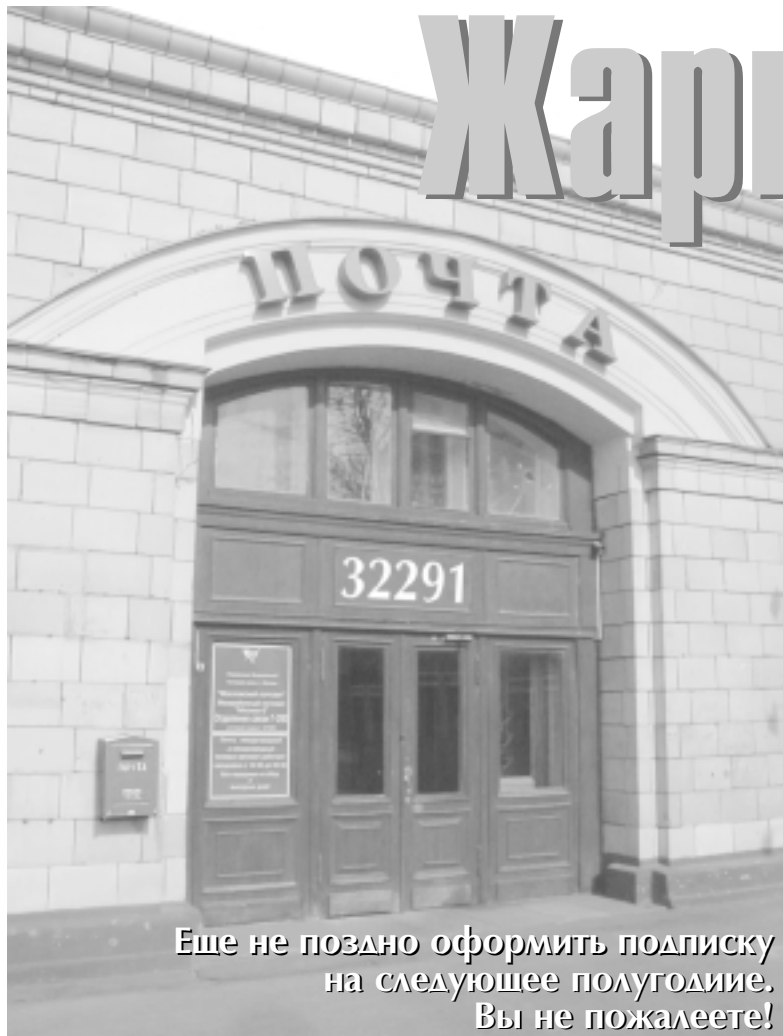


ИНФОРМАТИК

Электронные версии газеты «Первое сентября» и приложений <http://www.1september.ru>

Жаркое лето 99-го



Еще не поздно оформить подписку на следующее полугодие. Вы не пожалеете!

С июня по август в «Информатике» выйдет 12 спецвыпусков! Среди них — специально переработанные «Интернет для начинающих», «Турбо Паскаль», «Как это делаю я» и др.

К новому учебному году мы готовим учебник «Информатика в виртуальной школе», новое издание методического пособия для учителя «Земля Информатика», планируем издать материалы по изучению языка Visual Basic и др.

Напоминаем, что

**КАТАЛОЖНАЯ ЦЕНА
МЕСЯЧНОЙ ПОДПИСКИ НА НАШУ ГАЗЕТУ
ВСЕГО 24 РУБЛЯ!**



Конечно, за почтовые услуги с вас возьмут несколько больше, но все равно никогда и нигде вы не найдете такое количество методических материалов такого качества!

Жители Москвы и Московской области могут оформить подписку с получением газет в редакции. Месячная цена такой подписки 22 рубля. И никаких других расходов! (Конечно, такую подписку могут оформить не только жители московского региона. Единственный вопрос — получение газет. Но и его можно решить — газеты будут храниться в редакции, и вы сможете забирать их в любое удобное для вас время в течение полугодия.) **Постоянным подписчикам предоставляются скидки.**

Подписной купон на 16-й стр.

№	32291
Имя	
Фамилия	
Адрес	
Подпись	

Очерки истории информатики в России

Книга с таким названием вышла в конце 1998 г. в Новосибирске (в научно-издательском центре ОИГМ РАН). Ее редакторы-составители — академик РАН Д.А. Поспелов и доктор технических наук Я.И. Фет. Издание осуществлено при поддержке Российского фонда фундаментальных исследований.

Сам термин «информатика» для нас сравнительно новый — он появился в 80-х годах как название совокупности естественных наук, изучающих процессы передачи, обработки и хранения информации. В рамках информатики объединяются научные направления, тесно связанные с появлением компьютеров и проникновением их во все сферы деятельности человека. И вполне закономерен вопрос: откуда есть пошла информатика?

Прародительницей информатики является кибернетика. Ее предмет — общие закономерности строения управляющих систем и течения процессов управления. Изучение таких закономерностей ведет к построению теории способов хранения, передачи и переработки информации в машинах и живых организмах. Этой ролью кибернетики объясняется повышенное внимание к истории ее становления и развития.

Рецензируемая книга обращается к узловым моментам ранней истории кибернетики в нашем отечестве, которая охватывает период от середины 50-х до середины 70-х годов. Общая картина этого двадцатилетия дана во вступительной статье Д.А. Поспелова*.

Непосредственно ранней истории кибернетики посвящен первый раздел книги. Она представлена в виде документов того времени, статьи М.Г. Гаазе-Рапопорта (организатора первого в стране семинара по истории кибернетики), а также статей, авторы которых обращаются к личности и деятельности отца советской кибернетики — члена-корреспондента АН СССР Алексея Андреевича Ляпунова. Все началось с борьбы научной общественности за официальное признание кибернетики, на которой в то время висел ярлык буржуазной лженауки, препятствующий исследованиям в этой области. Общественное движение «за кибернетику» возглавили ведущие ученые нашей страны. Кроме А.А. Ляпунова, среди них были академики Сергей Львович Соболев и Андрей Николаевич Колмогоров. В книгу вошли материалы, сыгравшие поворотную роль в этой борьбе. Завершающими актами ее стали: опубликование в БСЭ статьи А.Н. Колмогорова «Кибернетика», раскрывающей истинную суть этой науки, и учреждение Совета по кибернетике при Президиуме АН СССР (1959 г.). Совет возглавил академик Аксель Иванович Берг. Так появился официальный орган, заботящийся о создании в стране научных уч-



реждений, работающих в области кибернетики. Как проходил этот процесс, можно узнать из материалов книги.

Когда кибернетика приобрела статус официально признанной науки и более того — науки, развитие которой необходимо для укрепления государства в целом, в ней начали выделяться самостоятельные направления исследований. В книгу включены материалы, относящиеся к трем из них — компьютерной лингвистике, кибернетическим вопросам биологии и экономической кибернетики (разделы II—IV).

Под компьютерной лингвистикой понимается область, охватывающая структурную и математическую лингвистику, семиотику и машинный перевод. В разделе II помещены материалы, знакомящие с историей становления этой области.

Раздел III «Кибернетические вопросы биологии» открывается яркой статьей Н.В. Тимофеева-Ресовского (выдающегося русского генетика, эволюциониста, биофизика, радиобиолога) и А.Г. Маленкова (специалиста в области физики и онкологии). Она дается в перепечатке. Статья посвящена А.А. Ляпунову. В ней с любовью описана личность А.А. Ля-

пунова и дана высокая оценка его научного наследия в области биокибернетики.

В этом же разделе раскрывается сущность вклада в биокибернетику, сделанного членами-корреспондентами АН СССР Николаем Александровичем Бернштейном и Игорем Андреевичем Полетаевым (автором «Сигнала»). Памяти этих трех ученых посвящены и другие статьи раздела, написанные специально для данной книги.

В разделе IV «Экономическая кибернетика» опубликованы перепечатки статей и доклада академика Леонида Витальевича Канторовича, лауреата Нобелевской премии по экономике, одного из основателей экономической кибернетики. Здесь же статьи о личности Л.В. Канторовича и его вкладе в различные области науки.

Предпоследний раздел V носит название «Биографические материалы» и содержит воспоминания современников об А.А. Ляпунове, А.Н. Колмогорове, Л.В. Канторовиче, И.А. Полетаеве, А.П. Ершове, А.И. Берге, М.Л. Цетлине, М.М. Завадском, П.П. Лазареве. Повествование о научных достижениях перемежается с фактами из биографии и воспоминаниями о личности этих ученых.

Безусловный интерес представляет и раздел «Приложения», которым завершается книга. Здесь помещены и статья из «Краткого философского словаря» 1954 г. издания, объясняющая кибернетику реакционной лженаукой, и статья А.Н. Колмогорова «Кибернетика» из 52-го тома БСЭ, и другие документы периода борьбы за кибернетику.

Книга является документом целой эпохи и свидетельствует о том, что в ранний период становления информатики наша страна была пионером во многих ее научных направлениях и разработках.

Р.И. ПОДЛОВЧЕНКО

* Статью Д.А. Поспелова читайте на с. 7—10.



Задача олимпиады института Вейцмана

Три пары монет

Первая пара — золотые, вторая — серебряные, третья — бронзовые. В каждой паре одна монета хорошая, другая — фальшивая. Все хорошие монеты одинакового веса, все фальшивые — тоже одинакового. Фальшивые монеты легче хороших.

Имеются весы с двумя чашками. Они показывают: одинаковые ли грузы на чашках, и если нет, то на какой груз больше.

Найти хорошие монеты за 2 (два) взвешивания.

Решение задачи читайте на с. 16

НАШИ ДЕТИ БУДУТ ЖИТЬ В XXI ВЕКЕ

2 3

КРУГЛЫЙ СТОЛ

• “РЕБЯТА, ДАВАЙТЕ ЖИТЬ ДРУЖНО”

Ю.А. ШАФРИН

“...Когда вы выбираете тот или иной учебник, вы должны, на мой взгляд, исходить в первую очередь из целей учебника, сопоставляя их с теми целями, которые перед курсом информатики ставите вы...”

Продолжение дискуссии “О судьбах школьной информатики” (см. также № 16, 21, 33, 44/98; 1, 7/99).

4 5 6

ЭКЗАМЕНЫ В ВУЗЫ

• ЛОГИЧЕСКИЕ ЗАДАЧИ НА ВСТУПИТЕЛЬНЫХ ЭКЗАМЕНАХ ПО ИНФОРМАТИКЕ

В.И. РАКИТИН

Окончание. Начало в № 17, 18/99.

“Методический обзор”. Содержит практически все типы задач по элементам логики, которые предлагались на вступительных экзаменах по информатике в различных вузах страны до настоящего времени.

В этом номере представлены три раздела: “Преобразование логических функций (выражений)”, “Текстовые логические задачи”, “Построение логической функции по заданной логической схеме”.

7 8 9 10

ВЕК ИНФОРМАТИКИ

• СТАНОВЛЕНИЕ ИНФОРМАТИКИ В РОССИИ

Д.А. ПОСПЕЛОВ

Согласно определению, данному в Большой советской энциклопедии (3-е изд., т. 10, М.: Советская энциклопедия, 1972), информатика — это “дисциплина, изучающая структуру и общие свойства научной информации, а также закономерности ее создания, преобразования, передачи и использования в различных сферах человеческой деятельности”. Однако с начала 80-х годов в нашей стране (и не только в ней) информатикой стало называться совсем иное...

