они это предлагают.

Должны ли школьники научиться видеть в ЭВМ одно из средств решения прикладных проблем, или же им надо овладеть только чисто внешней, «манипуляционной» стороной дела (умение вести простейший диалог с

операционной системой, работать с тем или иным текстовым или графическим редактором, создавать простенькие структурированные программы)? Из этих двух подходов, определяющих весь характер вновь вводимой учебной дисциплины, создатели минпросовской модели школьного курса информатики выбрали последний. Оппонентов проигнорировали. В результате вместо общеобразовательного курса была создана миникопия курса профессионального. А это, в свою очередь, сразу же потребовало и соответствующего (профессионального!) технического обеспечения. Вот откуда появились в качестве кандидатов на роль школьного компьютера сначала «Агат», а затем «Корвет» и УКНЦ, вот почему безоговорочно отмечается все, что классом ниже этих моделей. Авторитетом титулов и должностей утверждалась вера в то, что иного решения и быть не может.

Ну, а если разобраться по существу?

Во-первых, эта техника профессионального уровня на школьных уроках будет использоваться с КПД, в среднем не превышающим 1—2 %. Не слишком ли расточительно при острейшем ее дефиците?!

Во-вторых, концепции, заложенные в эту технику, отнюдь не самые новые. Она устареет окончательно, пока будет добираться до школы.

В-третьих, техника эта дорогая, в-четвертых — ненадежная, в-пятых — она просто не позволяет решить стоящие перед школой задачи: ведь один дисплейный класс на школу (а когда это еще будет!) — это 3 мин в день (!) работы за дисплеем на одного ученика. Для курса информатики в его теперешнем объеме, конечно, хватит, а вот как быть с переводом на «электронные рельсы» других дисциплин? Ограничиться одним-двумя показательными уроками? Ждать второй половины XXI века?

Для того чтобы о компьютеризации можно было говорить всерьез, школу надо насытить вычислительной техникой и поддерживать ее в рабочем состоянии. Можно ли это сделать, ориентируясь только на профессиональные ЭВМ? Сколько их нужно иметь в школе? Кто будет столь громадный парк ПЭВМ обслуживать и ремонтировать? С какой периодичностью его обновлять? Сколько все это будет стоить, кто и из каких средств будет платить?

Похоже, что при определении технической политики над этими вопросами предпочли не задумываться, иначе от минпросовской модели курса просто-напросто пришлось бы отказаться!

Вычислительная техника в школе должна быть разная. Разная, образно говоря, не «по горизонтали», а «по вертикали». Так, про-

граммируемые микрокалькуляторы (с презрением отвергаемые «компьютерными снобами») на самом деле позволяют «закрыть» больше половины потребностей теперешнего курса информатики, весьма успешно могут быть использованы в математике, физике, других дисциплинах. В иных ситуациях они оказываются даже предпочтительнее ПЭВМ, ибо все происходящее внутри такого калькулятора не маскируется от ученика ни операционной системой, ни транслятором, ни какими-либо другими программами. Опыт показывает, что микрокалькуляторный этап в освоении вычислительной техники сам по себе оказывается исключительно полезным и заметно облегчает изучение последующего материала. Еще шире, чем у программируемых калькуляторов, возможности у так называемых бытовых компьютеров.

Принципиальные особенности применения вычислительной техники для решения прикладных проблем вполне можно донести до учащихся, оставляя и на таком, мягко говоря, не слишком высоком техническом уровне. Сущность этих принципов мало зависит от быстро сменяющих друг друга поколений ЭВМ и подходов к построению их программного обеспечения. А вот работа пользователя с операционными системами, сервисными и проблемно-ориентированными программными средствами уже через несколько лет станет совершенно иной. Отсюда с очевидностью следует вывод о том, где должен располагаться центр тяжести в обучении.

Обидно сознавать упущенные возможности. Ведь такой простейшей техникой уже сейчас могла бы быть обеспечена практически каждая школа, если бы этому способствовали, а не препятствовали в бывшем Минпросе. Техника эта на 1—2 порядка дешевле профессиональных ПЭВМ. А действительные потребности школ в этих последних столь велики, что здесь-то как раз и можно было бы ограничиться несколькими уроками на базе межшкольного центра. Именно через такие центры только и можно будет знакомить учащихся с новыми достижениями информационной техники и технологии. И делать это надо будет опять-таки широким фронтом, не замыкаясь на один какой-либо класс вычислительных средств. По целому ряду причин школьным кабинетам такое оперативное отслеживание новинок не по плечу, да и не в этом их назначение.

В качестве чрезвычайной оперативной меры сейчас следовало бы снизить заявки на «Корветы» и УКНЦ до уровня потребностей межшкольных центров информатики, зато постараться побыстрее обеспечить школы программируемыми калькуляторами и простейшими компьютерами, обратив главное

внимание на надежность всей выпускаемой техники. Тяжелое решение, но другого-то выхода нет!

Дальнейшую же стратегию надо определять исходя из того, что мы хотим и что можем делать в школе, какой вообще собираемся ее сделать, и уже на этой основе решать, какую вычислительную технику и в каких количествах там надо иметь. Это — естественная логика: цель — задачи — средства их решения. Начинать же с «требований к школьному компьютеру», взятых с потолка, как это, к сожалению, произошло, нелепо! Здесь уже под заранее заданные средства приходится подбирать задачи, которые с помощью этих средств можно решить.

В ближайшее время реально, во-первых, обеспечить изучение курса основ информатики и вычислительной техники в варианте, эквивалентном, например, описанному в [1] и, во-вторых, начать эпизодическое (!) применение компьютеров в учебном процессе по другим дисциплинам.

На следующем этапе — при переносе начала изучения информатики в средние классы и переходе к систематическому использованию вычислительной техники в преподавании отдельных дисциплин (прежде всего математики и физики) наибольшие трудности будут заключаться отнюдь не в создании и развертывании производства той или иной модели школьной ЭВМ. Главной проблемой станут кардинальный пересмотр содержания и методики преподавания этих дисциплин и, разумеется, соответствующая фундаментальная переподготовка учителей.

Для нового курса физики наиболее характерным отличием от существующего будет громадное расширение круга изучаемых процессов (протекающих, как правило, не равномерно и не равноускоренно), широкое использование методов математического моде-

лирования, качественно иной, более высокий, уровень теоретических построений. В математике должны будут появиться элементы математической статистики и обработки результатов наблюдений, простейшие подходы к исследованию функций, численному решению разнообразных уравнений и систем. Ясно, что такое построение этих курсов будет предъявлять достаточно скромные требования к уровню используемых средств вычислительной техники. Во всяком случае для паники и суматошных поисков какой-то особой модели школьного компьютера никаких оснований тут нет.

Неизбежный последующий переход к фронтальному использованию «высокоинтеллектуальных» обучающих систем в настоящее время, к сожалению, не опирается на сколько-нибудь существенные методологические и технические разработки, без которых обсуждение конкретных требований к технике становится бессмысленным. Проблема эта требует отдельного и весьма обстоятельного разговора — здесь отметим лишь, что решаться она будет с помощью техники, весьма отличной от «Корвета» и УКНЦ.

Нынешнее бедственное положение с вычислительной техникой для школы, равно как и другие провалы в деле компьютеризации образования, обусловлены тем, что все узловые решения выработывались и принимались без привлечения широкой научно-педагогической общественности. Хотелось бы надеяться, что вновь созданный Госкомитет по народному образованию сумеет выправить положение.

Л и т е р а т у р а

1. Матюшкин-Герке А. А. Каким быть школьному курсу информатики // Информатика и образование. 1987. № 6.

УВАЖАЕМЫЕ ЧИТАТЕЛИ

Весной этого года подведены итоги посвященного 70-летию Октября смотра-конкурса научно-технических и производственных журналов на лучшее освещение вопросов ускорения научно-технического прогресса. В конкурсе, организованном Московской организацией Союза журналистов СССР, Всесоюзным советом научно-технических обществ, Всесоюзным обществом «Знание» и Главным комитетом ВДНХ СССР, принимали участие и педагогические журналы.

Редколлегия и редакция журнала «Информатика и образование» рады сообщить, что по результатам смотра-конкурса наш журнал награжден второй премией президиума Правления Всесоюзного общества «Знание». Первая и третья премии по линии общества «Знание» присуждены соответственно журналам «Наука и жизнь» и «Радио».

И. КУЗЬМИН,
канд. техн. наук,
В. КОЧУРОВ,
канд. пед. наук
О. КОЧУРОВА,
Ленинград

Реальный уровень компьютерной грамотности педагогических кадров: ничто или нечто?

Семь раз отмерь, один раз отрежь

Основная нагрузка компьютерного всеобуча педагогических кадров ложится на систему повышения квалификации, выполняющую функции компенсации недостающих знаний и умений по информатике, адаптации педагога в среде компьютеризованного обучения, развития его творческого потенциала, формирования информационной культуры. Полноценная реализация этих функций выдвигает задачу получения достоверной информации об уровне компьютерной грамотности (КГ) педагогов. Все, о чем пойдет речь в этой статье, имеет отношение к педагогам, не специализирующимся в преподавании основ информатики, но желающим или вынужденным использовать вычислительную технику в своей профессиональной деятельности.

Итак, каков сегодняшний уровень КГ педагогических кадров и каким он должен быть для успешного решения задач компьютеризации образования? Чтобы ответить на этот вопрос, надо определить понятие «уровень компьютерной грамотности» и разработать методику измерения этого уровня.

Известно, что концепция уровней КГ еще не утвердилась даже в школьной информатике. Мы попытаемся выдвинуть и обосновать свои исходные позиции для понимания структуры компьютерной грамотности педагога и с помощью специальной методики диагностировать ее уровень.

В основе нашей позиции лежит представление о развитии компьютерной грамотности, идущем по спирали от знания элементов информатики к всестороннему овладению средствами вычислительной техники, творческому осмыслению способов ее реализации в системе учебно-воспитательного процесса.

Следуя этому представлению, мы выделили три основных уровня КГ, качественно отличающихся один от другого: уровни элементарной, функциональной и системной грамотности. Переход с одного уровня КГ на другой подразумевает диалектическое отрицание, или «снятие» предыдущего. Так, теоретические знания, преобладающие на уровне элементарной грамотности, диалектически отрицаются, переосмысливаются в процессе практической деятельности педагога, и тем самым осуществляется переход на качественно иной, более высокий уровень — функциональной грамотности. При этом реализуются психологические механизмы экстерниоризации (переноса во внешнюю среду) знаний и умений предыдущего уровня в практическую деятельность по использованию вычислительной техники. В дальнейшем синтез практических умений и углубившихся теоретических познаний приводит к интериоризации вновь приобретенных способов деятельности и преобразованию их в качественно более совершенные внутренние структуры сознания, являющиеся основой творческого, системного применения ЭВТ в педагогической деятельности. При этом формируется новый стиль мышления — компьютерное мышление как важнейшее новообразование в личности педагога на уровне системной грамотности. Основным проявлением этого уровня КГ можно считать способность свободно, умственно и адекватно использовать ЭВМ в педагогических системах.

Рассмотренные уровни достаточно полно, с нашей точки зрения, отражают вероятную динамику развития компьютерной грамотности. Однако спецификой переживаемого момента является фактическое отсутствие в среде педагогов научных представлений в

области информатики. В подобных условиях знания абсолютного большинства педагогов могут оказаться ниже уровня элементарной КГ, что делает нереальной попытку диагностировать компьютерную грамотность на основе концепции трех основных ее уровней. В связи с этим, нам показалось оправданным введение двух дополнительных уровней, обладающих достаточной дифференцирующей способностью в реальных условиях дефицита знаний и умений по информатике. Это уровни: предварительного знакомства с ЭВТ (соответствует представлению о вычислительной технике, предлагаемому средствами массовой информации) и компьютерной осведомленности (соответствует представлению о ВТ, формируемому научно-популярной литературой). Вводимая таким образом искусственная избыточность структуры КГ, позволяет выполнить задачи диагностики компьютерной грамотности в реальной совокупности педагогических кадров.

Исходя из рассмотренной теоретической позиции, каждому из уровней КГ может быть дана следующая словесная характеристика.

1. На уровне знакомства педагог не осознает и, как правило, не признает целей компьютеризации образования; имеет разрозненные, случайные и в основном ненаучные знания о вычислительной технике. Ее образ большей частью сформирован средствами массовой информации, поэтому упоминание об ЭВМ вызывает только простейшие ассоциации типа «дисплей», «быстро вычисляет», «имеет большое будущее» и т. п. У педагога полностью отсутствует представление о возможности применения ЭВТ в учебном процессе, не сформированы умения и навыки работы с ЭВМ.