11 12 13 14

ЗАДАЧИ

• ЗАДАЧИ ПО ПРОГРАММИРОВАНИЮ

Ю.А. СОКОЛИНСКИЙ

Окончание. Начало в № 18/99.

Задачи для работы в классе на темы: “Массивы”, “Символьные строки” (в первой части статьи предлагались задачи на тему “Ветвления”).

Приводятся решения на языках: Паскаль, Бейсик (QBASIC), Си.

15 16

ВОКРУГ ПРЕДМЕТА

• ВИРТУАЛЬНАЯ ШКОЛА-2

А.И. СЕНОКОСОВ

Окончание. Начало в № 18/99.

Постоянный автор нашей газеты — об опыте организации “Виртуальной школы” (появление которой вызвало резкое повышение интереса детей к информатике), основанной на сетевых технологиях и свободе творчества детей. Описываемая “Виртуальная школа” организована в самой обычной средней школе № 104 г. Екатеринбург.

16

ЗАДАЧА

• ТРИ ПАРЫ МОНЕТ

ЗАДАЧА ОЛИМПИАДЫ ИНСТИТУТА ВЕЙЦМАНА

Логическая задача, связанная с выявлением (путем взвешивания) фальшивых монет.

Активная дискуссия о судьбах школьной информатики, начатая в апреле прошлого года газетой “Информатика” и продолженная на международных конференциях в г. Троицке, “ИТО’98/99”, на других очных и заочных встречах педагогов и авторов, на мой взгляд, не только не зашла в тупик, но и способствовала быстрому развитию взаимопонимания между специалистами по информатике. Мне очень понравилась мысль, высказанная А.Г. Кушниренко: авторы учебников, по сути, должны поддерживать друг друга, а не заниматься поисками недочетов, из коих, собственно, и состоят все действующие учебники по информатике. В какой-то степени созвучны этой мысли и четко сформулированные тезисы из “12 лекций о том, для чего нужен...” (А.Г. Кушниренко и Г.В. Лебедев, “Информатика”, № 1/99): “...Я не говорю, что цель (нашего. — Ю.Ш.) курса — научить школьников работать за компьютером, обучить их языкам программирования, современным базам данных или электронным таблицам. Все это полезные и важные вещи, но наш курс построен совсем по-другому и нацелен совсем на другое — на развитие в каком-то смысле интеллекта ученика, его умения думать.

...Когда вы выбираете тот или иной учебник, вы должны, на мой взгляд, исходить в первую очередь из целей учебника, сопоставляя их с теми целями, которые перед курсом информатики ставите вы.

Если, например, вы считаете, что вам надо, чтобы ученики знали Бейсик и умели работать за компьютерами, пользоваться базами данных или электронными таблицами, то наш учебник вам почти бесполезен”.

“Ребята, давайте жить дружно!”

Буквально те же соображения (выраженные другими словами) можно найти и во Введении к моему пособию “Информационные технологии” (М., “Лаборатория базовых знаний”, 1998). Возможно, я ошибаюсь, но мне кажется, многие идеологи и авторы близки к осознанию того факта, что пора отказаться от попыток декларировать свой подход как “единственно верное учение”, не противопоставлять пособия друг другу, а лишь тщательно разрабатывать и объяснять учителям свою концепцию. (Из сказанного вовсе не следует, что я призываю отказаться от заслуженной критики в адрес своих коллег — например, такой изящной, доброжелательной и блестяще аргументированной, как в статье А.Г. Гейна “Информатика” и “Земля Информатика” в 3-м номере нашей газеты, — без критики вообще нет продвижения вперед.)

Правда, оптимизм лозунга, вынесенного в заголовок этой заметки, вступает в некоторое противоречие с довольно жесткой позицией Минобразования РФ, которое, похоже, решило положить конец “самоуправству” авторов, предлагающих то один, то другой предмет под псевдонимом “Информатика”, и обязать всех следовать “Обязательному минимуму содержания по информатике”. Впрочем, это противоречие, на мой взгляд, преодолимо, так как ответить на *все* (или почти все) вопросы *минимума*, как сейчас окончательно выяснилось, вполне можно в рамках *различных* подходов и концепций (примеры тому — и учебники проф. Н.В. Макаровой, и базовый курс И.Г. Семакина с коллегами, и, как я надеюсь, мое пособие “Информационные технологии”).

Хотя я не во всем согласен с подходами Н.В. Макаровой и И.Г. Семакина (особенно в части ИТ), хотя я вижу огрехи и в своем пособии (несмотря на постоянные усилия их устранить), в целом мне кажется, что можно констатировать некоторое сближение точек зрения и надеяться на дальнейшее совершенствование и даже унификацию курса, даже на создание “канонического” учебника.

Тем не менее вновь вынужден обратиться к наиболее острому для себя вопросу: как выстроить преподавание современных информационных технологий в рамках курса, продиктованного минимумом? Пока, как мне представляется, в этом вопросе разноречивых мнений слишком велик (в отличие от базовой, фундаментальной или “инвариантной” части, по поводу которой со-

гласие практически достигнуто). Те решения, которые предлагаются нам и частично уже реализуются (в частности, категорическое требование устранить из курса “конкретные программные продукты”), мне кажутся весьма спорными. Вряд ли из этого что-то получится. “Конкретные программные продукты” — это не что иное, как синоним со-

временной объектно-ориентированной технологии как целостной системы, в рамках которой можно легко ввести и раскрыть фундаментальные понятия информатики: “информация и данные”, “система и структура”, “объект”, “модель”, “алгоритм”, “численный эксперимент” и т.д. Более того, эти понятия при таком подходе как бы автоматически наполняются конкретным содержанием и легко усваиваются учащимися.

К сожалению, именно эта целостность, так способствующая развитию системного, логического, даже алгоритмического мышления (я имею в виду возможность управлять функциями приложения с помощью аппарата макросов), чаще всего игнорируется. Мне никак не удается доказать (например, это видно по упомянутым выше учебникам), что где-то в крупном разделе, крупной главе или хотя бы в большом параграфе обязательно надо четко изложить основы гра-

2

фического интерфейса пользователя как единой триады инструментальных средств, как единого механизма, той самой “технологии решения задач на ЭВМ”, которая, собственно, и готовит школьника к решению основных задач (именно основных, все охватить невозможно!) — от подготовки текстов до решения интегральных уравнений, анимации и навигации в Internet. Учащиеся должны знать “таблицу умножения” операционной среды.

Тем не менее в одном из отзывов на мою книгу меня весьма жестко упрекнули в “переписывании инструкций пользователям конкретных продуктов”, упорно не замечая того, что основная задача пособия — описать не продукты, а *принципы*. Например, едва ли не вся глава 14 — это не инструкция к Access 2.0, а попытка показать “инвариантные” составляющие процесса создания и эксплуатации настольных информационных систем. При таком подходе меня совершенно не интересует, что в окне Access 7.0 названия вкладок (**Таблица**, **Запрос** и т.п.) расположены по горизонтали, а в Access 2.0 — по вертикали, или в Access 7.0 можно еще украсить форму каким-нибудь рисунком, да мало ли еще что! И когда я читаю в том же отзыве, что “**теоретическое** (выделено мной. — Ю.Ш.) рассмотрение базовых принципов современных информационных технологий заменяется изучением конкретных программ...”, никак не могу понять, о чем же идет речь (то ли о логических основах ЭВМ, то ли об изучении системы прерываний...). Да и можно ли “теоретически” изучать технологию?

С горечью приходится констатировать многочисленные ошибки и неточности при изложении ИТ, когда авторы пытаются дистанцироваться от “конкретных программных продуктов”, перемешивая все и вся. (Я считаю неэтичным называть конкретных авторов, так как изложенные ниже в качестве примеров представления характерны для многих, в том числе для весьма квалифицированных педагогов.)

Вот в одном из пособий по ИТ ведется рассказ о текстовых файлах, с которыми работает некий текстовый редактор (ТР). “...Обмен в ВЗУ возможен в двух направлениях: сохранение (запись, или *save*) и загрузка (считывание, или *load*) текста, который при хранении на ВЗУ называется файлом (текстовым файлом). При этом указывается путь к необходимому каталогу...”. Далее, в частности, утверждается, что мощный текстовый процессор позволяет форматировать “текстовый файл” шрифтами TrueType (Arial, Times New Roman и т.п.). К сожалению, из этого изложения ясно, что авторы, во-первых, не усматривают принципиальной разницы между ТР, создающими текстовые файлы, и ТР, создающими текстовые документы (т.е. файлы совсем другого формата). Во-вторых, в частной операции весьма невнятно описан лишь один из универсальных механизмов современной операционной среды — сохранение и загрузка файловых документов, — который одинаково работает во всех приложениях, независимо от формата документа (TXT, DOC, XLS, MDB и т.п.). Под этим углом зрения файл TXT отличается от файла DOC (текстового документа) точно так же, как он отличается от электронной таблицы или от файла базы данных (во всех перечисленных файлах имеются текстовые фрагменты).

Многие авторы объясняют неточности в своих определениях необходимостью адаптации сложных понятий информатики к уровню и возможностям среднего школьника. Однако адаптация не должна искажать фундаментальный смысл понятия, иначе она лишь закрепит на долгие годы неверное представление учащегося о предмете (кстати, искусству адаптации можно поучиться у выдающегося популяризатора Я.И. Перельмана). Например, часто можно встретить такую “адаптацию” понятия “пиксель”: “Пиксель состоит из трех зерен люминофора, каждое из которых светится зеленым, синим или красным цветом...” Ошибочность такого определения заметит любой человек, знакомый с элементарной логикой: если монитор может отображать по горизонтали, скажем, 640 или 1024 пикселя (по нашему выбору), то каким образом при изменении разрешения перемещаются по экрану зерна люминофора? Здесь смешаны разные вещи — физические характеристики экрана и функции совсем другого устройства — видеоадаптера.



Мне кажется, при разработке разделов, связанных с ИТ, ни в коем случае нельзя игнорировать бесспорный и принципиальный факт: ныне существуют устаревшая *процедурная* среда MS-DOS, приложения которой действительно далеки от стандарта, и современная *непроцедурная* объектно-ориентированная среда с единым графическим интерфейсом пользователя.

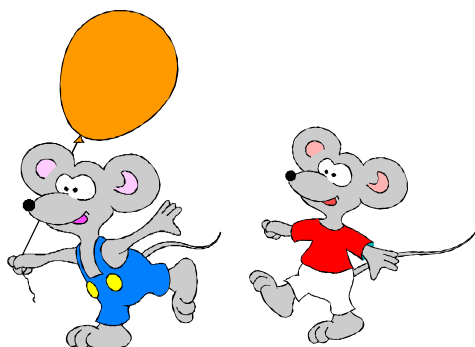
Трудно сказать, в какой мере устаревшие приложения MS-DOS (текстовые редакторы, SuperCalc, dBASE и т.д.) должны быть отражены в школьной программе. С одной стороны, “элитные” специалисты решительно отвергают их (об MS-DOS надо забыть!); с другой стороны, эти приложения по-прежнему широко используются там, где еще нет ни сил, ни средств поспевать за рыночным напором Microsoft и ей подобных (да и графическая среда опирается на файловую систему и другие понятия, восходящие к MS-DOS). Однако ясно, что приложения MS-DOS ни в коем случае нельзя считать равноправной альтернативой объектно-ориентированной среде, которая ныне обрела целостность и вывела ИТ на принципиально иной уровень.

Я уже высказывался по поводу едва ли не самого распространенного заблуждения по поводу современных ИТ. Среда Windows рассматривается многими как некое персональное “изобретение” Microsoft, которое заслуживает упоминания лишь в ряду других “изобретений”, как один из многих способов решения технологических задач (а сама Microsoft вызывает “классовое чувство”). Это принципиально неверно. В персональной вычислительной технике ныне используются фактически лишь три графические ОС (MacOS — кстати, Windows “выросла” из технологии Apple, — OS/2 и Windows), и отличаются они друг от друга лишь техническими деталями, названиями элементов интерфейса (да и то не всегда) и “качеством” исполнения, но отнюдь не “идеологией”, не фундаментальными принципами, которые, собственно, и должны изучаться в школе. Если это признать, то вряд ли придется доказывать, что в качестве “тренажерного стенда” для обучения уместно избрать именно Windows (как наиболее распространенную), даже если “качество” ее приложений нас не всегда устраивает.

Разумеется, многое в этих средствах может меняться: отмирают одни операции, появляются другие, меняется визуальное представление элементов интерфейса (и даже их названия) и т.п. Однако человек, который освоил общие идеи и механизмы, сможет самостоятельно ориентироваться в своих проблемах. Открывая незнакомое приложение, он может поинтересоваться, выполняет ли эта программа нужные ему функции и как она их выполняет (а если не выполняет, то что может предложить взамен). С другой стороны, можно действовать и в обратном порядке: если пользователь не совсем четко представляет, что ему нужно, он познакомится сначала с возможностями приложения. Вероятно, это знакомство и приобретенный ранее кругозор подскажут ему дальнейшие шаги.

Итак, освоив суть современной ИТ и приступая к изучению частных операций приложения в какой-то предметной области, мы можем сосредоточиться исключительно на тех особенностях технологии, которые характерны именно для этой области. Попутно мы изучаем и базовые понятия информатики (например, алгебру логики, математические и информационные модели), которые используются в данной области (причем сразу на живых примерах!).

Например, при редактировании текстовых документов (MS Word) надо обратить внимание на структурирование и форматирование документа, на принципы верстки, работу с иллюстрациями, таблицами и т.п. При изучении электронных таблиц мы знакомимся с разнообразными способами представления и обработки произвольной информации в ячейках таблицы (а отнюдь не только чисел, как часто думают), с математическим моделированием, с условными и логическими функциями, с наложением графических объектов и т.д. При изучении информационных систем школьник на прак-



тике осваивает фундаментальную триаду “объект (сущность) — атрибут — связь”, нормализацию таблиц, первичный ключ, целостность данных, запросы и поиск (и снова алгебра логики!) и т.д., и т.п. Даже при изучении телекоммуникаций мы концентрируем внимание учащихся лишь на специфических, принципиальных вещах: что такое Internet, базовые и прикладные протоколы, адреса, гипертексты и страницы Web, серверы и клиенты и т.д. Что касается навигатора (например, Internet Explorer), то он в нашем понимании оказывается лишь простеньким приложением Windows, функции которого (в представлении пользователя) заканчиваются где-то на уровне команд **Открыть...** (документ), **Сохранить как...**, **Печать...**, **Копировать...** Об электронной почте и говорить нечего: если вы хотя бы в общих чертах знакомы с графическим интерфейсом пользователя, неужели вам надо долго объяснять, как “вынуть” письмо из ящика почтового клиента или как вложить в сообщение файл?

Одновременно мы получаем возможность обратить внимание учащихся на исключительно важные механизмы интеграции приложений, механизмы обмена данными (которые, собственно, и объединяют ИТ в единую систему): текстовые документы и графика; тексты, электронные таблицы и базы данных; мультимедийные документы, таблицы и базы данных (OLE) и прочее.

Разумеется, такой подход вовсе не претендует на всеобщность. Несколько упрощая, можно сказать, что мы предлагаем некую “азбуку” ИТ — для всех образованных людей. А будущим специалистам предстоит изучать множество других вещей: мультимедиа-технологии (как частный случай интеграции графических и текстовых инструментов, СУБД, аудиофайлов, видеофайлов); Internet-технологии (графика, HTML, web-дизайн, Java и многое другое); Visual Basic и визуальное программирование; компьютерные сети и архитектуру “клиент — сервер” и т.д.

Мне кажется, назрел вопрос о проведении “круглого стола” или какой-либо другой дискуссии не на тему информатики вообще, а по более узкому вопросу: “Что такое современные ИТ и как их преподавать”. И неплохо было бы наконец выслушать мнения профессионалов из “того мира”, мира ИТ, тех, кто ежедневно работает с “базовыми принципами”, о которых так любят рассуждать наши уважаемые теоретики. Правда, профессионалы — не педагоги, и скорее всего они начнут критиковать нас с другого конца — не за недостаток фундаментальности, а за использование, скажем, Word 7.0 вместо Word 97.

Существуют проблемы, на которые я как автор не могу найти однозначного ответа. Например, как быть с тем фактом, что в некоторых школах изучение информатики начинается с 1-го класса, а в некоторых — с 10-го? Как ввести унифицированную терминологию (постоянно делаю такие попытки)? Как найти компромисс между федеральным и региональным компонентами, который особенно труден для информатики?

Наконец, как отделить преподавание собственно компьютерной технологии от преподавания какой-либо другой дисциплины *на основе* компьютерной технологии? В качестве примера можно указать на идею включения в курс ИТ приложения “1С: Бухгалтерия” (или ее компонентов). В моем понимании продукты фирмы “1С” — это безупречно разработанные приложения Windows (со стандартным интерфейсом), и поэтому изучение указанного продукта (после освоения вами среды Windows) — это изучение именно бухгалтерского учета, а вовсе не одного из разделов ИТ. Вот вам и пример общеобразовательной функции предложенного подхода!

Ю.А. ШАФРИН

3

Логические задачи на вступительных экзаменах по информатике

В.И. РАКИТИН

Окончание. См. № 17, 18/99

2.5. Преобразование логических функций (выражений)

Формы представления логических функций

Две логические функции, зависящие от одних и тех же логических переменных, называют **равносильными** (эквивалентными), если для всех возможных наборов значений переменных они принимают одинаковые значения (другими словами, равносильные функции имеют одну и ту же таблицу истинности).

Выделяют несколько специальных форм представления логических функций:

- Представление в виде суммы произведений переменных и их отрицаний. Это так называемая дизъюнктивная нормальная форма (ДНФ). Например,

$$f(A,B,C) = A \cdot \bar{B} + \bar{B} \cdot C.$$

Если в дизъюнктивной нормальной форме (ДНФ) функции каждое слагаемое представляет собой произведение всех переменных функции или их отрицаний, а в сумме присутствуют все подобные произведения, то такое представление называют совершенной дизъюнктивной нормальной формой функции (СДНФ). Такая форма может быть получена на основании таблицы истинности. Например, в задаче 5 п. 2.4

$$f(A,B,C) = \bar{A} \cdot \bar{B} \cdot C + A \cdot \bar{B} \cdot \bar{C} + A \cdot \bar{B} \cdot C.$$

- Представление в виде произведения сумм переменных и их отрицаний. Это так называемая конъюнктивная нормальная форма (КНФ). Например,

$$f(A,B,C) = (A + \bar{B} + \bar{C}) \cdot (A + B).$$

Если в конъюнктивной нормальной форме (КНФ) функции каждый сомножитель представляет собой сумму всех переменных функции или их отрицаний, а в произведении присутствуют все подобные суммы, то такое представление называют совершенной конъюнктивной нормальной формой функции (СКНФ). Такая форма может быть получена на основании таблицы истинности. Например, в задаче 5 п. 2.4

$$f(A,B,C) = (A + B + C)(A + \bar{B} + C)(A + \bar{B} + \bar{C})(\bar{A} + \bar{B} + C)(\bar{A} + \bar{B} + \bar{C}).$$

Задача 7. Представить функцию трех переменных $f(A,B,C) = A$ в виде СДНФ и в виде СКНФ.

Решение

Для преобразования данной функции в совершенную дизъюнктивную нормальную форму будем представлять 1 то в виде $B + \bar{B}$, то в виде $C + \bar{C}$. Далее раскрываем скобки, пользуясь свойством дистрибутивности операции умножения по операции сложения:

$$\begin{aligned} A &= A \cdot (B + \bar{B}) = A \cdot B + A \cdot \bar{B} = A \cdot B \cdot (C + \bar{C}) + A \cdot \bar{B} \cdot (C + \bar{C}) = \\ &= A \cdot B \cdot C + A \cdot B \cdot \bar{C} + A \cdot \bar{B} \cdot C + A \cdot \bar{B} \cdot \bar{C}. \end{aligned}$$

Для преобразования данной функции к виду совершенной конъюнктивной нормальной формы будем представлять 0 то в виде $B \cdot \bar{B}$, то в виде $C \cdot \bar{C}$. Далее расставляем скобки на основании свойства дистрибутивности операции сложения по операции умножения:

$$\begin{aligned} A &= A + B \cdot \bar{B} = (A + B) \cdot (A + \bar{B}) = (A + B + C \cdot \bar{C}) \cdot (A + \bar{B} + C \cdot \bar{C}) = \\ &= (A + B + C) \cdot (A + \bar{B} + \bar{C}) \cdot (A + \bar{B} + C) \cdot (A + \bar{B} + \bar{C}). \end{aligned}$$

Дизъюнктивная нормальная форма называется **минимальной**, если она содержит наименьшее число конъюнкций и наименьшее общее число вхождений логических переменных по сравнению со всеми ей равносильными дизъюнктивными нормальными формами.

Упрощение логических функций

На примерах п. 2.4 (задача 5) было показано, что одна и та же функция может быть представлена формулами в дизъюнктивной и конъюнктивной нормальной формах. Равносильными преобразованиями с использованием законов алгебры логики часто можно получить другой вид формулы, содержащий меньшее число логических операций над переменными.

Рассмотрим несколько задач на преобразования логических выражений.

Задача 8. Доказать тождество

$$(A + B + C)(A + \bar{B} + C)(A + \bar{B} + \bar{C})(\bar{A} + \bar{B} + C)(\bar{A} + \bar{B} + \bar{C}) = A \cdot \bar{B} + \bar{B} \cdot C.$$

Решение

Преобразуем левую часть равенства.