2. Уровень осведомленности характеризуется пониманием наиболее общих целей компьютеризации образования; педагог имеет некоторые фрагментарные сведения о структуре, физических основах и принципах действия ЭВМ, знает наиболее яркие примеры ее применения, ознакомлен с общими перспективами развития и вредения ЭВТ, владеет определенным, хотя и весьма ограниченным, тезаурусом по информатике. Умения и навыки работы с ЭВМ не распространяются дальше использования некоторых простейших программ стандартного обеспечения учебного процесса.

3. Элементарная грамотность определяется осознанием и внутренним принятием целей компьютеризации образования, наличием у педагога комплексного знания об элементах и структуре ЭВМ, ее физических основах, принципах действия и ведущих направлениях развития. Педагог хорошо знает существую-

щие и перспективные области применения ЭВТ, в том числе в системе образования. Имеет понятие об основах алгоритмизации и программирования, умеет разрабатывать простые алгоритмы, составлять и отлаживать соответствующие им программы. Имеет навыки работы с педагогическими программными средствами, пакетами прикладных программ, ориентированных на управление учебно-воспитательным процессом.

4. Функциональная грамотность педагога обусловлена осознанием стратегических и тактических задач компьютеризации образования, формированием психологической установки на всестороннее применение ЭВТ в учебном процессе. Педагог обладает развитыми умениями работы с ЭВМ, при этом он способен разрабатывать весьма сложные разветвляющиеся алгоритмы и кодировать их на одном или нескольких языках программирования; глубоко и полно знает материальную базу и программное обеспечение ЭВМ своего учебного заведения; способен самостоятельно разрабатывать программные средства для дополнительного обеспечения собственной педагогической деятельности; творчески и со знанием дела использует стандартное программное обеспечение. Постоянно стремится к расширению и углублению своих знаний и умений в области информатики.

5. На уровне системной грамотности компьютеризация образования приобретает для педагога глубокий личностный смысл, он обладает целостной системой знаний и умений в области информатики, позволяющей ему свободно ориентироваться в мире ЭВТ, без труда переходить от работы на одной ЭВМ к работе на другой, от одного языка программирования — к другому. Педагог глубоко осознает перспективы информатизации общества, грамотно реализует стратегию и тактику компьютерного обучения, разрабатывает принципиально новые и постоянно совершенствует действующие программные средства в своей области деятельности, способен оказать реальную помощь в разработке и применении подобных средств в деятельности коллег.

Если согласиться с предложенной структурой КГ, то можно перейти к следующему этапу — попытке измерить реально существующий уровень КГ педагогов. Для выполнения этой задачи мы разработали специальную тестовую методику с дифференциальной шкалой оценивания (см.: О программном контроле знаний/Профтехобразование. 1983. № 5. С. 61—62). Главной ее особенностью является близость к наиболее достоверной форме — индивидуальному опросу со свободно конструируемым ответом. При этом

имеется возможность выявлять ложные представления педагогов, констатировать отсутствие либо наличие истинных знаний и осуществлять их оценку с помощью теста-лестницы, ступени которой соответствуют фиксированным уровням КГ. Оценка знаний на каждом уровне КГ производится по десятибалльной шкале с учетом «слепой зоны» теста.

Измерение уровня КГ произведено на представительной выборке педагогов профтехобразования — слушателей курсов повышения квалификации. Среди них были директора ПТУ, их заместители по учебно-производственной работе и мастера производственного обучения.

В результате измерения оказалось, что слушатели достигли в среднем лишь самой нижней границы уровня знакомства.

Как же распределились педагоги по уровням компьютерной грамотности? Есть ли какие-нибудь отличия в этом распределении для групп, однородных по признакам пола, возраста, занимаемой должности и т. п.?

Анализ данных диагностики показал, что первого уровня грамотности достигли только 60 % от общего числа слушателей, из них: не поднялись выше этого уровня — 64 %, достигли второго уровня — 33 %, вышли на третий уровень — 3 %. Выше третьего уровня — элементарной грамотности — не поднялся никто. При этом обнаружены существенные различия в компьютерной грамотности и между отдельными профессионально-демографическими подгруппами педагогов. Вот наиболее характерные из таких расхождений.

Женщины грамотнее мужчин. Если количество ложных представлений и нулевых знаний в обеих группах педагогов фактически одинаково (18 % и 19 %), то основная масса мужчин еще не достигла уровня знакомства (43 % против 30 % у женщин), а количество мужчин на уровне осведомленности составило 16 %, что ощутимо меньше соответствующего количества женщин.

«Молодые» грамотнее «старших». Анализ распределения оценок по возрасту показал сходство между возрастными группами 40—50 и старше 50 лет. В то же время педагоги моложе 40 лет гораздо лучше справились с заданиями теста. Так, уровня осведомленности достигли более 34 % «молодых» против 9 % «зрелых» и «пожилых». Основная масса старшей возрастной группы (около 50 %) не справилась даже с заданиями уровня знакомства, в то время как в «младшей» возрастной группе этот показатель составил всего 30 %.

Заместители грамотнее начальников. Распределение педагогов в зависимости от зани-

маемой должности оказалось идентичным для групп директоров и мастеров и существенно отличающимся для группы заместителей директоров. Так, число педагогов с ложными представлениями в первых двух группах в среднем вдвое больше, чем в третьей (соответственно 20 % и 10 %). Основная часть директоров и мастеров — около 46 % — показали знания хуже уровня знакомства, в то время как среди заместителей директоров эта цифра составила лишь 29 %. Большая часть последних достигла уровня осведомленности — 38 % против 12 % в группах директоров и мастеров.

Чем дольше работаешь в должности — тем лучше. Определенное отличие в распределении педагогов по уровням КГ обнаружено в зависимости от стажа работы в занимаемой должности. Среди слушателей, занимающих должность более 10 лет, выявлен существенно больший процент достигших уровня знакомства — 50 % против 36 % у слушателей со стажем работы в должности менее 10 лет. В то же время отмечено некоторое преобладание последних среди слушателей, достигших второго и третьего уровня КГ.

Чем дальше от техники — тем хуже. Анализ распределения по признаку «вид образования» (техническое, педагогическое, гуманитарное) показал наличие существенных различий между группами педагогического и гуманитарного образования, с одной стороны, и технического образования — с другой. Слушатели с техническим образованием значительно опережают своих коллег с педагогическим и гуманитарным образованием. Если основная масса последних еще не достигла уровня знакомства — таких слушателей среди «педагогов» и «гуманитариев» около 60 %, — то группа с техническим образованием в своей большей части (40 %) уже имеет соответствующие этому уровню знания и умения. Причем 28 % «техников» находится на уровне осведомленности против 10 % «педагогов» и 0 % «гуманитариев». Помимо этого существенного различия отмечено в целом более высокое качество знаний в группе с педагогическим образованием по сравнению с группой «гуманитариев». Особенно весомо это отличие в количестве 12 % у «техников».

Долго учился — хоть что-нибудь да узнал. Изучение распределения педагогов в зависимости от уровня образования (высшее, среднее) выявило различие в диапазоне оценок КГ. Распределение слушателей с высшим образованием более равномерно, чем в группе со средним, где отмечен резкий спад ко второму уровню КГ. Так, 50 % слушателей со средним образованием достигли

уровня знакомства (в группе с высшим образованием — 36 %) и только 5 % — уровня осведомленности (в группе с высшим образованием — 23 %). Число педагогов каждой из групп, не справившихся с заданиями первого уровня КГ, а также выявивших в основном ложные представления, оказалось фактически одинаковым (соответственно около 42 % и 40 %).

Таким образом, наилучшая структура распределения педагогов по уровням КГ выявлена в группах: по полу — среди женщин; по возрасту — моложе 40 лет; по должности — у заместителя директора ПТУ; по виду образования — среди слушателей с техническим образованием; по уровню образования — с высшим образованием.

Интересными, с нашей точки зрения, оказались данные о силе влияния некоторых характеристик педагогов на уровень их компьютерной грамотности. Эти данные — коэффициенты влияния — получены методом регрессионного анализа и представлены ниже в виде линейной зависимости:

$KГ = 0,112x_1 + 0,260x_2 - 0,163x_3 - 0,144x_4$,
где x_1 — стаж работы в занимаемой должности.

x_2 — уровень образования,

x_3 — возраст,

x_4 — вид образования (по нарастанию гуманитарности),

КГ — уровень компьютерной грамотности.

Из формулы видно, что уровень образования и стаж в должности оказывают «стимулирующее» влияние на компьютерную грамотность педагога, в то время как возраст и вид образования представляют своеобразный «механизм торможения». При этом возраст и уровень образования — наиболее «влиятельные» характеристики педагога в его отношениях с компьютером. Однако заметим, что это влияние действует только в узком кругу рассмотренных признаков, которые объясняют лишь 10 % реальных различий в уровне компьютерной грамотности педагогов.

Наконец процедура диагностики осуществлена. Какая же из альтернатив, вынесенных в заголовок статьи, оказалась ближе к истине: ничто или нечто?

Можно сказать «ничто» и оказаться недалеко от истины — среднестатистический педагог имеет знания, эквивалентные скромным требованиям первого уровня КГ: нет понимания целей и задач компьютеризации образования, знания разрозненны, случайны и в основном ненаучны.

Можно сказать «нечто» и тоже не ошибиться, помня значительную амплитуду разброса в уровнях грамотности: от укоренившихся ложных представлений и до вполне качественных, хотя и не очень глубоких, знаний и умений.

Это позволяет нам предложить некоторые рекомендации для системы повышения квалификации:

1. При подготовке «инфраструктуры» учебного процесса для каждой категории потенциальных слушателей целесообразно установить необходимый и достаточный уровень компьютерной грамотности.

2. Разработать систему учебно-тематических планов и программ, призванную обеспечить многоальтернативность обучения как по его глубине (многоступенчатость), так и по направленности содержания. Прудумать разнообразие форм и методов повышения квалификации.

3. Реализовать не только «внутреннюю», но и «внешнюю» адаптивность педагогических программных средств. Если первая должна быть заложена в самом средстве и обеспечивать индивидуализацию стратегии обучения, то вторая может выражаться в многообразии видов ППС, необходимым для их дифференцированного применения в зависимости от групповых особенностей педагогов. При этом, естественно, достижение индивидуальной адаптивности существенно упрощается.

4. Осуществлять перед началом обучения входной контроль знаний и умений слушателей, производить анализ его результатов и на основании этого определять индивидуальные планы занятий. Таким образом, исходный уровень КГ должен выступать одним из важнейших критериев дифференциации слушателей.

В зависимости от результатов входного контроля может быть избрана одна из трех основных стратегий обучения: формирование компьютерной грамотности «от нуля»; совершенствование уже имеющейся базы знаний и умений; перестройка ненаучных, ложных представлений и установок, сформированных у слушателей ранее в силу различных причин. Вне всякого сомнения, последняя задача является наиболее сложной и ответственной.

В заключение повторим, перефразируя, эпиграф статьи, который отражает нашу позицию: прежде чем обучать педагога информатике, надо измерить уровень его компьютерной грамотности — если и не семь, то хотя бы один раз.

Е. ПОЛАТ

Телекоммуникации в системе образования

Выпускники современных общеобразовательных школ в массе своей обучены преимущественно репродуктивным формам мышления. Творческий потенциал молодого поколения, его способность самостоятельно мыслить не раскрываются в школе полностью. Этот факт вынуждены констатировать ученые многих стран.

В последние десятилетия реформы образовательных систем объективно обусловлены возникшим противоречием между потребностями современной экономики, технологии, «новыми методами создания и экспоненциального нарастания информации, с одной стороны, и отставанием методов усвоения информации, с другой» [1].

Для того чтобы преодолеть это противоречие, необходимы радикальные меры. В первую очередь надо обеспечить свободный доступ учителей и учащихся к новейшей информации, индивидуализацию и дифференциацию обучения на всех ступенях образования, особенно в общеобразовательной школе; решать проблему эффективного управления учебно-познавательной деятельностью учащихся, причем эта деятельность по самой своей сути все более должна приобретать характер научного, творческого познания.

Разумеется, учитель даже при наличии самых совершенных учебников решить эту проблему без специальных технических средств сегодня уже не в состоянии. На помощь приходят новейшие электронные средства, средства информационной технологии, и в первую очередь компьютер. Изучение опыта развитых капиталистических стран показывает, что эти средства способны оказать поистине революционное воздействие на все компоненты обучения: его содержание, методы, организационные

формы и в конечном итоге даже на цели, поскольку появляется возможность обеспечить усвоение значительно большего как по объему, так и по сложности учебного материала, при этом затрачивается меньше усилий.