- Упростим сначала произведение $(A + B + C)(A + \bar{B} + C)$. Используя коммутативность и ассоциативность по операции сложения (равенства 7 и 8 п. 1), получим $((A + C) + B)((A + C) + \bar{B})$. Далее на основании

закона дистрибутивности сложения по операции умножения (равенства 9 п. 1) имеем $(A + C) + (A + C)(\bar{B} + B) + B \cdot \bar{B}$. Учтем наконец законы 6 п. 1: $\bar{B} + B = 1$, $B \cdot \bar{B} = 0$ — и равенство 3 п. 1: $(A + C) + (A + C) = A + C$ — и приведем всю последовательность преобразований.

$$\begin{aligned} (A + B + C)(A + \bar{B} + C) &= ((A + C) + B)((A + C) + \bar{B}) = \\ &= (A + C) + (A + C)(\bar{B} + B) + B \cdot \bar{B} = A + C. \end{aligned}$$

- Произведение $(\bar{A} + \bar{B} + C)(\bar{A} + \bar{B} + \bar{C})$ преобразуется аналогично предыдущему. Здесь имеем $(\bar{A} + \bar{B} + C)(\bar{A} + \bar{B} + \bar{C}) = ((\bar{A} + \bar{B}) + C)((\bar{A} + \bar{B}) + \bar{C}) = (\bar{A} + \bar{B}) + (\bar{A} + \bar{B})(\bar{C} + C) + C \cdot \bar{C} = \bar{A} + \bar{B}$.

- При упрощении произведения $(A + C)(A + \bar{B} + \bar{C})$ используется, в частности, закон поглощения: $A + A \cdot \bar{B} + A \cdot \bar{C} + C A = A$, и тогда $(A + C)(A + \bar{B} + \bar{C}) = A + A \cdot \bar{B} + A \cdot \bar{C} + C A + C \cdot \bar{B} + C \cdot \bar{C} = A + \bar{B} \cdot C$.

- На последнем шаге преобразования произведения $(A + \bar{B} \cdot C)(\bar{A} + \bar{B})$ закон поглощения позволяет записать, что $\bar{A} \cdot \bar{B} \cdot C + \bar{B} \cdot C = \bar{B} \cdot C$. Окончательно:

$$\begin{aligned} (A + \bar{B} \cdot C)(\bar{A} + \bar{B}) &= A \cdot \bar{A} + A \cdot \bar{B} + \bar{B} \cdot C \cdot \bar{A} + \bar{B} \cdot C \cdot \bar{B} = \\ &= \bar{A} \cdot \bar{B} \cdot C + A \cdot \bar{B} + \bar{B} \cdot C = A \cdot \bar{B} + \bar{B} \cdot C. \end{aligned}$$

Левая часть равна правой — тождество доказано.

Задача 9. Доказать тождество

$$A \cdot \bar{B} \cdot \bar{C} + \bar{A} \cdot B \cdot \bar{C} + \bar{A} \cdot \bar{B} \cdot C + A \cdot B \cdot \bar{C} + \bar{A} \cdot B \cdot C + A \cdot \bar{B} \cdot C + A \cdot B \cdot C = A + B + C.$$

Решение

Преобразуем левую часть равенства.

$$A \cdot \bar{B} \cdot \bar{C} + A \cdot B \cdot \bar{C} = A \cdot \bar{C} (\bar{B} + B) = A \cdot \bar{C},$$

$$\bar{A} \cdot \bar{B} \cdot C + \bar{A} \cdot B \cdot C = \bar{A} \cdot C (\bar{B} + B) = \bar{A} \cdot C,$$

$$A \cdot \bar{B} \cdot C + A \cdot B \cdot C = A \cdot C (\bar{B} + B) = A \cdot C,$$

$$A \cdot \bar{C} + \bar{A} \cdot C + A \cdot C + \bar{A} \cdot B \cdot \bar{C} = A + \bar{A} \cdot C + \bar{A} \cdot B \cdot \bar{C} = A + \bar{A} \cdot (C + B \cdot \bar{C}) =$$

$$= A + \bar{A} \cdot (C + B) = A + B + C,$$

здесь при упрощении используется закон дистрибутивности умножения по операции сложения (равенства 9 п. 1) в виде $A + \bar{A} \cdot (C + B) = (A + \bar{A}) \cdot (A + B + C)$.

Левая часть равна правой — тождество доказано.

Построение минимальной дизъюнктивной нормальной формы логической функции

Минимальную ДНФ некоторой функции можно найти, перебрав конечное число равносильных ей ДНФ и выбрав среди них ту, которая содержит минимальное число переменных. При большом числе переменных это трудновыполнимая задача.

Рассмотрим на примере один из способов нахождения минимальной дизъюнктивной нормальной формы, называемый методом минимизирующих карт.

Задача 10. Дана совершенная дизъюнктивная нормальная форма (СДНФ) функции

$$f(A,B,C) = A \cdot \bar{B} \cdot C + A \cdot \bar{B} \cdot \bar{C} + \bar{A} \cdot \bar{B} \cdot C,$$

полученной при решении задачи 5 п. 2.4. Найти минимальную дизъюнктивную нормальную форму этой функции.

Решение

Строим таблицу:

•	$A \cdot B \cdot C$	A	B	C	$A \cdot B$	$A \cdot C$	$B \cdot C$
•	$A \cdot B \cdot \bar{C}$	A	B	\bar{C}	$A \cdot B$	$A \cdot \bar{C}$	$B \cdot \bar{C}$
	$A \cdot \bar{B} \cdot C$	A	\bar{B}	C	$A \cdot \bar{B}$	$A \cdot C$	$\bar{B} \cdot C$
	$A \cdot \bar{B} \cdot \bar{C}$	A	\bar{B}	\bar{C}	$A \cdot \bar{B}$	$A \cdot \bar{C}$	$\bar{B} \cdot \bar{C}$
•	$\bar{A} \cdot B \cdot C$	\bar{A}	B	C	$\bar{A} \cdot B$	$\bar{A} \cdot C$	$B \cdot C$
•	$\bar{A} \cdot B \cdot \bar{C}$	\bar{A}	B	\bar{C}	$\bar{A} \cdot B$	$\bar{A} \cdot \bar{C}$	$B \cdot \bar{C}$
	$\bar{A} \cdot \bar{B} \cdot C$	\bar{A}	\bar{B}	C	$\bar{A} \cdot \bar{B}$	$\bar{A} \cdot C$	$\bar{B} \cdot C$
•	$\bar{A} \cdot \bar{B} \cdot \bar{C}$	\bar{A}	\bar{B}	\bar{C}	$\bar{A} \cdot \bar{B}$	$\bar{A} \cdot \bar{C}$	$\bar{B} \cdot \bar{C}$

Первый столбец содержит метки “•”, второй состоит из всех возможных произведений переменных и их отрицаний, далее следуют столбцы переменных и парных произведений переменных, которые вместе с отрицаниями входят в соответствующие произведения второго столбца.

4

Последовательность построения минимальной ДНФ:

1. Отметим метками в первом столбце таблицы строки, в которых произведения не принадлежат СДНФ-функции, и удалим все элементы помеченных строк.
2. Одновременно в каждом столбце удалим элементы, совпадающие со всеми удаляемыми из помеченных строк.
3. В каждой строке оставляем элементы, имеющие наименьшее число сомножителей.

•							
•							
					$A \cdot \bar{B}$		$\bar{B} \cdot C$
					$A \cdot \bar{B}$		
•							
•							
							$\bar{B} \cdot C$
•							

4. В каждой строке выберем по одному оставшемуся элементу и составим из них ДНФ: $A \cdot \bar{B} + A \cdot \bar{B} + \bar{B} \cdot C, \bar{B} \cdot C + A \cdot \bar{B} + \bar{B} \cdot C$ — формы совпали и имеют вид $A \cdot \bar{B} + \bar{B} \cdot C$.
5. Из всех ДНФ (в данном примере имеем только одну), полученных в предыдущем пункте, выберем минимальную:

$$f(A,B,C) = A \cdot \bar{B} + \bar{B} \cdot C.$$

Последовательность действий в описанном методе, который называют методом минимизирующих карт, будет восприниматься более естественно, если сравнить два метода построения функции в задаче 5 п. 2.4 в различных формах:

в виде СДНФ

$$f(A,B,C) = A \cdot \bar{B} \cdot C + A \cdot \bar{B} \cdot \bar{C} + \bar{A} \cdot \bar{B} \cdot C;$$

в виде СКНФ

$$f(A,B,C) = (A + B + C)(A + \bar{B} + C)(A + \bar{B} + \bar{C})(\bar{A} + \bar{B} + C)(\bar{A} + \bar{B} + \bar{C}).$$

Действительно, функция $f(A,B,C)$ равна 0 в помеченных строках, и, следовательно, одна из сумм в СКНФ обращается в ноль вместе с каждым слагаемым и любыми произведениями, составленными из этих слагаемых. В частности, например, если $A + B + C = 0$, то $A = B = C = 0$; $A \cdot B = A \cdot C = B \cdot C = 0$; $A \cdot B \cdot C = 0$ и все эти комбинации исключаются из столбцов таблицы. Так мотивируется упрощение вида функции исключением всех меченых строк из таблицы и всех элементов, совпадающих с элементами меченой строки.

Заметим, что в задаче 8 этого раздела при доказательстве тождества в левой части была логическая функция в совершенной конъюнктивной форме

$$f(A,B,C) = (A + B + C)(A + \bar{B} + C)(A + \bar{B} + \bar{C})(\bar{A} + \bar{B} + C)(\bar{A} + \bar{B} + \bar{C}).$$

Правая часть тождества — это минимальная дизъюнктивная нормальная форма этой же функции.

Минимальную дизъюнктивную нормальную форму функции удобно использовать, но в общем случае трудно получить из произвольной ДНФ только с помощью алгебраических преобразований. Часто можно ограничиться так называемой **сокращенной** дизъюнктивной нормальной формой. Сокращенная ДНФ может быть получена, например, по следующему алгоритму. Имея некоторую произвольную конъюнктивную нормальную форму исходной функции, раскрываем скобки, используя свойства дистрибутивности, упрощаем, используя равносильные выражения типа $A \cdot A = A$ или законы поглощения $A + A \cdot B = A$, $A \cdot (A + B) = A$ и удаляя ложные константы типа $A \cdot \bar{A} = 0$. В результате получаем сокращенную ДНФ. В задаче 8 сокращенная ДНФ $A \cdot \bar{B} + \bar{B} \cdot C$ оказалась минимальной.

Задача 11. Заданы логические функции: F_1 , равная 1 на наборах 3, 5, 6, 7 и равная 0 на остальных наборах, и

$$F_2 = (\bar{X}_2 \cdot X_1 + \bar{X}_3 \cdot X_1 + X_2 \cdot X_3) \cdot (\bar{X}_1 \cdot \bar{X}_3 + \bar{X}_3 \cdot X_2).$$

Проверить, являются ли эти функции тождественными.

Решение

1. Простейший способ проверки тождественности двух различных представлений функций — построение таблиц истинности для каждой из функций без их алгебраических преобразований.