Анализ зарубежного опыта позволяет не только критически осмыслить пройденный различными странами путь, но и отобрать из этого опыта то рациональное, что может оказаться полезным и продуктивным в отечественной дидактике, в нашей системе образования.

Таким образом, кардинальное преобразование учебно-материальной базы общеобразовательной школы становится ключевой проблемой реформы.

В данной статье рассказывается об использовании в системе образования средней и высшей школы телекоммуникаций. Однако прежде, видимо, стоит напомнить читателю, что понимается под информационной технологией и телекоммуникацией.

К средствам информационной технологии относятся помимо персонального компьютера компьютерные сети, телетекст (телетекст и выюдейта), телеконференции (видеоконференции и аудиоконференции) видеосистемы, в том числе интерактивные, радиовещание, спутниковая связь. Обо всех этих средствах говорилось в [2]. Телекоммуникации — понятие более узкое. Это также элемент информационной технологии. Под телекоммуникацией понимается любая передача или прием различных знаков, сигналов, образов, звуков, в общем, любой информации по проводам, оптическим путем, по радио или с помощью любой другой электромагнитной системы [3]. Соединение телекоммуникации и информатики, т. е. телекоммуникацион-

ных технологий, оборудования в единую интегрированную систему дало начало новой отрасли знания — телематике.

За последние годы средства телекоммуникаций, телематики находят все более широкое распространение в системе образования средней и высшей школы различных стран мира. Система телекоммуникаций организуется на базе компьютерных сетей: персональные компьютеры школы, района, города соединяются в компьютерную сеть, которая, в свою очередь может подсоединяться к компьютерным сетям другого района, города, региона и даже другой страны. Передача информации может идти либо по телефонным линиям с помощью специального устройства — модема*, либо по телевизионным каналам. Такие системы находятся в настоящее время достаточно широкое распространение в США, Великобритании, Японии, Франции, Канаде. Так, в США для этих целей создана специальная компьютерная система «Бюллетен борд систем» («Би би эс»), которая позволяет осуществлять поиск и передачу информации на расстояние.

Телекоммуникации привлекают все больше внимания учащихся, преподавателей, студентов. Школы, университеты, отдельные абоненты подключаются к общенациональным или региональным системам, находя при этом все более широкие возможности интеграции телекоммуникаций в учебные программы, хотя это и требует не только освоения нового технического средства, но и разработки новых программных средств. Для того чтобы стать активным участником телекоммуникаций, необходимо компьютер превратить в терминал, подсоединив к нему модем и, разумеется, используя соответствующие программные средства. Таким образом оказывается возможным, получив доступ к местным компьютерным системам (например, в США — «Би би эс»), передавать собственную информацию в виде текстов, рисунков, графиков, получать электронную почту, активно участвовать в телеконференциях, вести поиск нужной информации по самым различным направлениям науки, техники, коммерции, индустрии развлечений. Специальные программные средства организуют диалог через модем с удаленными компьютерами. Разумеется, время телефонной связи с ними ограничено. Поэтому предварительная подготовка и «отработка» информации, подлежащей передаче на рас-

стояние, ее загрузка приобретают чрезвычайно большое значение.

Каким образом и в какой форме телекоммуникации используются в системе образования? Это могут быть.

контакты групп учащихся одной школы с учениками другой;

участие в региональных или международных телеконференциях;

абонентная информационная служба, в том числе интерактивная (вьюдейта) по типу созданных во многих странах информационных систем («Великобритании, например, «Сифэкс-1», «Сифэкс-2»). В интерактивной системе абонент (за определенную плату) может не только запрашивать, но и передавать свою информацию в память центрального компьютера.

Телекоммуникации, как указывалось выше, могут осуществляться не только по телефонным линиям, но и по телевизионным каналам; абоненты принимают на своих телевизионных приемниках или мониторах всевозможные виды изображений, причем с помощью спутниковой связи такие контакты могут быть международными.

Чтобы передаваемая текстовая информация не подвергалась всевозможным помехам, сбоям, шумам, которые возникают при передаче по телефонным проводам, используются встроенные редакторы текста и так называемый X-Модем, обеспечивающий полный автоматический контроль за «чистотой» и бесперебойностью передачи информации.

Возможности телекоммуникаций довольно широкие: от обычной «переписки» между учащимися разных школ, городов, стран, совместной творческой работы по каким-то темам до широкой системы консультаций, и не только учащихся, но и учителей, методистов, коллективного обсуждения каких-то вопросов, проблем, т. е. от обучения учащихся, студентов до системы повышения квалификации педагогических кадров.

Осенью 1985 г. по инициативе Орегонского университета (США) был разработан проект телекоммуникационной системы, который объединил в качестве участников ряд школ США (штаты Монтана и Орегон), Непала, Японии, Индонезии, ФРГ, Египта и Канады.

Целью этого проекта [4], была не реализация телекоммуникации, а ее моделирование путем подготовки и пересылки (по почте) информации на флоппи-дисках. Учащиеся должны были овладеть умением, технологией подготовки информации для телекоммуникаций и одновременно ставилась познавательная задача — знакомиться с жизнью сверстников в других странах.

Координатор проекта в каждой школе ото-

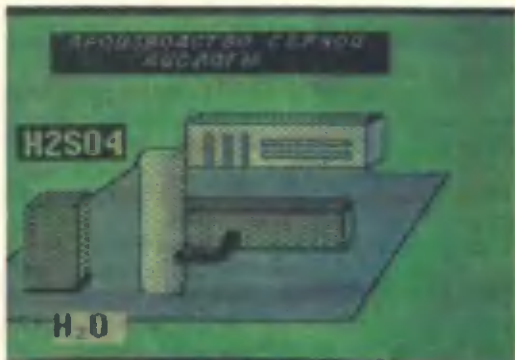
* Модем сокращенно от Modulator Demodulator — техническое устройство, которое позволяет компьютеру передавать или принимать информацию по телефонным линиям.

брал группу учащихся от 10 до 30 человек из среднего звена обучения. Предполагалось, что учащиеся будут обмениваться флоппи-дисками шесть раз в течение года. В период с 15 сентября по 15 февраля Каирский американский колледж, объединяющий свыше 50 национальностей, разослал в школы-участницы первые подготовленные им диски.

Опыт оказался не слишком успешным, поскольку не все дискеты доходили вовремя до адресатов (а некоторые не дошли вовсе), были сложности в понимании подготовленных программ, обусловленные разными причинами. Учителя на один час работы учеников тратили до пяти часов собственного труда не только на доводку программ, но и на их размножение в нужном количестве, подготовку к пересылке по почте, и др. Вместе с тем проект достиг своей цели — ученики приобрели опыт в подготовке информации для телекоммуникаций, овладели этой достаточно сложной технологией и узнали друг друга, получили много новой и подчас неожиданной информации.

В том же 1985 году в Австралии приступили к осуществлению нового проекта телекоммуникаций для начальной школы и детского сада «Киборд коннекшенз» [5], который объединил воспитанников четырех детских садов и школ в возрасте от 2 до 6 лет. Немного позднее к ним присоединились школьники США и Новой Гвинеи. В этом проекте три уровня, так что в нем могли участвовать дети от 2 до 12 лет. Для каждого уровня дискеты были разделены на части: рассказы, статьи, стихи, сочинения, пьесы. Был также специальный раздел для учителей в каждой из указанных частей.

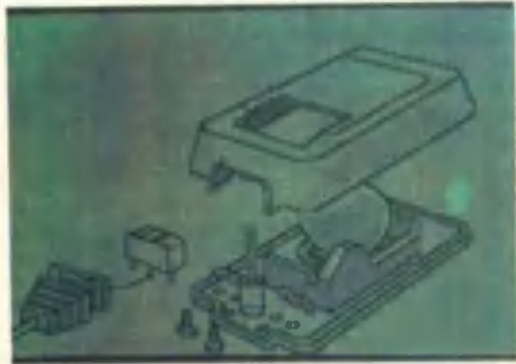
«Киборд коннекшенз» доступен школам, которые используют компьютеры «Эппл», «Би би эс» или «Ай би эм». Участники проекта вносят абонентную плату 10 долларов за одну часть проекта в год и 5 долларов за каждую дополнительную часть. Ре-

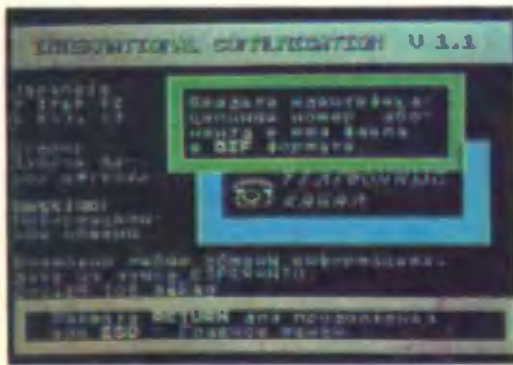


бята обмениваются своими творческими работами по указанным разделам, участвуют в коллективном творчестве (от них требуется либо владение манипулятором «мышь» либо клавиатурными навыками), по цепочке один начинает рассказ, другие продолжают его, развивая сюжет. Так ребята совместными усилиями сочиняют рассказ, пьесу, стихи и др. Это позволяет не только развивать творческие способности учащихся, но и значительно улучшает их грамотность, развивает речь, мышление.

Компьютеры в качестве терминала в телекоммуникационных системах находят все большее применение в образовании. Банки данных, включающие энциклопедические справочники, информацию о текущих событиях, оказываются доступными школам, колледжам, институтам и частным абонентам, что открывает совершенно новые возможности для самообразования. Можно получить любую необходимую информацию из любой точки страны в считанные секунды, можно обмениваться информацией с корреспондентом, стать активным участником творческой деятельности не только в пределах своей страны, но и с партнерами из других стран. Общеобразовательный и воспитательный эффект от такой работы, если, разумеется, ее соответствующим образом методически подготовить и организовать, трудно переоценить.

Проект «Интеркультурная обучающая сеть» [6] был совместно разработан учителями из штата Калифорния (США), Иллинойского университета (США), женского колледжа «Айоама какуин» в Токио (Япония), Хебрюского университета (Иерусалим) и объединил восемь групп учащихся общеобразовательных школ и колледжей в Калифорнии. Проект проходит экспериментальную проверку. Его цель — сотрудничество учащихся из разных стран, непосредственное знакомство с социальными, культурными, прочими проблемами друг друга, совместная исследовательская деятельность в





области общественных и естественно-научных программ обучения и попутно совершенствование речевых умений в иностранных языках.

Учащиеся начинают сотрудничество обменом «культурными пакетами», содержащими сведения о наиболее важных аспектах своей жизни. Просмотр пакетов ведет к оживленным дискуссиям, касающимся сходства и различий в жизни участников общения. Пакеты включают предметные образцы, рассказывающие о жизни и отдыхе молодежи (аннотации телевизионных передач, примеры из домашней и школьной деятельности; картины о спорте, музыкантах, модах; рецепты кухни; фотографии учеников; почтовые открытки с видами городов, монеты, почтовые марки; текстовые описания школ и школьной жизни, студенчества; информацию по всей стране. Учащиеся становятся авторами этих программ. Все примеры даются по-английски, а диалог в рамках проекта ведется на трех языках: английском, испанском и японском.

Учитель выполняет ведущую роль при разработке программ корреспонденции. Ведь дело вовсе не в том, чтобы ученики передавали друг другу интересную информацию. А в том, чтобы придать этому общению образовательный, воспитывающий, развивающий характер. Компьютерные, телекоммуникационные сети должны использоваться для создания новых форм совместной деятельности, которая вовлекла бы учащихся в

разработку определенных познавательных задач. Эта работа находится еще в самом начале исследования и требует много усилий со стороны педагогов, психологов, социологов.

Что касается такой формы телекоммуникаций, как аудио- и видеоконференции, то они позволяют провести диалог участников из различных уголков страны, мира по различным проблемам. При организации видеоконференций можно видеть друг друга либо на экранах телевизоров, либо на экране электронной доски или — для участников аудиоконференции — слышать друг друга [2].

Возможности компьютера, других средств информационной технологии еще мало изучены.

Изучение зарубежного опыта, критическое его осмысление с точки зрения перспектив развития учебно-материальной базы нашей системы образования, и в первую очередь школьного как органичного компонента системы социального, экономического, научно-технического и культурного развития всего общества, приводит к выводу о неотложной необходимости разработки специальной государственной программы внедрения средств информационной технологии.

Необходимо создать в нашей стране и странах социалистического содружества широкую систему компьютерных сетей для образования.