	X_1, X_2, X_3	$F_1(X_1, X_2, X_3)$
0	0, 0, 0	0
1	0, 0, 1	0
2	0, 1, 0	0
3	0, 1, 1	1
4	1, 0, 0	0
5	1, 0, 1	1
6	1, 1, 0	1
7	1, 1, 1	1

X_1, X_2, X_3	$\bar{X}_2 \cdot X_1 + \bar{X}_3 \cdot X_1 + X_2 \cdot X_3$	$\bar{X}_1 \cdot \bar{X}_3 + \bar{X}_3 \cdot X_2$	$F_2(X_1, X_2, X_3)$
0, 0, 0	0	0	1
0, 0, 1	0	0	1
0, 1, 0	0	0	1
0, 1, 1	1	1	1
1, 0, 0	0	1	0
1, 0, 1	1	1	1
1, 1, 0	1	1	1
1, 1, 1	1	1	1

Очевидно, что функции $F_1(X_1, X_2, X_3)$ и $F_2(X_1, X_2, X_3)$ равносильны.

2. Другой способ — построить совершенную дизъюнктивную нормальную форму для функции F_1 по таблице истинности:

$$F_1(X_1, X_2, X_3) = \bar{X}_1 \cdot X_2 \cdot X_3 + X_1 \cdot \bar{X}_2 \cdot X_3 + X_1 \cdot X_2 \cdot \bar{X}_3 + X_1 \cdot X_2 \cdot X_3,$$

преобразовать функцию F_2 к виду СДНФ (совершенная форма — единственная) с помощью тождественных преобразований и сравнить результаты. Заметим, что вид функции F_2 не является ни ДНФ, ни КНФ. Последовательно преобразуем F_2 :

- $F_2 = (\bar{X}_2 \cdot X_1 + \bar{X}_3 \cdot X_1 + X_2 \cdot X_3) \cdot (\bar{X}_1 \cdot \bar{X}_3 + \bar{X}_3 \cdot X_2) =$
 $= (\bar{X}_2 \cdot X_1 + \bar{X}_3 \cdot X_1 + X_2 \cdot X_3) \cdot (\bar{X}_1 + X_3 + \bar{X}_3 \cdot X_2),$
- $F_2 = (\bar{X}_2 \cdot X_1 + \bar{X}_3 \cdot X_1 + X_2 \cdot X_3) \cdot (\bar{X}_1 \cdot \bar{X}_3 + \bar{X}_3 \cdot X_2) =$
 $= (\bar{X}_2 \cdot X_1 + \bar{X}_3 \cdot X_1 + X_2 \cdot X_3) \cdot (\bar{X}_1 + X_2 + X_3),$
- $F_2 = X_1 \cdot \bar{X}_2 \cdot X_3 + X_1 \cdot X_2 \cdot \bar{X}_3 + \bar{X}_1 \cdot X_2 \cdot X_3 + X_2 \cdot X_3,$
- $F_2 = X_1 \cdot \bar{X}_2 \cdot X_3 + X_1 \cdot X_2 \cdot \bar{X}_3 + \bar{X}_1 \cdot X_2 \cdot X_3 + X_2 \cdot X_3 \cdot (X_1 + \bar{X}_1),$
- $F_2 = X_1 \cdot \bar{X}_2 \cdot X_3 + X_1 \cdot X_2 \cdot \bar{X}_3 + \bar{X}_1 \cdot X_2 \cdot X_3 + X_1 + X_1 \cdot X_2 \cdot X_3,$
- $F_2(X_1, X_2, X_3) = \bar{X}_1 \cdot X_2 \cdot X_3 + X_1 \cdot \bar{X}_2 \cdot X_3 + X_1 \cdot X_2 \cdot \bar{X}_3 + X_1 \cdot X_2 \cdot X_3.$

Виды функций $F_1(X_1, X_2, X_3)$ и $F_2(X_1, X_2, X_3)$ совпадают — функции равносильны.

2.6. Текстовые логические задачи

Задача 12. Разбирается дело Батончика, Ленчика и Пончика. Кто-то из них нашел и утаил клад. На следствии каждый из них сделал два заявления:

- а) Батончик: “Я не делал этого. Пончик сделал это”;
- б) Ленчик: “Пончик невиновен. Батончик сделал это”;
- в) Пончик: “Я не делал этого. Ленчик не делал этого”.

Суд установил, что один из них дважды солгал, другой дважды сказал правду, третий один раз солгал, один раз сказал правду. Кто из них утаил клад?

Решение

Обозначим буквами следующие утверждения: B — Батончик утаил клад, L — Ленчик утаил клад, P — Пончик утаил клад. Тогда каждое из заявлений, состоящее из двух утверждений, можно представить так: заявление Батончика — \bar{B}, P ; заявление Ленчика — \bar{P}, B ; заявление Пончика — \bar{P}, \bar{L} .

Здесь правильный ответ можно получить, анализируя всего лишь три возможные версии на их соответствие каждому утверждению. Анализ версий оформлен в виде таблицы характера совпадений версий с заявлениями.

Версии	Высказывания из двух заявлений					
	Батончика		Ленчика		Пончика	
1. Батончик утаил клад — B	0	0	1	1	1	1
2. Ленчик утаил клад — L	1	0	1	1	1	0
3. Пончик утаил клад — P	1	1	0	0	0	1

Версия 3 соответствует условию задачи. У Батончика оба заявления верны: $P \cdot \bar{B} = 1, P \cdot P = 1$. Ленчик дважды солгал: $P \cdot \bar{P} = 0, P \cdot B = 0$. Пончик один раз солгал, другой — нет: $P \cdot \bar{P} = 0, P \cdot \bar{L} = 1$.

Окончание на с. 6

ЭКЗАМЕНЫ В ВУЗЫ

1999 № 19 ИНФОРМАТИКА

5

Логические задачи на вступительных экзаменах по информатике

Окончание. См. с. 4–5

Задача 13. Петя, Вася и Маша остались дома одни. Кто-то из них ел варенье. На вопрос мамы, кто это сделал, они сказали:

- Петя: “Я не ел. Маша тоже не ела”;
- Вася: “Маша действительно не ела. Это сделал Петя”;
- Маша: “Вася врет. Это он съел”.

Выясните, кто ел варенье, если известно, что двое оба раза сказали правду, а третий один раз соврал, а один раз сказал правду.

Решение

Обозначим высказывания Пети через \bar{P}, \bar{M} ; высказывания Васи — \bar{M}, P ; высказывания Маши — $\bar{M} \cdot \bar{P} = M + \bar{P}, V$.

Здесь, как и в предыдущей задаче, удобно проанализировать три возможные версии и проверить их соответствие каждому утверждению. Совпадению версии с заявлением соответствует единица в таблице, в противном случае — ноль. Анализ версий представлен в таблице. Версия 2 “Вася съел варенье” удовлетворяет условию задачи и является верной.

Версии	Высказывания из двух заявлений					
	Пети — \bar{P}, \bar{M}		Васи — \bar{M}, P		Маши — $M + \bar{P}, V$	
1. Петя съел — P	0	1	1	1	0	0
2. Вася съел — V	1	1	1	0	1	1
3. Маша съела — M	1	0	0	0	1	0

В случае 2 (Вася съел — V) оказывается, что у Пети оба заявления совпали с версией: $V \cdot \bar{P} = 1, V \cdot \bar{M} = 1$; у Васи одно заявление совпало, другое — нет: $V \cdot \bar{M} = 1, V \cdot P = 0$; у Маши оба заявления совпали с версией: $V \cdot (M + \bar{P}) = 1, V \cdot V = 1$.

Задача 14. На соревнованиях Виктор, Роман, Юра и Сергей заняли первые четыре места. Когда их спрашивают о распределении мест, они дают три ответа:

- Ответ 1. Юра занял второе место, а Виктор — четвертое ($Ю_2, В_4$).
 Ответ 2. Сергей занял второе место, а Виктор — третье ($С_2, В_3$).
 Ответ 3. Сергей занял первое место, а Роман — второе ($С_1, Р_2$).

Как распределились места, если

- в каждом ответе только одно высказывание верно;
- в каждом ответе хотя бы одно высказывание верно?

Решение

В данной задаче ответом является один из вариантов перестановок четырех человек по четырем местам, т.е. один из случаев вида $V_i P_j Ю_k С_n$ ($i, j, k, n = 1, 2, 3, 4$ и $i \neq j \neq k \neq n$). Общее число таких перестановок равно 24. Сверять каждую из возможных 24 комбинаций с условиями задачи и избежать ошибок достаточно трудно. Для решения задач будем строить логические функции, соответствующие условиям задачи.

а) Построение логической функции для условия “в каждом ответе только одно высказывание верно”. Логические выражения для каждого ответа даны в таблице:

В первом ответе только одно высказывание верно	Во втором ответе только одно высказывание верно	В третьем ответе только одно высказывание верно
$\bar{Ю}_2 \bar{В}_4 + Ю_2 \bar{В}_4$	$\bar{С}_2 \bar{В}_3 + С_2 \bar{В}_3$	$\bar{С}_1 \bar{Р}_2 + С_1 \bar{Р}_2$

Функция будет иметь вид

$$f_a(Ю_i, В_j, С_k, Р_n) = (\bar{Ю}_2 \bar{В}_4 + Ю_2 \bar{В}_4) (\bar{С}_2 \bar{В}_3 + С_2 \bar{В}_3) (\bar{С}_1 \bar{Р}_2 + С_1 \bar{Р}_2).$$

Упростим логическое выражение, раскрывая скобки и учитывая, что два разных человека не могут занимать одно и то же место, что один человек не может находиться на разных местах и что каждый занимает какое-либо место. Тогда, например, $Ю_2 С_2 = 0, С_2 С_1 = 0$ и $\bar{Ю}_2 Ю_2 = 0$.

$$\text{Имеем: } (\bar{Ю}_2 \bar{В}_4 + Ю_2 \bar{В}_4) (\bar{С}_2 \bar{В}_3 + С_2 \bar{В}_3) (\bar{С}_1 \bar{Р}_2 + С_1 \bar{Р}_2) = (\bar{Ю}_2 \bar{В}_4 \bar{С}_2 + Ю_2 \bar{С}_2 \bar{В}_3) (\bar{С}_1 \bar{Р}_2 + С_1 \bar{Р}_2) = Ю_2 С_1 \bar{В}_3 \bar{Р}_3.$$

Тогда $f_a(Ю_i, В_j, С_k, Р_n) = С_1 Ю_2 \bar{В}_3 \bar{Р}_3$. Эта функция равна 1 только в случае, когда Сергей занимает первое место, Юра — второе, Виктор — третье и Роман — четвертое.

б) Построение логической функции для условия “в каждом ответе хотя бы одно высказывание верно”.

В первом ответе хотя бы одно высказывание верно	Во втором ответе хотя бы одно высказывание верно	В третьем ответе хотя бы одно высказывание верно
$Ю_2 + \bar{В}_4$	$С_2 + \bar{В}_3$	$С_1 + \bar{Р}_2$

Функция высказываний $f_b(Ю_i, В_j, С_k, Р_n) = (Ю_2 + \bar{В}_4) (С_2 + \bar{В}_3) (С_1 + \bar{Р}_2)$, где ($i, j, k, n = 1, 2, 3, 4$ и $i \neq j \neq k \neq n$). Преобразуем логическую функцию, используя свойства операций: $(Ю_2 С_2 + Ю_2 \bar{В}_3 + \bar{В}_4 С_2 + \bar{В}_4 \bar{В}_3) (С_1 + \bar{Р}_2) = (Ю_2 \bar{В}_3 + \bar{В}_4 С_2) (С_1 + \bar{Р}_2) = Ю_2 \bar{В}_3 С_1$. Так же, как и в случае (а), в преобразованиях учтено, что одно и то же лицо не может занимать разные места, а также одно место не может быть занято разными людьми. Это дает, например, $С_2 С_1 = 0, Ю_2 \bar{Р}_2 = 0$ и т.д. Здесь $f_b(Ю_i, В_j, С_k, Р_n) = С_1 Ю_2 \bar{В}_3$.