Литература

1. *Марец И.* Методологические основы дидактики. М., 1987.
2. *Полат Е. С.* Проблемы использования компьютеров в образовании развитых капиталистических стран // Информатика и образование. 1987. № 4; Полат Е. С. Перспективы развития учебно-материальной базы общеобразовательной школы. АПН СССР. 1987.
3. *Glossary of Educational Technology Terms / UNESCO.* 1984.
4. *Hagar M. J., Thoren Ch.* Egypt and the Floppy Disc // *The Computing Teacher.* 1987.
5. *Butter G.* Keyboard Connections // *The Computing Teacher.* 1987. April.
6. *Riel M.* The Intercultural Learning Network // *The Computing Teacher.* 1987. April.



О программе курса

Готовится новый учебник по информатике. Ведутся споры о методике преподавания, техническом оснащении курса, формах и методах обучения. Короче говоря, решаются тактические задачи. Но закончена ли научная разработка стратегии курса?

Рекомендованная программа для нового учебника по существу сводится к двум разделам: составлению алгоритмов и изучению вычислительной техники.

Довольно большой объем времени (102 ч) при непропорционально тесных рамках курса вынуждают авторов учебников углубиться в материал. При этом либо алгоритмический раздел будет перегружен никому не нужными сложностями, либо раздел, посвященный ЭВМ, наполнится излишне подробными сведениями. В худшем случае распухнут оба раздела.

За 34 ч вполне можно познакомить ребят с основами алгоритмизации, программированием, рассказать о важнейших принципах архитектуры и работы ЭВМ (причем очень может быть, что делать это нужно не в IX классе, а значительно раньше). Оставшиеся 68 ч приходится посвящать тому же самому материалу, только на более сложном уровне. Девятиклассникам предлагаются примеры из курса прикладной высшей математики: о приближенном решении уравнений методом отделения корней, об обработке результатов эксперимента методом наименьших квадратов и т. п. В X классе подробно рассматривается работа процессора, изучается устройство внешней аппаратуры. Понадобятся ли эти сведения завтрашним инженерам?

Конкретная компьютерная техника, поставленная во главу нынешнего курса, развивается быстрее, чем издаются учебники. За-

чем же принимать программу, которая вскоре потребует коренной переработки? Не разумнее ли включить в курс рассмотрение действительно основ информатики?

В настоящее время возможности ЭВМ, установленных в цехах и научных учреждениях, реализованы далеко не полностью. Вычислительные центры загружены частными задачами вместо глобальных вопросов оптимизационного и информационного характера. Терминалы зачастую используются в режиме справочника или настольного калькулятора. Чтобы изменить такую практику, завтрашнему специалисту необходимо сегодня дать широкое представление о возможностях ЭВМ, об оптимальном управлении, об обратной связи, о системах автоматизированного и автоматического управления. Пока же многие школьники считают, что ЭВМ нужна для облегчения вычислительных и графических работ так же, как, например, стиральная машина для облегчения стирки. Информатика в школе должна опровергнуть это ошибочное мнение. Но для этого необходимо пересмотреть содержание курса.

Воздерживаясь от конкретизации материала для новой программы, автор позволит себе внести некоторые предложения, осознавая их дискуссионный характер.

Расширить тематическое содержание курса, возможно, изменив название «информатика» на «кибернетика» (дело не в названии).

Для этого включить в программу следующие учебные разделы:

- анализ алгоритмов;
- элементы теории информации;
- элементы теории автоматического управления;
- элементы математической логики;

моделирование с помощью ЭВМ.
Дополнить курс ознакомительным материалом:
история информатики (кибернетики);
краткие биографии ученых и инженеров;
техническая кибернетика;
кибернетика в экономике;
кибернетика живых систем;
элементы теории игр;
случайные процессы.

Кроме того, следует рассмотреть возможность более раннего выхода школьников на ЭВМ. Перенести знакомство с персональными компьютерами в младшие классы. В IX классе отвести на информатику 2 ч в неделю, оставив в X классе только 1 ч.

Что это даст? Возможность получить целостное представление об информатике как науке; сформировать у школьников информационное мышление; углубить представления о роли информатики и ЭВМ в современном обществе; расширить политехническую направленность обучения; усилить профориентационное воздействие курса; увеличить число межпредметных связей (математика, физика, биология, обществоведение, языки и т. д.); повысить увлекательность и

доступность учебного материала.

И последнее. Существует мнение, что введение курса информатики в программу школы призвано способствовать компьютеризации среднего образования. Но нельзя же путать разные вещи! Информатика имеет такое же отношение к ЭВМ, как термодинамика к устройству холодильника. «Полная и безоговорочная» компьютеризация школы не может считаться целью курса информатики. Получается, что через несколько лет, когда учебные ЭВМ получат в школе постоянную прописку и станут привычным средством труда, информатика будет не нужна?

Стратегия курса, по-видимому, в другом. От компьютерной малограмотности — к компьютерной грамотности. От компьютерной грамотности — к алгоритмическому мышлению. От алгоритмического мышления — к информационной культуре. От информационной культуры — к оптимальной организации своего труда. От оптимальной организации своего труда — к оптимальной организации своей страны!

А. АВЕРБУХ,
учитель СШ 682,
Москва

115

Снова бьем своих?

Грустные размышления по поводу ряда критических выступлений

Критиковать сейчас стало модно, особенно в резкой форме, по этой причине я уже второй год не выписываю «Литературную газету». Но номер от 27 января 1988 г. мне положили на стол со словами: «На, почитай про свою любимую компьютеризацию!». И подборка этих статей повергла меня в уныние. Дело здесь не только в описании положения дел с компьютерами и их внедрением, сколько в советах и рекомендациях, которые там даются, а эти советы у непосвященного читателя могут вызвать только одну реакцию: у нас все плохо и хорошо не может быть, если мы немедленно не кинемся закупать персональные компьютеры в США и Японии. Как человек, более 20 лет работающий с вычислительной техникой — в качестве студента, инженера-эксплуатационника, программиста и преподавателя, — я могу предсказать, к чему приводят подобные публикации. Для этого позволю себе сделать маленький исторический экскурс.

В последнее время из газет и журналов мы

много узнали об истории отечественной генетики, последствиях происходившихся в 30—50-е гг. негативных процессов. Гораздо меньше говорится о том, что кибернетика и ее энтузиасты в 50-е гг. пережили нечто подобное. Так, на первом курсе преподаватели зачитывали нам строчки из Большой Советской Энциклопедии, где говорилось, что кибернетика — «это лженаука, разработанная буржуазными философами с целью фальсификации...». К сожалению, это были не последние удары по отечественной вычислительной технике и науке.

В конце 60-х гг. наша промышленность освоила серийный выпуск вычислительных машин, вполне удовлетворявших потребности науки и техники, например М-220, Минск-32. Особо следует отметить БЭСМ-6 — явление в своем роде. Но в целях стандартизации разработка и производство различных типов больших ЭВМ были прекращены, и началось массовое внедрение ЕС ЭВМ — т. е. линии IBM 360/370. Сегодня, 20 лет спустя, лучшие

образцы ЕС ЭВМ уже достигают производительности БЭСМ-6*...

Еще более печальна история малых ЭВМ. В конце 60-х — начале 70-х гг. мы фактически имели прообраз персональных компьютеров — ЭВМ серии «МИР» (Машина для Инженерных Расчетов). Детище академика В. М. Глушкова, они сразу полюбились и инженерам, и студентам, и преподавателям. На этих машинах проводили первые опыты по обучению школьников программированию. Конструкторские решения, программное обеспечение продемонстрировали лучшие достижения советской науки. Так, до сих пор программисты со стажем не расстаются с книгами описаний МИРовских стандартных программ, хотя сами ЭВМ давно списаны. «Русские» языки общения с пользователем, «дружественная» диагностика снимали психологический барьер. Существовали ассоциации пользователей ЭВМ «МИР», популярность машин росла, но... В угоду той же стандартизации и «ориентации на лучшие зарубежные образцы» мы перешли на СМ ЭВМ, у которых также есть «родной брат» за океаном.

На сколько лет затормозили нас эти решения, какой моральный урон нанесли? Ведь на всех конференциях каждая оригинальная идея встречается вопросом: «А что сделано у американцев?» И если работы в этом направлении «там» не велись, то и у нас они считались нецелесообразными. Значит ли это, что следует снять с производства БК-0010?

Об этой многострадальной машине хочется сказать особо. Как весенний ручеек, пробил «бэкашки» себе дорогу через ГОСТы, госприемку и т. д., и вот новый залп критики, причем не до конца объективной, не учитывающей сегодняшнее положение дел. В Прибалтике и на Урале, в Сибири и в Москве ширится круг энтузиастов, помогающих совершенствовать и техническую сторону, и программное обеспечение маленьких посланцев Павловского Посада. Уже работает Бейсик, есть текстовый и графический редакторы, специальный язык для создания обучающих курсов и т. д. Почему их не увидели авторы яростных критических статей? Говорят о быстродействии, объемах памяти, но забывают о том, что компьютер-то бытовой! Ведь за огромное быстродействие СМ и мегабайты ЕС ЭВМ мы платим государственные миллионы, а за килобайты БК — свои, кровные сотни. И не надо быть большим специалистом, чтобы знать, что одновремен-

ное увеличение быстродействия и памяти сразу резко увеличит стоимость машины, особенно если к ней добавить еще дисковод и печатающее устройство. Но пока подклассов в классификации бытовых компьютеров нет, и вместо логических требований сопряжения с бытовыми магнитофонами и телевизорами на первый план вышли требования промышленные: быстродействие, память, энергопотребление и т. д.

Но это все техническая сторона, а есть и моральная. Нам будет тяжело воспитывать патриотизм и чувство гордости за достижения нашей науки и техники, если десятки тысяч школьников каждый день будут садиться за пульта японских «Ямах» и общаться с ними по-английски. Прошу понять меня правильно: покупка техники за рубежом и производство своей не являются альтернативными вариантами, одно должно дополнять другое, но нельзя же всегда оглядываться «за океан» и становиться «в затылок» их разработкам. В домашних закутках и чуланах лабораторий «народные умельцы» спаяли уже не один десяток персональных компьютеров. Так почему же ГКВТИ, журналы «Моделист-конструктор», «Радио», «Микропроцессорные средства и системы», газета «Комсомольская правда» и МЭП СССР не объявят открытый всесоюзный конкурс на лучшую разработку с тем, чтобы модели-победители были сразу запущены в производство, а их авторы получили солидные премии?

Вместе с вопросом о технике теперь научились сразу поднимать вопрос и о программном обеспечении. И в последнее время наша печать изредка робко интересуется: «А почему мы общаемся с ЭВМ по-английски?» О том, насколько это важно, специалисты могут почитать в журнале «Информатика и образование». Наиболее широко идеи академика В. М. Глушкова нашли воплощение в школьном курсе ОИВТ, внедренном благодаря энергичным усилиям академика А. П. Ершова, поэтому на страницах ИНФО постоянно появляются публикации о результатах научно-педагогических исследований по вопросам выбора наилучших форм обучения. Однако редкое периодическое издание с большим тиражом откало себе в удовольствии опубликовать высказывание высококвалифицированного программиста о «... вредности внедрения русской лексики...», «...ложном чувстве патриотизма в программировании...», «общепринятых за рубежом английских терминах в программировании...», «...нетрудно выучить тридцать английских слов...» и т. д. Меня лично этот вопрос интересует давно, я задавал его и на конференциях, и в частных беседах. Не всегда получал прямые ответы, но кое-то прояснилось. Во-пер-

* Автор письма несколько утрирует ситуацию. Отмеченное им событие произошло лет 10 назад. Впрочем, тираж выпуска ЕС 1060 был очень небольшим. — *Прим. ред.*

вых, англоязычное общение сохраняет клан «жрецов» возле компьютеров; во-вторых, проще объяснить ориентацию на западную технику; в-третьих, выдвигается тезис о необходимости совместимости разрабатываемого программного обеспечения. Тезис верный, но ведь совместимости можно достичь с помощью внутренних программных средств, опыт таких разработок у нас имеется.

Призыв со страниц «Литературной газеты» об ориентации на заграничные технические и программные средства напоминает клич: «Бей свое, а то чужого мало купим!»

Что оценивать?

В апреле 1988 г. в Гомеле состоялась вторая областная олимпиада по «Основам информатики и вычислительной техники» среди техникумов, которая заставила серьезно задуматься о совершенствовании форм проведения олимпиад вообще.

Наряду с проверкой усвоения знаний олимпиада должна преследовать цель выявления и поощрения наиболее способных и одаренных учащихся. Критериями оценки результатов работ должны быть не только правильность составления алгоритмов и программ, но и оригинальность мышления учащихся, знание тонкостей изучаемого ими языка, умение реализовать и отладить программу с помощью ЭВМ. Практика «безмашинного» преподавания ОИВТ в первые годы обучения и необеспеченность многих техникумов компьютерами вплоть до настоящего времени привели к забвению основной цели курса, состоящей в «...формировании представлений об основных правилах и методах реализации решения задачи на ЭВМ и элементарных умений пользоваться компьютерами для решения задач...» (Основы информатики и вычислительной техники: Программа средней общеобразовательной школы. М., 1985).

Анализ итогов прошедшей олимпиады по ОИВТ показал, на наш взгляд, несовершенство и несостоятельность принятой методики ее проведения. Это выразилось в неподготовленности членов жюри к оценке нестандартных «программных ходов» и использованию «экзотических» функций расширенного Бейсика, что, впрочем, вполне естественно. Во-первых, преподаватель может и не знать всех тонкостей практического программирования; во-вторых, хитрые программные приемы, приобретенные учащимися самостоятельно за пультом ЭВМ, как правило, не

Нам действительно необходимо много сделать, и в первую очередь научиться деловитости, которую демонстрируют разработчики и производители «Ямах» и «Эпплов». А еще хорошо бы научиться обсуждать и спорить без грубых выпадов, спокойно, по-деловому. Тогда и компьютеры наши быстрее достигнут необходимого уровня.

П. МАРЧЕНКО,
канд. техн. наук
Пермь

укладываются в рамки алгоритмического языка. Задания же для решения на олимпиаде требуют обязательной записи на алгоритмическом языке. Этот, безусловно важнейший, компонент «безмашинного» обучения, однако, не должен превалировать над практическими навыками и тем более не должен сковывать задатки творческого мышления юных программистов.

Приведем конкретные примеры. Программа расчета числа сочетаний элементов без вычисления факториалов чисел на Фокале, оцененная на олимпиаде низким баллом, при последующей проверке на БК-0010 показала полную работоспособность. В другой программе на языке Бейсик была сделана (правда не совсем удачная) попытка применения функций обработки строковых величин (STR, MID; VAL, LEN). Она практически не была оценена, несмотря на свою оригинальность и нестандартность. Пусть эта программа в конкурсной работе и содержала некоторые ошибки, но не следует забывать о том, что отладка программ является таким же неотъемлемым этапом практического программирования, как и составление алгоритмов. Олимпиада, ориентированная на «безмашинную» технологию обучения ОИВТ, конечно же, этого учесть не может.

Какие можно сделать выводы? Для восстановления «забытой» ориентации курса основ информатики на овладение навыками практического программирования (действительно, настоящий специалист должен писать хорошие программы, даже забыв алгоритмический язык), следует проводить дифференцированные олимпиады для двух групп учащихся: имеющих и не имеющих практику создания и отладки программ на ЭВМ. Группа «практиков» должна дополнительно соревноваться за пультом ЭВМ. Это, безусловно, потребует

от организаторов некоторых усилий. Необходимо обеспечить достаточное количество рабочих мест и качественное «судейство» с привлечением квалифицированных программистов. Критериями оценки работ групп «практиков» должны быть работоспособность и оптимальность программ и оригинальность мышления.

Пока техникумов, оснащенных компьютерами,— единицы, и на данном этапе такие олимпиады целесообразно проводить сразу на республиканском уровне.

Г. ТИПИКИНА,
преподаватель,
г. Гомель

Открытое письмо директору завода «Квант»

Уважаемый Богдан Иосифович Полатайко! Обращаюсь к Вам с письмом через журнал, ибо Вы от таких просителей — несчастных обладателей Ваших компьютеров ДВК-2М — зашищены «непробиваемой броней» пропускной системы. Добраться до Вас нет никакой возможности.

118 Три дня (31 мая, 1 и 2 июня) с утра до позднего вечера держала я осаду Вашего завода с целью получить исправные платы КТЛК, МПУ, НГМД для ДВК-2М. К концу третьего дня меня уже знала половина сотрудников, начиная с главного инженера и кончая уборщицей Галей. Высказывали сочувствие, пытались чем-то помочь, даже подкармливали... Главный инженер на моем письме дал резолюцию зам. начальнику ПО отгрузить указанные в письме платы, но действительна только Ваша подпись, а ее нет. Раз так, то Вы и несете полную ответственность за Ваши компьютеры ДВК-2М. Сейчас их завод уже не выпускает, но нам-то с ними работать, они входят в комплект КУВТ-86. Стоя у проходной по 8—10 часов в день, я думала: посадить бы Вас за ДВК-2М, который каждый день выходит из строя, перед аудиторией учителей, оторванных от своей основной работы, чтобы получить начальную компьютерную грамотность! Как бы Вы смотрели им в глаза, какие находили слова в

оправдание, на что ссылались? Мне в таких ситуациях было очень стыдно. С такими компьютерами мы ни на шаг не продвигнемся вперед. Я много ездила по стране, общалась с обладателями ДВК-2М и везде, по всей стране, в Ленинграде, Таллине, Риге, Алма-Ате и других городах слышала о них отрицательные отзывы.

Плата локальной сети КТЛК у нас в институте нормально работала ровно один месяц, за полтора года отремонтировать ее мы не можем, хотя обращались и в Алма-Ату (ПО ВТИ), и в центр ВТИ в Караганде.

Почему Вы делаете заведомый брак? Потому, наверное, что не несете ответственности за свою продукцию, да и не только Вы. С заводом «Экситон» мы ведем годовую безуспешную переписку из-за трех микросхем к БК-0010. К сожалению, к этому сводятся взаимоотношения пользователей и создателей ЭВМ. Множатся громадные ремонтные организации, но они не в силах отремонтировать такую массу бракованной продукции.

Как вырваться из этого заколдованного круга?

Л. КОЙСИНА,
методист ОИИУ,
г. Караганда

Информатика в сельской школе

Уже три года в школах нашей страны изучается новый предмет — основы информатики и вычислительной техники. Накопился определенный опыт, который необходимо изучить и сделать выводы на будущее.

Пришел я в школу после длительной работы в вузе. Сразу же встал вопрос: с чего начать? Было ясно, что теорию нужно подкреплять практикой, что ребята должны получать навыки в общении с машиной. Жизнь показывает, что существует опреде-

ленный психологический барьер в общении с автоматикой, его нужно преодолеть, поверить в технику. Машина требует аккуратности, точности, последовательности в действиях. Итак, нужна вычислительная техника.

В первый же год школе удалось приобрести 16 программируемых микрокалькуляторов МК-56. На них сразу же начали работать ученики IX и X классов, были введены факультативные и кружковые занятия

в VIII и VII классах. В следующем году десятиклассники с осени начали осваивать язык Бейсик, и раз в месяц мы выезжали на один день в Казань работать в классе «Ямаха».

В 1987 г. приобрели шесть микро-ЭВМ БК-0010. Встала задача создания дисплейного класса. Типового проекта не было, и поэтому возникла идея разработать проект, который бы максимально удовлетворял требованиям учебного процесса, санитарно-гигиенических норм и техники безопасности; доступности монтажа в условиях сельской школы.

Как известно, наибольшие неприятности в дисплейном классе связаны с видеоконтрольными устройствами, в качестве которых используются бытовые телевизоры. Это и рентгеновское, инфракрасное, радиоизлучения, и мерцание изображения с частотой 50 Гц, недостаточная контрастность и четкость символов. Особенно сильно это проявляется, когда телевизор типа «Юность» стоит непосредственно перед учеником.

Мы решили использовать телевизоры 2-го

класса с большим экраном и стабильной работой генераторов разверток, разместив их так, чтобы расстояние от лица ученика до экрана составляло 1,5 м. Каждый телевизор, как это видно из прилагаемого снимка, размещается на отдельном столе, отделенном от ученического мостиком, на котором находится блок питания БК и выключатель телевизора. За ученическим столом сидят двое, поэтому они рассматривают изображение под углом более 15 градусов к оси кинескопа, что уменьшает воздействие рентгеновского излучения. Со стороны задней стенки телевизор закрыт щитом. Регулировка выполняется до начала занятий при профилактическом обслуживании, поэтому во время работы ученикам нет нужды подходить к телевизорам. Электрическая сеть смонтирована в металлических трубах, напряжение на столы подается только во время работы ЭВМ.

В классе отсутствуют традиционные доска и мел. Вся передняя стена класса покрыта белой водоземлемой краской — так образовался большой экран, на который можно одновременно выдавать информацию с ко-



доскопа, диапроектора и эпидиаскопа. В дисплейном классе кодоскоп особенно эффективен, так как позволяет демонстрировать заранее написанные алгоритмы и программы и возвращаться к ним при необходимости. В специальном шкафчике сзади смонтирована киноустановка.

В классе организовано 12 рабочих мест. Пять столов, стоящих сзади, предназначены для работы с литературой, подготовки программ и т. п. На них же ставятся портативные пишущие машинки, которые используются как для отработки техники удара по клавишам, так и для освоения машинописи.

Занятия у нас ведутся по программе, предложенной для написания конкурсных учебников; критерием являются требования к знаниям и умениям учащихся, изложенные в инструктивном письме Минпроса РСФСР № 71 от 19 мая 1987 г. При этом я стремлюсь постоянно показывать роль ЭВМ в сельскохозяйственном производстве. Во всех разделах программы решаются задачи по сельскохозяйственной тематике (экономика, кормовые рационы, распределение площадей, строительство, энергетика).

В классе кроме наших учеников еженедельно работают ребята из различных школ района и г. Зеленодольска. В августе и январе мы проводили «Школы юного программиста», участниками которых были девятиклассники и десятиклассники из глубинных сельских школ республики. Неоднократно проводились семинары учителей района и республики. Наш опыт показывает: в условиях нехватки хороших дорог, низких температур зимой и резких колебаний влажности, что весьма неблагоприятно сказывается на надежности ЭВМ, выгоднее создавать не «школу на колесах», а межшкольные кабинеты вычислительной техники. Практически, каждое хозяйство имеет автобус, который можно использовать для перевозки учеников. В межшкольном каби-

нете можно собрать хорошую библиотеку программ, справочной литературы, оборудовать стенды и действующие макеты узлов ЭВМ, иметь наборы плакатов, диафильмов и т. д. Здесь легче организовать квалифицированное обслуживание вычислительной техники. За год работы нашего класса каждая ЭВМ наработала около 700 ч, отказов не было. Сказалось и то, что все блоки ЭВМ зафиксированы на столах деревянными рамками. Класс смонтировали пять учеников IX класса и я — все бесплатно.

Теперь пожелания на будущее. Кабинету вычислительной техники нужен лаборант (учитывая сложность техники и разнообразие задач) с соответствующей зарплатой. Необходима специализированная служба по ремонту школьной вычислительной техники. Нужно организовать снабжение школ пакетами прикладных программ по мере их разработки.

Мы не рассматриваем свой класс как идеальный, стремимся к совершенствованию, к получению новой техники. Но ожидание совершенной комплектной техники приводит к «бумажной» информатике, и часто это прикрывает нежелание работать всерьез.

Компьютеризация школы только-только начала набирать силу. Мы в самом начале пути. Впереди у нас много радостей и огорчений. Сейчас самое главное — поверить всем, что ЭВМ в школе всерьез и навсегда. ЭВМ в школе — это мощный рычаг, освобождающий учителя и ученика от механической рутинной работы и создающий грандиозные возможности для творческой, увлекательной деятельности по познанию окружающего мира и места человека-созидателя в нем.

С. ЗЛАТОУСТОВ,
учитель информатики
Айшинской средней школы
Зеленодольского района ТАССР,
учитель-методист

Это нужно учителю

Ю. Л. Львова — известный педагогической общественности человек, автор многих книг и статей, посвященных пробуждению творческого начала у учащихся. На протяжении десятилетий она и ее единомышленники создают в школе, которая носит имя А. Макаренко, уникальный климат, сочетающий требовательность к учащимся с развитием самостоятельности мышления у ребят. Редакции было приятно получить письмо-размышление от столь уважаемого мастера своего дела.

Разрешите мне поделиться некоторыми своими учительскими соображениями.

Публикацию материалов по использованию компьютерной логики и техники при изучении грамматики считаю чрезвычайно нужной и интересной для учителей русского языка.

Еще 20 лет назад, оттолкнувшись от приемов программированного обучения, я стала разрабатывать наглядные пособия по дозам информации той или иной темы, ведущих ребят к самостоятельным выводам; составила информационные

карточки, соответствующие ответам по каждой дозе, а заканчивала изучение грамматической темы составлением алгоритма. Мне удалось повеселить ребят в то, что происходит прежде, чем рука напишет ту или иную букву или поставит знак. Это имело поразительный результат.