Ответ. Распределение мест таково: Сергей занимает первое место, Юра — второе, Виктор — третье и Роман — четвертое.

Задача 15. По обвинению в ограблении перед судом предстали три человека — А, В и С. Установлено следующее:

1) если или А невиновен, или В виновен, то С виновен ($(\bar{A} + B \Rightarrow C) = \bar{A} + B + C = \bar{A}B + C$);

2) если А невиновен, то С невиновен ($(\bar{A} \Rightarrow \bar{C}) = A + \bar{C}$).
 Установить, виновен ли А.

Замечание: здесь использовано соотношение $(A \Rightarrow B) = \bar{A} + B$.

Решение

Логическая функция, соответствующая условию $f(A, B, C) = (\bar{A} + B \Rightarrow C) (\bar{A} \Rightarrow \bar{C})$. Преобразуем логическую функцию, используя свойства операций: $f(A, B, C) = (\bar{A} \cdot \bar{B} + C) (A + \bar{C})$. $A \cdot \bar{B} \cdot \bar{C} + AC = 1, A \cdot (\bar{B} \cdot \bar{C} + C) = 1$. Произведение равно 1, когда $A = 1$, и $C + \bar{B} \cdot \bar{C} = 1$ ($C + \bar{B} = 1$). Отсюда следует, что А виновен, или С виновен, или В невиновен: С виновен, В виновен; С невиновен, В невиновен; С невиновен, В виновен.

Ответ: А виновен.

Задача 16. Определить, кто участвовал в ограблении, если известно, что

1) если А участвовал, то и В участвовал ($(A \Rightarrow B) = \bar{A} + B$);

2) если В участвовал, то или С участвовал, или А не участвовал ($(B \Rightarrow C + \bar{A}) = \bar{B} + C + \bar{A}$);

3) если D не участвовал, то А участвовал, а С не участвовал ($(\bar{D} \Rightarrow A \cdot \bar{C}) = D + A \bar{C}$);

4) если D участвовал, то А участвовал ($(D \Rightarrow A) = \bar{D} + A$).

Решение

Логическая функция $f(A, B, C, D) = (A \Rightarrow B) \cdot (B \Rightarrow C + \bar{A}) \cdot (\bar{D} \Rightarrow A \cdot \bar{C}) \cdot (D \Rightarrow A)$. Преобразуем логическую функцию, используя свойства операций:

$$\bullet f(A, B, C, D) = (\bar{A} + B) (\bar{B} + C + \bar{A}) \cdot (D + A \bar{C}) \cdot (\bar{D} + A);$$

$$\bullet f(A, B, C, D) = (\bar{A} + \bar{A}C + B \cdot C) (A \cdot D + A \cdot \bar{C} \cdot \bar{D} + A \cdot \bar{C});$$

$$\bullet f(A, B, C, D) = A \cdot B \cdot C \cdot D; \quad A \cdot B \cdot C \cdot D = 1 \Rightarrow (A = 1, B = 1, C = 1, D = 1).$$

Ответ: в ограблении участвовали А, В, С, D.

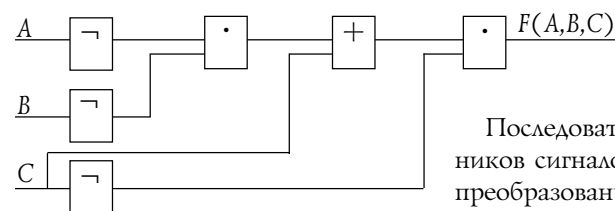
2.7. Построение логической функции по заданной логической схеме

Будем понимать под логической схемой графическое представление логической функции. Схема строится из логических элементов и отрезков прямых. Логический элемент — это прямоугольник, в котором ставится символ логической операции. Указанные операции производятся над логическими переменными на входе в прямоугольник (слева); результат операции передается на выход (справа). Логическую схему можно интерпретировать как электрическую цепь с преобразователями сигналов в виде логических элементов.

Логические элементы представлены в следующей таблице:

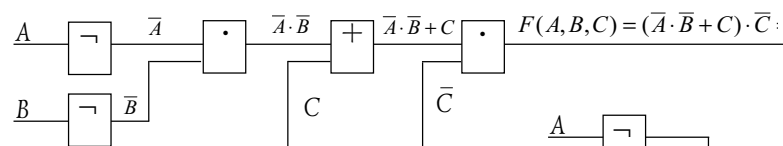
Логический элемент “ИЛИ”	Логический элемент “И”	Логический элемент “НЕ”

Задача 17. Дана логическая схема. Составить по ней логическую функцию.

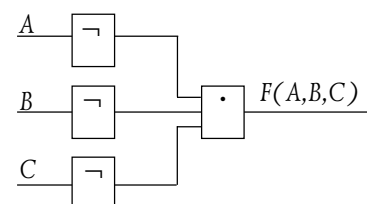


Решение

Последовательно, начиная от источников сигналов А, В, С, осуществляем преобразования по схеме.



Очевидно упрощение логической функции $F(A, B, C) = (\bar{A} \cdot \bar{B} + C) \cdot \bar{C} = \bar{A} \cdot \bar{B} \cdot \bar{C}$ и соответствующей логической схемы.



1. Замечания по терминологии

История информатики в нашей стране (сначала СССР, а затем России) насыщена драматическими коллизиями и резкими изменениями приоритетов. Это ощущается даже в терминологии. Термин информатика для обозначения совокупности научных направлений, тесно связанных с появлением компьютеров и их стремительным вхождением в ноосферу, определяемую жизнедеятельностью людей, у нас относительно новый. Он получил “права гражданства” в начале 80-х годов, а до этого согласно определению, данному в Большой советской энциклопедии, информатика рассматривалась как “дисциплина, изучающая структуру и общие свойства научной информации, а также закономер-

ности ее создания, преобразования, передачи и использования в различных сферах человеческой деятельности” [1, с. 1031].

мин кибернетика, затем на роль общего названия той же области исследований стала претендовать прикладная математика. Следы этой разногласности хорошо видны в наименовании высших учебных заведений и научных институтов. Факультет в МГУ, готовящий специалистов в области информатики, носит название “Вычислительная математика и кибернетика”, а институты, ведущие исследования в данной области, могут называться и Институт кибернетики Национальной АН Украины, и Институт прикладной информатики РАН, и Институт прикладной математики РАН.

Поэтому, говоря об истории информатики в бывшем СССР и теперешней России, по сути, надо излагать историю отечественной кибернетики и частично прикладной математики и вычислительной техники. Именно так построена эта статья.

- числовые и символьные вычисления (компьютерно-ориентированные методы вычислений, модели переработки информации в различных прикладных областях, работа с естественно-языковыми текстами и т.п.);

- системы человеко-машинного взаимодействия (модели дискурса, распределение работ в смешанных системах, организация коллективных процедур, деятельность в телекоммуникационных системах и т.п.);

- нейроматематика и нейросистемы (теория формальных нейронных сетей, использование нейронных сетей для обучения, нейрокомпьютеры и т.п.);

- использование компьютеров в замкнутых системах (модели реального времени, интеллектуальное управление, системы мониторинга и т.п.).

Эти области информатики возникли не одновременно. История информатики связана с постепенным расши-

рением области ее интересов. Возможность расширения диктовалась развитием компьютеров и накоплением моделей и методов их применения при решении задач различного типа. Следуя историческим реалиям, мы в данной книге постараемся отразить эту особенность истории информатики.

тротехники АН УССР под руководством С.А. Лебедева уже началась работа по созданию вычислительной машины, использующей двоичную систему счисления, но начавшаяся война прервала эти исследования [3]. После нее наступило время их продолжить. В 1951 году в Киеве заработала первая в континентальной Европе вычислительная машина — МЭСМ, созданная коллективом, возглавляемым С.А. Лебедевым.

Работы, имевшие для страны большое значение, как это было принято, поручались сразу нескольким организациям. Поэтому МЭСМ и вскоре последовавшая за ней БЭСМ не оказались одиночками. В 1952 году стали действовать машины М-1 и М-2, созданные в коллективе И.С. Брука, в 1953 году появился первый экземпляр ЭВМ “Стрела”, а с 1954 года началось семейство машин

Становление информатики в России

Д.А. ПОСПЕЛОВ

2. Структура информатики

На протяжении полувековой истории информатики в ней неоднократно возникали и исчезали те или иные направления. В настоящее время ее структура, по-видимому, определилась. В нее входят следующие основные области исследования:

- теория алгоритмов (формальные модели алгоритмов, проблемы вычислимости, сложность вычислений и т.п.);

- логические модели (дедуктивные системы, сложность вывода, нетрадиционные исчисления: индуктивный и абдуктивный вывод, вывод по аналогии, правдоподобный вывод, немонотонные рассуждения и т.п.);

- базы данных (структуры данных, поиск ответов на запросы, логический вывод в базах данных, активные базы и т.п.);

- искусственный интеллект (представление знаний, вывод на знаниях, обучение, экспертные системы и т.п.);

- бионика (математические модели в биологии, модели поведения, генетические системы и алгоритмы и т.п.);

- распознавание образов и обработка зрительных сцен (статистические методы распознавания, использование призначных пространств, теория распознающих алгоритмов, трехмерные сцены и т.п.);

- теория роботов (автономные роботы, представление знаний о мире, децентрализованное управление, планирование целесообразного поведения и т.п.);

- инженерия математического обеспечения (языки программирования, технологии создания программных систем, инструментальные системы и т.п.);

- теория компьютеров и вычислительных сетей (архитектурные решения, многоагентные системы, новые принципы переработки информации и т.п.);

- компьютерная лингвистика (модели языка, анализ и синтез текстов, машинный перевод и т.п.);

ности ее создания, преобразования, передачи и использования в различных сферах человеческой деятельности” [1, с. 1031].

Подобное определение связывало информатику с библиотековедением, библиографией, методами поиска информации в массивах документов. Когда в 1952 году был создан Институт научной информации АН СССР, позже преобразованный в ВИНТИ — Всесоюзный институт научной и технической информации, он должен был стать главным академическим учреждением в области информатики.

То, что стало называться информатикой в начале 80-х в нашей стране, было совершенно иным. Ближе всего содержание этого понятия подходит к тому, что в США и большинстве других стран называется *computer science*, т.е. компьютерные науки.

В [2] говорится, что “компьютерные науки” концентрируют свое внимание на различных аспектах, связанных с протеканием и использованием информационных процессов, с теми структурами, в которых представляется информация, и теми процедурами, которые используются при ее переработке. Последнее связывает область “компьютерных наук” с теорией машин для переработки информации — компьютеров — и методами их использования в системах переработки информации.