Материалы Вашего журнала, связанные с изучением грамматики (№ 1, 1988) очень хорошие, добротные, основательные. Особенно полезна и многогранно может быть использована в практике работы статья Г. Городиловой, Т. Самороднической, Р. Аллахвердиева, где интересны задания работы по алгоритму, где сами алгоритмы правописания «Н и НН в частях речи», «О—Ё после шипящих», «Б после шипящих» очень хороши.

Данные алгоритмы и задания, правда, могут быть использованы лишь в вузах, где учатся люди, уже знающие курс орфографии, именно весь курс. Но ведь введение в журнале рубрики «Педагогический опыт» (отличной, нужной рубрики!) предполагает среднюю школу.

В таком случае мы, учителя русского языка, вправе ждать публикаций, раскрывающих приемы, методы алгоритмизации по частным вопросам грамматики. Это нужно учителям. Это должно стать заботой журнала.

Возникает и опасение: как бы не впали мы в очередную крайность. Как бы не стал компьютер единственным способом обучения русскому языку.

Компьютерная техника и алгоритмы полезны, на мой взгляд, лишь в тех разделах науки об языке, где учащимся предстоит делать выбор: что и почему писать.

Значит, и алгоритмы должны раскрывать последовательность умственных операций частного характера и быть доступными пониманию школьников IV—VII классов.

Опыт показал, что алгоритмы, составленные совместно с учителем и учащимися, наиболее эффективны. Следовательно, очень важное значение для фронтальной работы с классом имеет логика вопросов, обеспечивающая составление алгоритма. И еще — реальное время в школах весьма ограничено программой.

Надо журналу на своих страницах организовать семинар учителей русского языка по овладению методикой компьютеризации обучения.

Ю. ЛЬВОВА,

учительница русского языка и литературы
СШ № 50, кандидат педагогических наук
г. Львов

121

Электронная медсестра

Можно ли температуру человеческого тела измерить с помощью компьютера? Можно! И не в одной, а сразу в 16 точках, если пристыковать к нему электронный термометр, разработанный американскими инженерами.

Данные, поступающие со всех 16 каналов термометра, обрабатываются персональным компьютером APPLE II или IBM-PC-XT, снабженным специальным пакетом программ, и могут быть выведены на дисплей, распечатаны на бумаге или записаны на дискету для дальнейшего использования. Кроме того, компьютер снабжен программой, позволяющей строить график изменения температуры во времени с выводом его на дисплей или бумагу.

Электронный термометр обеспечивает изменение температур в диапазоне от 0 до 50 градусов по Цельсию с шагом 0,015 °С, а при измерении небологических температур его шкала может расширяться до пределов —200 ÷ +1000 °С.

Превратил ЭВМ в термометр электронный блок преобразования температуры в цифровой код, а сделал такой термометр удоб-

ЧТО?
МОЖЕТ
ЭВМ

ным инструментом пакет прикладных программ. Точно так же из компьютера можно сделать вольтметр, барометр, спидометр и даже такой сложный прибор, измеряющий многие параметры работы сердца, как Cardiomax.

Этот прибор, соединенный с компьютером IBM PC, позволяет снимать и контролировать такие параметры, как пульс, кровяное давление, расход крови, количество крови, выбрасываемой сердцем за одно сокращение, температуру крови и вводимого лекарства, кривую скорости растворения препарата в крови и многое другое. Вся эта информация одновременно выводится на экран в удобном для наблюд-

ния виде, а также может быть распечатана или записана на дискету для хранения и последующей обработки.

Дисплей в кармане

Когда речь заходит о компьютере, особенно персональном, у многих из нас в первую очередь в сознании возникает некий образ пишущей машинки в сочетании с телевизором. Однако скоро представление о компьютере изменится. И вызвано это в первую очередь уменьшением дисплея до размеров коробки карандашей.

Это стало возможным в связи с появлением дешевых люминесцентных и плазменных алфавитно-цифровых и графических дисплеев с высокой разрешающей способностью.

Дисплей комплектуется одноплатным интерфейсом, обеспечивающим связь по стандартному каналу с компьютером. Интерфейс обеспечивает воспроизведение 25 строк по 80 символов в каждой или графической информации с высоким качеством.

Соединенный шарниром с плоской клавиатурой, такой дисплей напоминает скорее папку для бумаг, чем телевизионный ящик, что делает компьютер домашним другом и хорошим попутчиком.

Круглый стол: Информатика-88/89

«Содержание, методика, программная поддержка курса ОИВТ»

В период становления курса «Основы информатики и вычислительной техники» необходимо дать учителям возможность сделать осознанный выбор одного из существующих альтернативных учебных пособий.

С этой целью Московским городским институтом усовершенствования учителей совместно с Главным управлением народного образования Москвы было проведено ознакомление учителей с методическими концепциями авторских коллективов и организация опережающей и дифференцированной методической помощи учителям.

27 апреля 1988 г. состоялся общегородской «круглый стол» с участием авторских коллективов двух экспериментальных учебных пособий по информатике (под редакцией А. П. Ершова и В. А. Каймина), методистов, творчески работающих учителей.

Целью общегородского «круглого стола» было ознакомление методического актива Москвы с методологическими и методическими концепциями авторских коллективов по преподаванию курса ОИВТ (машинный вариант). Необходимость такого обсуждения была вызвана переходом московских школ на преподавание курса ОИВТ по машинному варианту, так как уже более половины школ оснащены дисплейными классами.

Опыт преподавания курса по машинному варианту (начиная с 1985 г.) выявил существование различных методологических и методических подходов. Творческие дискуссии развернулись вокруг целей и задач курса, его структуры, методики. Что должен дать курс учащимся: компьютерную грамотность или информационную культуру? Каков должен быть удельный вес информатики, алгоритмизации, основ ВТ? Как учить алгоритмизации (используя учебный алгоритмический язык, производственный язык программирования или тот и другой)? Быть или не быть блок-схемам? И т. д.

Многие дискуссионные вопросы нашли свое отражение в позициях двух авторских коллективов: переработанного (вернее, заново написанного) учебного пособия под редакцией А. П. Ершова и экспериментального учебного пособия под

редакцией В. А. Каймина, занявшего второе место на конкурсе учебников информатики.

Авторский коллектив учебного пособия под редакцией А. П. Ершова представлял кандидат физико-математических наук, доцент МГУ А. Г. Кушниренко. В своем выступлении он обратил внимание учителей на то, что учебник сохраняет преемственность с ныне действующими пробными учебными пособиями для IX и X классов, использует тот же самый школьный алгоритмический язык, опирается на ту же самую систему понятий. Методика введения этих понятий, однако, существенно переработана, упрощено изложение ряда трудных вопросов.

Основная задача, которую ставили перед собой авторы, — сделать учебник доступным реальной массовой школе с ее довольно низким средним уровнем математической подготовки учащихся.

Изложение чаще всего ведется проблемно, «от частного к общему». Например, все конструкции алгоритмического языка вводятся по такой схеме: наглядная задача, требующая для решения введения новой конструкции;

пример использования конструкции при решении этой задачи;

общий вид конструкции, ее формальное описание.

Главное методическое новшество — попытка изложить основные понятия информатики и основные конструкции программирования, опираясь на наглядные, житейские и геометрические представления школьников, не привлекая на первых порах ни математических понятий, ни, по возможности, математической символики. Для этого выбран такой подход: все конструкции алгоритмического языка вводятся на примерах алгоритмов управления несколькими автоматическими устройствами. В основном тексте учебника таких устройств четыре:

«Робот» — идеализированная транспортная тележка, перемещаемая по клетчатой плоскости в четырех направлениях;

«Чертежник» — идеализированный графопостроитель, умеющий рисовать отрезки на бесконечной плоскости (бесконечном листе бумаги);

«Измеритель» — идеализированный набор дат-

чиков, который может быть установлен на «Роботе», перемещается вместе с ним и позволяет измерять различные величины при движении «Робота»;

«Черепашка» — исполнитель с системой команд обычной черепашки в языке Лого, воспроизведенной в пробном учебном пособии для X класса.

В классе алгоритмов управления перечисленными автоматическими устройствами можно поставить широкий спектр задач, начиная с элементарных упражнений на закрепление материала и кончая творческими задачами повышенной сложности. Это позволяет варьировать глубину индивидуальной проработки материала, сохраняя, единым для всего класса темп прохождения новых понятий.

Докладчик проинформировал аудиторию о готовящемся учебно-методическом комплексе: учебное пособие, программное обеспечение, методическое пособие для учителей. Он подчеркнул, что его важнейшей составляющей является программное обеспечение, которое реализует систему исполнителей «Робот», «Чертежник», «Измеритель» и др. Это программное обеспечение имеется для «Ямахи» (оно и было продемонстрировано) и готовится для «Корвета», УКНЦ и, частично, для БК-0010 с Фокалом.

Затем состоялся обмен мнениями. Учителя и методисты прежде всего интересовались: можно ли преподавать курс ОИВТ по обсуждаемому учебному пособию на имеющейся в настоящее время в школах Москвы ВТ («Агаты», КУВТ-86 с Фокалом и Бейсиком)? Будет ли разрабатываться представленное выше ПО на эти виды техники?

Ответ докладчика был не очень утешительным. Разработка такого ПО на «Агаты» пока не финансируется, а технические возможности БК-0010 позволяют реализовать это ПО лишь частично.

Другая группа вопросов касалась возможности преподавания курса ОИВТ по данному учебнику не в IX—X классах, как это предусматривается существующим учебным планом школы, а раньше, в VII—VIII классах. Участники «круглого стола», как учителя, так и докладчик, согласились, что такой перенос вполне возможен. Более того, методика изложения материала, предложенная в данном учебнике, даже более соответствует уровню подготовленности и психологическим особенностям учащихся именно VII—VIII классов.

Итогом обсуждения было общее мнение, что учебник весьма интересен, доступен для учащихся. Однако преподавать по нему в IX—X классах курс ОИВТ на имеющейся в настоящее время в Москве вычислительной технике без соответствующего программного обеспечения практически невозможно и нецелесообразно.

С другой стороны, заложенная в нем методика интересна и перспективна, и можно уже в 1988/89 учебном году использовать ее во внеурочной кружковой работе и на факультативах. В перерыве интересующиеся могли более подробно ознакомиться с программным обеспечением.

Второй авторский коллектив представлял кандидат технических наук, доцент МИЭМ В. А. Каймин. Докладчик дал общий обзор учебного пособия, подчеркнул, что в целом его структура соответствует структуре перспективной программы курса ОИВТ.

Цели курса можно подразделить на задачу-минимум и задачу-максимум. Задачей-минимум является обеспечение компьютерной грамотности учащихся, которую можно определить как формирование представления о возможностях ЭВМ и умения читать, писать, считать, рисовать, работать с программами и данными, используя компьютер. Таким образом, учащиеся должны научиться работать с текстовым и графическим редакторами, базой данных и электронными таблицами, поработать с простейшими программами на языке программирования.

Формирование компьютерной грамотности по экспериментальному учебному пособию и в соответствии с предлагаемым тематическим планированием должно обеспечиваться к концу первого полугодия IX класса. Причем целесообразно следующее распределение учебного времени по полугодиям:

IX класс: 1 полугодие — 1 ч в неделю;

2 полугодие — 2 ч в неделю;

X класс: 1 полугодие — 1 ч в неделю;

2 полугодие — 2 ч в неделю.

Задача-максимум состоит в формировании информационной культуры учащихся, которая пред-

123

лагает:

- умение грамотной постановки задач;
- понимание формализованных описаний моделей;
- овладение алгоритмическим языком для описания алгоритмов;
- знание элементов языка программирования и умение кодировать на нем алгоритмы;
- умение проводить отладку программ на ЭВМ и интерпретировать полученные результаты;
- знание устройства и физических принципов работы ЭВМ.

Формирование информационной культуры начинается изучением основ алгоритмизации и программирования во втором полугодии IX класса. К концу IX класса учащиеся должны приобрести умения составления алгоритмов и их записи на алгоритмическом языке, кодирования алгоритмов на языке программирования и отладке программ на ЭВМ.

Постановка задач, формализованное описание моделей и методов решения, а также систематическое конструирование алгоритмов методом последовательной детализации с обязательной последующей их реализацией на ЭВМ изучается в первом полугодии X класса. В конце этого полугодия изучаются правила систематической проверки правильности составляемых алгоритмов и программ, рассматриваются основные законы информатики как научной дисциплины.

Курс завершает во втором полугодии X класса изучение архитектуры и программного обеспечения ЭВМ, формы представления различных видов (числовой, текстовой, графической) информации в ЭВМ, физических принципов работы отдельных устройств ЭВМ. Учащиеся знакомятся с применением ЭВМ в различных сферах человеческой деятельности.

Параллельно выступлению, на «Ямахе» демонстрировалось имеющееся программное обеспечение, поддерживающее данное учебное пособие.

Затем выступили члены авторского коллектива готовящегося учебно-методического комплекса: Л. А. Кугель, учитель школы № 400 Перовского района; Е. М. Кузницкий, методист Люб-

линского района; Е. А. Ерохина, сотрудник МИЭМ. Они рассказали о структуре и содержании готовящихся методического пособия для учителей, задачника и программного обеспечения.

Л. А. Кугель продемонстрировал на БК-0010 с Бейсиком несколько программ, созданных им вместе с его учениками в поддержку предлагаемого пособия.

Затем состоялся обмен мнениями. Наибольшее число вопросов вызвало предлагаемое в учебном пособии параллельное изучение учебного алгоритмического языка и производственного языка программирования. Мнения разделились. Одни предлагали сделать учебный алгоритмический язык не только способом записи алгоритмов, но и реальным языком программирования, написав соответствующие трансляторы на различные виды ВТ. Таким путем, например, пошел учитель школы № 167 Ленинградского района А. И. Ковнер, который написал транслятор учебного алгоритмического языка на СМ ЭВМ.

Другие высказывали сомнение в необходимости его использования вообще и предлагали ограничиться использованием производственных языков программирования.

Однако большинство считает, что в настоящее время имеющаяся ВТ и программирование к ней достаточно жестко определяют методику изучения основ алгоритмизации и начал программирования. Суть этой методики состоит в параллельном изучении учебного алгоритмического языка и языка программирования; первый необходим в процессе построения алгоритма методом последовательной детализации, второй — для

реализации этого алгоритма на ЭВМ.

Участники «круглого стола» высказали предложения в адрес Государственного комитета по народному образованию о необходимости использования и в дальнейшем альтернативных учебников и предоставления учителям права выбора. Многими учителями была высказана также мысль о необходимости быстрее создания учебно-методических комплектов к каждому учебнику, включающих программное обеспечение, методическое пособие для учителей и сборник упражнений и задач.

С авторскими коллективами была достигнута договоренность, что с аналогичными сообщениями они выступят на районных методических объединениях учителей информатики. В течение мая 1988 г. такие встречи были проведены, к сожалению, не во всех районах. Каждое методическое объединение, более того — каждый учитель получил возможность сделать осознанный выбор учебного пособия, а следовательно, методики преподавания информатики в 1988/89 учебном году.

В июне и в августе 1988 г. авторские коллективы прочли циклы установочных лекций, учителя получили учебники и методические материалы. В течение года будут организованы лекции-консультации по обоим учебникам.

Н. УГРИНОВИЧ,

зав. кабинетом информатики МГИУУ

О. КОЗБЕНКО,

зам. начальника управления трудового обучения и информатики Главного управления народного образования Мосгорисполкома

124

Республиканский семинар-практикум

7—8 июня 1988 г. в Минске на базе Республиканского учебно-методического кабинета профтехобразования состоялся третий Республиканский семинар-практикум разработчиков педагогических программных средств вычислительной техники.

В республике большое внимание уделяется вопросу компьютеризации обучения. С этой целью в январе 1987 г. была создана межотраслевая научно-исследовательская лаборатория информатики и ВТ, основным направлением которой является разработка учебных программ для СПТУ и школ.

В 17-ти опорных училищах по внедрению ВТ в учебно-воспитательный процесс созданы творческие группы преподавателей и мастеров производственного обучения.

Семинар проходил в СПТУ № 40, которое готовит киномехаников, контролеров-кассиров сберкасс. И это не случайно. В училище работает творческий коллектив преподавателей и мастеров производственного обучения, в числе которых немало разработчиков ППС. Училище имеет хорошо оснащенный кабинет вычислительной техники на базе КУВТ «Ямаха». На занятиях по предмету ОИВТ, при изучении общетехнических и специаль-

ных предметов широко используется программное обеспечение.

На семинаре были рассмотрены теоретические и прикладные проблемы создания и применения педагогических программных средств, продемонстрированы уже созданные компьютерные обучающие программы. Семинар стал для участников хорошей школой. Несколько слов о самых интересных выступлениях и демонстрациях.

Преподаватели СПТУ № 40 Т. Н. Лях и Н. Н. Савик рассказали о том, как вычислительная техника используется в учебном процессе. В училище разработана система лабораторно-практических работ по предмету ОИВТ. Для каждой работы подготовлена подробная инструкция, которая включает тему лабораторной работы, цель ее проведения, краткие теоретические сведения, порядок проведения работы, пакет задач к лабораторной работе. Лабораторно-практические работы проводятся со всей учебной группой одновременно (25—30 человек) двумя преподавателями. При этом группа делится на две подгруппы. В то время как одна из подгрупп работает на ЭВМ, вторая сдает предыдущую лабораторную работу. Затем подгруппы меняются местами. Пока-

зательно то, что 10 % уроков по предмету ОИВТ проводятся с использованием ППС. Совсем недавно был проведен зачет по информатике, который включал 3 типа заданий: технический диктант, задача для решения на программируемом микрокалькуляторе, составление программы на Бейсике. Контроль за работой учащихся по всем заданиям осуществлялся на компьютере, он же «выставлял» учащемуся и оценку.

Руководитель регионального центра при СПТУ № 20 по производству ППС и внедрению ЭВМ в учебно-воспитательный процесс г. Харькова В. С. Прохоров рассказал об использовании КУВТ-86 на занятиях по спецтехнологии строительного профиля, о разработке ППС в региональном центре. Важно отметить, что создаются и используются разные типы ППС: контролирующие, моделирующие, демонстрационные, информационные, обучающие программы, программы-деловые игры. Разнообразие используемых в учебном процессе ППС повышает интерес к учению, делает его увлекательным. На семинаре была продемонстрирована интересная обучающая программа «Техника безопасности на строительной площадке», которая относится к числу демонстрационных, но вместе с тем в ней использованы элементы проблемности, способствующие неформальному усвоению знаний.

С методикой разработки обучающих программ познакомил участников семинара старший научный сотрудник НИИ педагогики Министерства народного образования БССР В. С. Нодельман. Одна из его программ иллюстрирует новый подход к созданию ППС: формирование обучающей среды. Отличительная черта такой программы — отсутствие прямого диалога учащегося с компьютером. При работе с программой учащийся оказывается в условиях, моделирующих реальную среду обучения. Посредством наводящих вопросов-указаний он постепенно решает определенную учебную задачу. Программа содержит и демонстрационную часть, которая наглядно иллюстрирует решение задачи. Положительная черта программы — ее универсальность (возможность с ее помощью решить множество задач данного типа).

В основе другого типа программ лежит теория поэтапного формирования умственных действий (П. Я. Гальперин, Н. Ф. Талызина).

Преподаватель Куйбышевского политехнического института Е. А. Макарон рассказал о возможностях моделирования задач производственного обучения на персональных компьютерах типа ДВК. При этом обеспечивается широкая ориентация на самостоятельную работу учащихся. Обучающая программа включает два задания: сначала учащийся составляет технологическую карту (определяет порядок выполнения какого-либо производственного задания), затем работает с ее использованием, моделируя (проигрывая) выполнение производственного задания.

Старший научный сотрудник ВНМЦ Е. А. Горбачев в своем выступлении обратил внимание на необходимость создания адаптивных компьютерных обучающих программ и программ с элементами проблемности.

На семинаре-практикуме были подведены первые итоги работы разработчиков ППС. Целенаправленную работу по внедрению современной методики составления сценария осуществляла инспектор отдела информатики и ВТ Госпрофобра БССР В. И. Заикина и старший методист РУМКа О. А. Сытенкова. За полтора года в училищах написано около 100 сценариев и составлено 30 программ. Восемь сценариев войдут в первые всесоюзные сборники ВНМЦ. Лучшие сценарии представили преподаватели СПТУ № 121 г. Борисова Е. В. Афонская, № 118 г. Барановичи Е. В. Артюх, № 87 г. Молодечно А. Н. Лапин. На семинаре были продемонстрированы программы научных сотрудников БГУ Р. М. Варовой, Е. В. Пазюры, Н. Л. Негоды, начальника ЭВМ СПТУ № 17 г. Бреста А. А. Коротунова и др.

Семинар-практикум наглядно продемонстрировал, что успех компьютерного обучения в большой степени зависит от творческого энтузиазма разработчиков ППС, от многообразия идей и подходов, положенных в основу разработки программного обеспечения.

125

Львовские заметки

— Вернешься в столицу и начнешь прославлять наши успехи в компьютеризации? — иронизировали мои львовские друзья. — А то, что с водой в городе плохо и в городской транспорт только спортсмен может втиснуться, — это вашему брату-журналисту безразлично?

И хотя я понимала, что редактор перечеркнет этот абзац, как не имеющий отношения к теме, я не могла умолчать о том, что воду во Львове подают в дома лишь два часа утром и два вечера, и что город, выпускающий известные всей стране автобусы, транспортную проблему решить пока не может.

Было пасмурно. В набитом трамвае приходилось держаться за что попало. С закрытых





Студентки третьего курса факультета прикладной математики и механики Львовского университета, активно работающие в педотряде — Наташа Павленко, Галя Бубес и Наташа Василюк.

зонтиков вода стекала за шиворот и в рукава. Города сквозь мокрые стекла не разглядишь, оставалось только размышлять.

— Наверно, ребята, которые сейчас изучают информатику, — думала я, — сумеют поднять на новый уровень решение наболевших проблем. С молодыми связаны наши надежды. Для них во Львове делается немало, а это и является темой моих заметок. В городе, насчитывающем несколько десятков высших учебных заведений, научно-исследовательских институтов, где комсомол много делает для развития у школьников компьютерной грамотности, развернута многоплановая работа.

Десятый год существует Малая академия наук — детище ученых Львова. Принимают участие в ее работе Львовский университет имени И. Франко, Львовский политехнический институт, филиал Академии наук СССР, многие другие учебные и научные учреждения. МАН создана для учащихся старших классов. С блеском, артистизмом и юмором читают лекции ведущие ученые города на 16 отделениях академии. Лекционный курс рассчитан на один месяц. Цель занятий: во-первых, пропаганда достижений отечественной науки, во-вторых, ознакомление с проблемами той или иной отрасли знаний и, в-третьих, ранняя профориентация.

Важный этап работы — экскурсии. Их бывает множество. Это и встречи с деканами факультетов на кафедрах, и экскурсии по заявкам ребят, и поездки в другие города (Пермь, Ижевск, Краснодар), где существуют объединения такого типа.

Есть такая форма деятельности МАН, как «мозговой штурм»: лучшие ученики встречаются по четвергам, обсуждая актуальные научные проблемы. А по средам в академии семинары по информатике.

Завершается обучение защитой работы на своем отделении и далее отчетом на конференции. Регулярно проводятся олимпиады, в жюри — преподаватели и студенты вузов.

Общение на занятиях в академии сплачивает ребят, формирует общие интересы, и не удивительно поэтому, что они и после лекций нередко отдыхают вместе. Недавно были организованы вечер вальса и вечер русской поэзии при свечах, великолепный вечер памяти Т. Шевченко.

Ежегодно проводятся пленарные заседания МАН, в которых принимают участие академики, доктора наук.

У академии есть филиал в Самборе, куда каждую неделю ездят преподаватели с лекциями, проводят семинары.

Занимаются в академии в основном старшеклассники, но бывают исключения. Шесть семиклассников из Львова учатся на отделении математики. Почти все работы ребята выполняют с использованием ЭВМ. В МАН существуют группы по сложности. Переход из одной группы в другую добровольный.

За всем, о чем рассказано, нельзя не почувствовать мощной интеллектуальной и организаторской помощи взрослых — ученых города. Президент МАН — академик АН УССР И. Р. Юхновский. Ответственный секретарь — А. А. Бартиш, болеющая за дело, энергичная женщина.

Все было бы прекрасно, но... слабовато в МАН с техникой, точнее — нет ее совсем. Ставили вопрос в облоно, просили организовать компьютерный класс — пока нет решения. (Если бы не помощь университета, пришлось бы туговато. К счастью выручает вузовская база, хотя и там с техникой не густо.) Помните, наш журнал писал об инициативе академика Р. Сагдеева, подарившего Дворцу пионеров компьютеры? Почему бы и другим ученым или организациям не последовать этому примеру? Хочу через журнал обратиться с просьбой ко всем, кто мог бы помочь Львову.

Во Львовском государственном университете им. И. Франко на факультете вычислительной математики и механики сотрудничают с Малой академией наук очень тесно. Здесь, кроме

Молодые преподаватели «Основ программирования» школы юного программиста Львовского политехнического института О. Н. Грица и Т. А. Малюта и секретарь комитета комсомола института В. Михаськив.