Для термина информатика в [2] нет отдельной статьи, а есть лишь ссылки на термины: компьютерные науки, компьютерное обучение и информационные науки. Содержание понятия информационные науки в [2] полностью совпадает с толкованием термина информатика в БСЭ [1].

С начала 80-х содержание того, что скрывается за термином информатика, ближе всего к тому, что понимают французы, когда говорят о науке, носящей название *informatique*.

До этого совокупность научных направлений, называемых теперь информатикой, именовалась по-разному. Сначала объединяющим названием был тер-

мин “Урал”, главным конструктором которого был Б.И. Рамеев [3, 4].

Все фундаментальные исследования и инженерные разработки, которые могли использоваться в военной сфере, в СССР были скрыты от общественности завесой секретности. Поэтому первая научная монография по теории ЭВМ и программированию [5] имела гриф и выдавалась лишь по предъявлению документа о допуске к государственным секретам.

3. Борьба за признание

Во второй половине 30-х годов в нескольких странах появились первые проекты электромеханических и электронных устройств, нацеленных на выполнение массовых вычислений. Первый проект, завершившийся созданием прообраза будущих вычислительных машин, был выполнен в США. К декабрю 1939 года Дж. Атанасов и К.Берри создали макет процессора, а в мае 1942 года первая в мире вычислительная машина начала действовать. Эти работы велись в условиях секретности, что впоследствии породило судебное разбирательство по вопросу о приоритете с разработчиками машины ЭНИАК, созданной в США в период с 1943-го по 1946 год.

Однако именно с ЭНИАКом связано начало той вычислительной техники, которая породила сначала кибернетику, а затем и информатику. В этой машине впервые была реализована структура, предложенная Дж. фон Нейманом. Программа вычислений стала объектом, доступным для преобразования с помощью вычислительной машины. Так возникло программирование.

В нашу страну сведения о создании новых видов переработчиков информации поступили довольно быстро. Исходя из интересов страны (прежде всего из необходимости поддерживать высокий уровень военных разработок), в СССР начались работы по созданию отечественных вычислительных машин. В конце 30-х годов в Институте элек-

трики, главным конструктором которого был Б.И. Рамеев [3, 4].

Все фундаментальные исследования и инженерные разработки, которые могли использоваться в военной сфере, в СССР были скрыты от общественности завесой секретности. Поэтому первая научная монография по теории ЭВМ и программированию [5] имела гриф и выдавалась лишь по предъявлению документа о допуске к государственным секретам.

Изданная в 1948 году книга американского математика Норберта Винера “Кибернетика, или Управление и связь в животном и машине” попала на полки с секретными изданиями по другой причине. Ее автор высказал идеи, не согласующиеся с официальными доктринами, пропагандируемыми в советском обществе.

Для Винера было абсолютно ясно, что многие концептуальные схемы, определяющие поведение живых организмов при решении конкретных задач, практически идентичны схемам, характеризующим процессы управления в сложных технических системах. И более того, он был убежден, что социальные модели управления и модели управления в экономике могут быть проанализированы на основе тех же общих положений, которые разработаны в области управления системами, созданными людьми.

Эти крамольные идеи не могли стать достоянием советских граждан, которым настойчиво внушался тезис марксистской философии о несводимости “высших форм” существования материи к “низшим формам”. Поэтому место книги Винера было однозначно определено — спецхран.

В этом состоит главная причина того, что у истоков развития кибернетики (информатики) в СССР стояли сотрудники различных закрытых ведомств и предприятий, в большинстве своем носившие военную форму. Забегая вперед, отметим, что все первые книги в области кибернетики, вычислительных машин и программирования, выпу-

ВЕК

ИНФОРМАТИКИ

Становление информатики в России

Продолжение. Начало на с. 7

ценные уже во второй половине 50-х годов без грифа секретности, были написаны военными. Этот нетривиальный для истории науки факт имел для отечественной информатики немаловажное значение. Если бы не активная, наступательная позиция военных, поддержанная членами АН СССР, то идеологические концепции, охраняемые представителями консервативной философской элиты, задержали бы на много десятилетий развитие информатики, как это случилось с генетикой и другими неудобными придворной философии науками. Время для очередного разгрома — начало 50-х годов — было весьма подходящим.

Первой ласточкой стала статья, помещенная на страницах идеологического офидиоза “Вопросы философии” в марте 1950 года [6]. В ней критике были подвергнуты некоторые теоретические положения математической логики, противоречащие, по мнению авторов статьи, догмам материализма. Статья была откликом на публикацию переводов книг Д.Гильберта и В.Аккермана “Основы теоретической логики” (М.: Издательство иностранной литературы, 1947) и А.Тарского “Введение в логику и методологию дедуктивных наук” (М.: Издательство иностранной литературы, 1948). Редактором перевода и автором предисловия к первой из книг была С.А. Яновская, в издании и комментировании второй книги, кроме нее, участвовал еще Г.М. Адельсон-Вельский.

Они и послужили мишенью для идеологического разноса. Авторы работы [6] не скупятся на резкие высказывания: “Классики марксизма-ленинизма дали нам ясные и совершенно достаточные указания для правильного понимания философских вопросов математики” (с. 331); “...изъяты всякого содержания в пользу “чистой” и субъективной формы, творящей содержание, противоречит марксизму и науке” (с. 333); “Речь идет не о том, чтобы “ликвидировать” математическую логику, а о том, чтобы отсечь реакционную тенденцию в ней, извращения ее, отражающие идеологию враждебных нам классов” (с. 336). И наконец: “Эти работы являются выражением примиренчества к идеализму в математике” (с. 337).

Редактору книг С.А. Яновской пришлось оправдываться за “идеологические просчеты”. Ее письмо по этому поводу помещено сразу же после текста погромного опуса (с. 339—342). В этом же номере журнала помещена и статья, по-видимому, призванная смягчить впечатление от разгрома формального метода в логике. Она называется “О предмете формальной логики”. Автор этой статьи М.С. Строгович пишет: “Сейчас отношение к формальной логике изменилось коренным образом: указаниями товарища И.В. Сталина формальная логика восстановлена в своих правах. На основании постановления ЦК ВКП(б) преподавание ее введено в средних шко-

лах, а также во многих высших учебных заведениях” (стр. 309). Вождь, по-видимому, вспомнил о годах своей учебы в семинарии и упомянул о пользе логики. Но что дозволено “льву”, не всегда дозволено остальным.

После математической логики настала очередь массивной атаки на те направления в физиологии, которые нарушали чистоту учения И.П. Павлова, объявленного марксистскими философами венцом учения о поведении животного и той части поведения человека, которая регулировалась его центральной нервной системой. В 1953 году наступила очередь кибернетики.

В четвертом издании “Краткого философского словаря” (1954) в статье “Кибернетика” эта наука была определена как “реакционная лженаука, возникшая в США после второй мировой войны и получившая широкое распространение и в других капиталистических странах; форма современного механицизма” [7]. В унисон с этим “определением” звучат тексты рефератов статей по кибернетике, которые в эти годы публикуются в реферативном журнале “Математика” (кстати, в инструкции, которой должны были руководствоваться авторы рефератов, было прямо сказано, что реферат должен излагать содержание работ абсолютно нейтрально, никакие оценочные суждения не должны иметь место, но, по-видимому, Д.Ю. Панов, редактировавший эти рефераты, считал, что идеология превыше декларированных редакцией журналом принципов “невмешательства”).

Апофеозом наступления на кибернетику стала статья, напечатанная в пятом номере журнала “Вопросы философии” в 1953 году [8]. Она была помещена в разделе, носившем название “Критика буржуазной идеологии” и называвшемся “Кому служит кибернетика”. Написавший этот пасквиль, по-видимому, чувствуя некоторый страх, скрылся под псевдонимом Материалист.

В конце концов не важно, кто именно и “по велению сердца” или “по заданию сверху” написал этот донос. Его появление носило знаковый характер. Это была затравка для массового наступления на кибернетику.

Как и статья, направленная против математической логики, статья против кибернетики разделяла технологический и теоретический аспекты. Все, что касалось развития вычислительной техники как таковой, когда вычислительные машины уподоблялись очень быстро работающим арифмометрам, объявлялось полезным и нужным для социалистического отечества. В подобном качестве вычислительные машины ничем не отличались от устройств, создаваемых человеком для облегчения своего труда. Но когда речь заходила об использовании этих машин для моделирования различных процессов или для символических преобразований, то натренированный на поиске идеологического криминала ум борца за чистоту марксистско-ленинского учения немедленно подавал сигнал опасности: “По мнению Винера, деятельность вычислительных машин дает ключ к познанию самых разнообразных природных и общественных явлений. Эта в корне порочная идея послужила Винеру основанием для создания новой “науки” — кибернетики” [8, с. 212].

Итак, вычислительные машины не могут внести качественно новую струю в процесс познания окружающего мира. Чтобы эта мысль дошла до всех читателей статьи, автор формулирует ее еще раз: “Теория кибернетики, пытающаяся распространить принципы действия вычислительных машин новейшей конструкции на самые различные природные и общественные явления без учета их качественного своеобразия, является механицизмом, превращающимся в идеализм. Это пустоцвет на древе познания, возникший в результате одностороннего и чрезмерного раздувания одной из черт познания” [8, с. 218].

Набор ярлыков для кибернетики (пустоцвет, лженаука, идеологическое оружие империалистической реакции, порождение лакеев империализма и т.п.) свидетельствовал, что никакой патриотически настроенный ученый в СССР не может заниматься столь одиозной наукой. Надо было немедленно свертывать все исследования в этой области.

Но, как уже говорилось, практические задачи (и прежде всего задачи укрепления обороноспособности страны) требовали не прекращения работ в области кибернетики, а расширения и активизации этих исследований. Это понимали даже партийные чиновники из оборонного отдела ЦК КПСС и отдела науки того же всемогущего ведомства. И поэтому, когда один из первых отечественных специалистов по применению вычислительных машин в военной области А.И. Китов, математик с энциклопедическим стилем мышления А.А. Ляпунов и известный своими теоретическими работами, связанными с созданием атомной бомбы, математик С.Л. Соболев объединились как авторы статьи, в которой давался ответ Материалисту, и принесли ее в тот же журнал “Вопросы философии”, то: “Как ни странно, редколлегия спорить не стала. Единственное, что они попросили сделать, так это получить на опубликование статьи разрешение ЦК КПСС” [9].

В 50-х годах высшие чиновники Коммунистической партии никогда не действовали от своего имени. Все их директивы подкреплялись “мнением широких народных масс” или специалистов в определенной области. Поэтому в отделе науки ЦК КПСС, ознакомившись с текстом статьи в защиту кибернетики и “посоветовавшись кое с кем”, сказали, что инициатива авторов статьи своевременна, но хорошо бы заручиться поддержкой их точки зрения на кибернетику среди научной общественности страны.

А.И. Китов и А.А. Ляпунов организовали серию выступлений на научных семинарах в академических институтах, высших учебных заведениях и в организациях, в которых методы кибернетики могли бы принести практическую пользу. К этой деятельности подключились их коллеги по работе в Вычислительном центре Министерства обороны и других военных организациях: М.Г. Гаазе-Рапопорт, Н.А. Криницкий, И.А. Полетаев и другие. В Московском университете идеи кибернетики нашли отклик у признанного в СССР авторитета в области математической логики А.А. Маркова, а в Институте автоматизации и телемеханики эти работы были поддержаны М.А. Айзерманом, М.А. Гавриловым и А.А. Фельд-

баумом. Известный специалист в области поведения животных Л.В. Крушинский, ознакомившись с текстом будущей статьи, занял позицию безусловной поддержки нового научного направления.