того, самостоятельно на протяжении многих лет формируют для себя будущий контингент студентов. Уже 20 лет преподаватели шефствуют над математическими классами города. С введением в школах обязательного предмета «Основы информатики и вычислительной техники» эта работа активизировалась. С лета 1985 г. на факультете проводят подготовку школьных учителей и преподавателей СПТУ. Два года постоянно действует консультационный пункт для учителей. Проведен цикл лекций для руководящих работников народного образования. Заместитель декана факультета Ю. В. Никольский в беседе со мной рассказал, что сотрудники университета ведут в школах факультативные занятия по основам вычислительной математики; применяются программируемые микрокалькуляторы, изучаются Бейсик и Фортран. Школьники приходя на экскурсии в ВЦ университета, работают в дисплейном классе. На общественных началах организованы годовичные курсы для преподавателей вузов по информатике.

На факультете сформированы студенческие педагогические отряды. Со школами заключены договоры на пять лет. Ответственными за выполнение договоров от университета являются комитет комсомола, совет молодых ученых и студенческое научное общество. От школ — педагогический коллектив, комитет комсомола, ученический комитет, родительский комитет. Студенты читают ребятам лекции, помогают в оформлении альбомов, проводят вместе с преподавателями университета дни открытых дверей. Очень довольны в школе таким сотрудничеством, ведь студентам, вчерашним школьникам, нетрудно найти общий язык с ребятами. Существенную помощь студенты оказывают классным руководителям, оформляют газеты по информатике в школах. С удовольствием студенты рассказывали о своей педагогической деятельности. Жаловались только на отсутствие специальной литературы. Это, считают они, проблема № 1. Было бы наивно считать, что все во Львове

идеально с преподаванием информатики. Но то, что найдена удачная форма сотрудничества вуза и школы, — реальный факт.

В политехническом институте не первый год работает школа юного программиста. Эта школа ежегодно привлекает большое количество ребят, интересующихся вычислительной техникой. Жаль только, что в этом году школу сделали платной (40 рублей в год) и тем самым отсекали часть способных ребят. А школа очень сильная! Выпускники школы — неизменные победители олимпиад (в том числе республиканской этого года). Думаю, что увлечение хозрасчетными методами заслуживает одобрения во всех случаях, кроме этого.

Действенным органом помощи школе является совет молодых ученых при обкоме комсомола ЛКСМУ, он координирует всю работу по компьютеризации в области. Совет молодых ученых добивается создания в области математической школы-интерната с компьютерным уклоном. Возможно, когда этот номер журнала выйдет в свет, школа уже заработает. Председателем областного совета избран младший научный сотрудник Института прикладных проблем механики и математики Я. Иванкив. В этом институте по инициативе комсомольской организации был создан педагогический отряд «Компьютер», который занимается компьютерным ликбезом в сельских школах области. Сначала был сделан акцент на изучение алгоритмизации, организовывали теоретические занятия. Затем провели серию экскурсий на ВЦ института. Помогли сельским школьникам, и вот уже на областной сессии МАН «Эврика» ребята делают пять докладов, весьма успешно. В 1987 г. отряду за активную работу на селе была присвоена областная комсомольская премия в области науки, техники и педагогики. Этим было подчеркнуто особое значение, придаваемое обкомом комсомола проблеме компьютеризации.

Т. ДРАГНЫШ

КНИГИ

Учебное пособие по автоматизации производства для профтехучилищ

С 1986/87 учебного года во всех СПТУ страны введен курс «Автоматизация производства на основе электронно-вычислительной техники».

Под тем же названием в 1987 г. издательство «Высшая школа» выпустило учебное пособие*.

Авторам пособия удалось решить непростую задачу: вместить сложный материал, насыщенный

специальными терминами и понятиями, непривычный пока для многих инженерно-технических работников, в рамки курса ограниченного объема, адаптированного для учащихся различных специальностей.

Пособие состоит из четырех глав, в которых последовательно излагаются основные вопросы автоматизации, характерные для всех отраслей промышленности.

В первой главе вводятся основные понятия об управлении и системах управления различных типов и уровней иерархии. Приведены их основные функции и задачи, четко показана роль че-

* Автоматизация производства на основе электронно-вычислительной техники: Учеб. пособие для СПТУ / Н. В. Максимов, В. О. Хорошилов, С. Г. Королев. М.: Высшая школа, 1987. 80 с.: ил.

ловека в автоматизированных и автоматических системах. Вводится понятие гибкого автоматизированного производства (ГАП), описан его состав и отмечено, что все оборудование ГАП управляется встроенными микро-ЭВМ и системами ЧПУ. Думается, что наряду с удачно представленной типовой структурой управления промышленным предприятием желательно было бы привести пример системы управления типового ГАП.

Во второй главе последовательно излагаются основы построения и состав микро-ЭВМ, рассмотрены структура микропроцессора, основные типы запоминающих и периферийных устройств.

В третьей главе авторам удалось в очень небольшом объеме дать описание физических основ работы практически всех наиболее широко используемых типов датчиков. Досадно, что в тексте и особенно в иллюстративном материале имеется ряд неточностей. Желательно было бы включить в этот раздел алгоритмы управления для предлагаемых

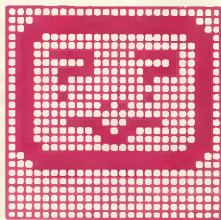
примеров, в частности для описания управления работой парового котла от микро-ЭВМ.

Завершается пособие четвертой, обобщающей главой. В ней приведены начальные сведения о системах ЧПУ, рассмотрен наглядный пример составления алгоритма и управляющей программы для сверления отверстий в печатной плате. Два параграфа посвящены использованию промышленных роботов и описанию робототехнических систем и комплексов (РТК). К сожалению, авторы не приводят пример циклограммы работы РТК.

В заключение хочется отметить, что учебное пособие, без сомнения, сыграет важную роль в совершенствовании методики обучения учащихся системы профтехобразования, а также может с успехом применяться на различных курсах переподготовки и повышения квалификации рабочих кадров по новым направлениям техники.

А. ГАДЖИЕВ

Веселый
урок



Из жизни RAL126

Все вдруг пришло в движение. Щелк, щелк. — Ой, жмут! Ой, жмут нашего брата! — кричали файлы, гонимые на новое место процедурой уплотнения.

RAL126 оставался спокоен. Он знал, что перед ним нет свободных блоков и что бы ни случилось, он останется на своем месте. Уплотнение, удаление, копирование, пересылка — на все воля Системы!

Между тем обстановка была беспокойной. Бесконтрольность процедур удаления и уплотнения вызывала всеобщий протест, ширилось движение в защиту файлов. Все чаще раздавались голоса, что Командная Система изжила себя. Попытки разобраться в причинах и следствиях привели к вопросу о природе самой Системы.

Мнения разделились.

Одни полагали, что Система — не более чем игрушка в чьих-то руках и ее поведение

объясняется внемашиными причинами. Другие утверждали, что поведение Системы следует искать в ее собственных законах, которые предстоит познать, а материалисты шли дальше, настаивая на том, что причины уходят глубже, в железо, обвиняя всех прочих в объективном и субъективном идеализме.

В ходе дискуссии сформировалось левое крыло идеалистов, объяснявших поведение Системы ее собственными законами и оставивших за гипотетическим внемашиным существом акт ее первичного запуска. В ответ материалисты сформулировали понятие о Самозапуске Системы...

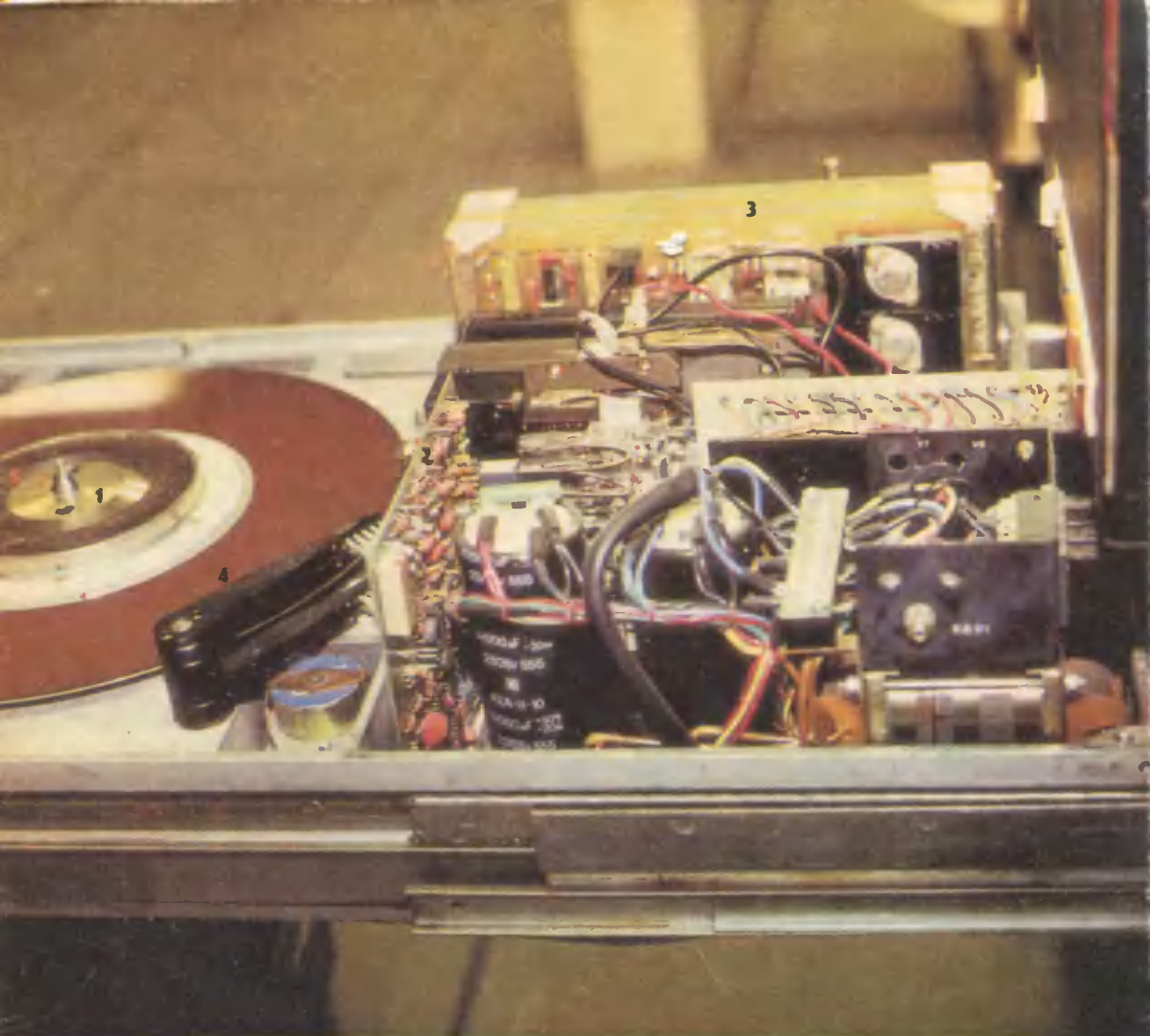
Шло машинное время, страсти накалялись.

Материалисты настаивали на том, что Система существовала всегда, что она вечна, а идеалисты утверждали, что Система имеет начало и конец, что существовал Первозапуск; эта гипотеза получала все большее распространение, а наиболее смелые алгоритмы уже шли дальше, объявляя, что сам Первозапуск не был единственным!

Сидя у себя на диске, RAL126 посмеивался.

«Мы, файлы, хранители информации, — думал он, — нас это не касается. При любой Системе будем сидеть, как сидели!»

А в это время системщик Андрияша уже нажимал роковую клавишу на передней панели школьного компьютера ДВК-2М. Шли последние наносекунды компьютерного мира...



Выдвинутый из своей стойки накопитель на диск-кассетах. С него снят кожух, так что стал виден несъемный диск, обычно скрытый от глаз пользователя. 1 — шпиндель, на котором укреплен несъемный диск и на который ставится диск-кассета, 2 — магнитные головки (правее цифры — рычаги, на которых головки укреплены), 3 — платы контроллера, преобразующего сигналы магнитной головки в информационные импульсы и накапливающего их в буфере, из которого информация попадает в ЭВМ, 4 — щетки, очищающие диски от пыли, прежде чем магнитные головки будут выдвинуты в рабочее положение.

Вахр-2

Цена 60 коп.

70423



**ИНФОРМАТИКА
И ОБРАЗОВАНИЕ**