Сохранилась стенограмма одного из докладов. Он был прочитан А.А. Ляпуновым 24 июня 1954 года в Энергетическом институте АН СССР и назывался “Об использовании математических машин в логических целях”. Полемицируя с теми, кто буквально истолковывает способность машин к реализации творческих действий, Ляпунов показывает, что даже в тех случаях, когда внешне действия машины выглядят разумными и творческими (для иллюстрации он рассматривает задачу управления лифтами в высотном здании и гипотетическую в то время, но принципиально возможную задачу доказательства теорем в планиметрии), истинная творческая деятельность осуществляется не машиной, а человеком, составившим программу ее работы. Этот основной аргумент против необоснованной критики возможностей вычислительных машин Ляпунов обсуждает в своем докладе несколько раз.

Подготовка положительной реакции на дезавуирование кампании против кибернетики заняла около полутора лет. Не все проходило гладко и безболезненно. Я помню выступление А.А. Ляпунова на семинаре по машинной математике МГУ в 1954 году. Дискуссия, развернувшаяся после этого выступления, в которой самое активное участие принимали университетские философы и биологи, была настолько горячей, что пришлось сделать два продолжительных перерыва, чтобы часть возражений против кибернетики снять в процессе личных контактов.

Где-то в начале 1955 года текст статьи С.Л. Соболева, А.Т. Китова и А.А. Ляпунова попал в редакцию журнала “Вопросы философии”. На заседании редколлегии журнала ее содержание обсуждалось вместе со статьей “Что такое кибернетика” чешского философа Э.Кольмана, жившего тогда в СССР. Обсуждение носило главным образом позитивный и доброжелательный характер. За два года, прошедших со времени опубликования статьи Материалиста, в жизни страны произошли определенные перемены, сталинские методы управления наукой были уже непопулярны, а кибернетика получила массовую поддержку научно-технической интеллигенции. Обе статьи появились на страницах журнала в 1955 году [10, II].

В этих статьях нет прямой полемики с Материалистом. Необходимость в ней отпала из-за отсутствия официальной поддержки негативных отношений к кибернетике. Поэтому в [10, II] в основном излагаются принципы кибернетики и поясняется практическая значимость исследований в этой области. Все возражения Материалиста снимаются без ссылки на высказывания из [8]. Например, в [10] на стр. 141 утверждается: “Следует подчеркнуть большое методологическое значение вопроса, поставленного кибернетикой, о необходимости обобщения, объединения в широком плане результатов и достижений различных областей науки, развивающихся в известном смысле изолированно друг от друга, например, таких областей, как физиология и автоматика, теория связи и статистическая механика”. И далее на стр. 144: “Принципы работы электронных счетных машин вполне позволяют реализовать на этих машинах логические процессы, подобные процессу выработки условных рефлексов у животных и человека”. А чтобы вновь не звучали обвинения в механицизме,

на той же стр. 144 говорится: “Следует ясно представлять коренное, качественное отличие процессов мышления человека от работы счетной машины”. Эта же мысль звучит как рефрен и в самом конце статьи: “Следует вести борьбу также и против вульгаризации метода аналогий в изучении процессов высшей нервной деятельности, отвергая упрощенную, механистическую трактовку этих вопросов, тщательно исследуя границы применимости электронных и механических моделей и схем для представления процессов мышления”.

Философские размышления Э.Кольмана [11] подкрепляли основные положения статьи С.А. Соболева и его соавторов. Написанная в традиционном для читателей философского журнала стиле, она привлекала на сторону кибернетики тех, кто должен был дать идеологическую оценку новой науке.

В СССР знали, что статьи, появившиеся в таком органе, как “Вопросы философии”, выражают официальную точку зрения. Одним из свидетельств этого явилось исключение погромного текста статьи “Кибернетика” при допечатке в 1955 году тиража 4-го издания “Философского словаря”. Борьба против кибернетики была в основном закончена, люди, отстаивавшие новую науку, победили.

4. Начальный период

К этому периоду можно отнести время с 1955 года до создания в 1959 году в АН СССР Научного совета по комплексной проблеме “Кибернетика”. За эти пять лет в СССР возникла инфраструктура, поддерживающая новое научное направление.

Своим возникновением она обязана А.И. Бергу, чей талант организатора науки позволил преодолеть все препоны и рогатки бюрократического государства. В лице адмирала Берга, в 1953—1957 годах занимавшего пост заместителя министра обороны СССР по радиоэлектронике, кибернетика обрела того человека, который обеспечил этой науке условия для ее становления и расцвета.

А.И. Берг начал свою деятельность не на пустом месте. К концу 1957 года, когда он освободился от поглощающих все время обязанностей крупного военного руководителя и смог начать полноценную научно-организационную деятельность, уже был заложен солидный фундамент под будущую инфраструктуру.

Продолжались разработка новых вычислительных машин и развитие методов решения на них разнообразных задач. В 1948 году были созданы Институт точной механики и вычислительной техники АН СССР и Специальное конструкторское бюро Министерства приборостроения и средств автоматизации (СКБ-245). В них, а также в ряде других организаций АН СССР и различных ведомств (в лаборатории электросистем Энергетического института им. Г.М. Кржижановского (позднее — Институт электронных управляющих машин), НИИ электронных математических машин в Москве, лаборатории вычислительной техники Института математики АН УССР в Киеве (позже преобразованной в ВЦ АН УССР), Ереванском институте математических машин; Пензенском институте управляющих вычислительных машин) активно развивалась теория вычислительных машин, разрабатывалась технология программирования. Теоретические исследования активно проводились в Московском, Ленинградском и Киевском

университетах, Институте автоматики и телемеханики АН СССР, в созданном в 1955 году Вычислительном центре АН СССР.

В 1950 году в ИТМиВТ АН СССР начал работать первый постоянный семинар по программированию, которым руководил Л.А. Люстерник. В 1952 году в МГУ была создана кафедра вычислительной математики (кафедру возглавил С.А. Соболев), для студентов и аспирантов которой в 1952/53 учебном году А.А. Ляпунов впервые прочитал курс “Принципы программирования”. В 1953 году в отделе прикладной математики Математического института АН СССР был создан во главе с А.А. Ляпуновым отдел программирования [12]. В этом же году появилась первая доступная всем интересующимся этой областью книга по программированию [13]. В 1955 году был создан Вычислительный центр МГУ, специализирующийся на разработке и применении вычислительных методов для решения сложных научных и прикладных задач.

В конце 50-х годов был получен ряд результатов, стоящих на уровне мировых достижений.

- Была разработана теория логического анализа и синтеза релейно-контактных, а позже и функциональных схем, в которой аппарат математической логики был использован в области технических наук. Начатые в 40-х годах М.А. Гавриловым в Институте автоматики и телемеханики АН СССР [14], эти работы были продолжены О.Б. Лупановым и С.В. Яблонским в отделе прикладной математики АН СССР. В результате этих исследований в СССР возникли две научные школы, сыгравшие важную роль в создании теории дискретных управляющих устройств и методов инженерного проектирования устройств такого типа (в частности, схем, узлов и устройств вычислительной техники).

- В 1952—1953 годах А.А. Ляпуновым был предложен операторный метод для описания программ [15]. Практически впервые был создан способ представления программ на обозримом уровне. Вместо неэффективного для человека задания программ в машинных кодах А.А. Ляпунов предложил формализованное представление высокого уровня. Особенно важным было то, что операторный метод позволял создать теорию синтаксических структур программ [16].

- В 1953 году А.А. Ляпунов сформулировал постановку задачи автоматизации программирования. Эта оригинальная постановка была успешно использована в первых отечественных трансляторах, называвшихся тогда программирующими программами. Летом 1954 года появилась программирующая программа ПП-1 (отдел прикладной математики Института математики АН СССР), а в 1955 году — ее улучшенный вариант ПП-2. Чуть позднее была создана еще одна программирующая программа [17].

- В 1953—1954 годах Л.В. Канторович разработал технологию крупноблочного программирования, которая также давала обозримое описание программ и обеспечивала степень формализации, достаточную для исследования синтаксических структур программ и создания программирующих программ [18].

- М.А. Цетлин впервые поставил вопрос о возможности моделирования с помощью простейших технических средств

сложных форм поведения. Эти идеи послужили началом создания теории коллективного поведения технических систем, намного опередившей аналогичные исследования в других странах [19].

К середине 50-х годов у ведущих специалистов в области вычислительной техники было ясное представление о путях развития отечественной информатики. Примером может служить статья В.М. Глушкова, работавшего тогда в лаборатории вычислительной техники Института математики АН УССР в Киеве [20]. В середине 1957 года автор статьи четко определяет направления стратегических исследований в области информатики. По мнению В.М. Глушкова, основой прогресса развития вычислительных машин должны стать теория их работы, разработка методов автоматизации проектирования ЭВМ и развитие методов автоматизации программирования. Он подчеркивает важную роль исследований в области теории алгоритмов и теории конечных детерминированных и стохастических автоматов, принципиальное значение разработки методов символических преобразований на ЭВМ (аналитических преобразований, доказательства теорем, машинного перевода), отмечает центральную роль, которую играет задача оптимизации программ (особенно для управляющих машин), а также указывает на обратное влияние развития вычислительных машин на дальнейшие работы в области вычислительной математики.

С 1953 года в нашей стране налажен серийный выпуск вычислительных машин. Первой в серию пошла “Стрела”, созданная в СКБ-245 под руководством Ю.Я. Базилевского. В 1958 году в серию пошла машина М-20, созданная в коллективе С.А. Лебедева в ИТМиВТ АН СССР. Эта машина сыграла большую роль в развитии программирования, а позже на ее базе была создана транзисторная машина М-220. Таким образом, к началу 60-х годов были заложены теоретические и технические основы для развития информатики.

В Московском, Ленинградском и Киевском университетах началась подготовка специалистов по вычислительной математике, а в ряде технических высших учебных заведений появились курсы по вычислительной технике, а затем стали открываться кафедры вычислительной техники или вычислительных машин.

Свидетельством окончательного официального признания кибернетики стала статья “Кибернетика” в 51-м томе второго издания Большой советской энциклопедии, написанная А.Н. Колмогоровым. В ней нет и намек на ту травлю, которой подверглась эта наука всего несколько лет назад. Начало статьи необычно для советских изданий того времени, тем более для БСЭ, которая должна была стоять на страже советской науки, всячески выпячивать ее мировой авторитет и значимость: “Кибернетика — научное направление, задачи которого были сформулированы в работах американского ученого Н. Винера, опубликованных в 1948-м; по Винеру и его последователям, кибернетика есть наука о “связи”, “управлении” и “контроле” в машинах и живых организмах”. Далее в статье расшифровывается содержание понятий, взятых в кавычки в определении кибернетики, и устанавливается связь кибернетики с теорией информации, опирающейся на идеи К.Шеннона. Отголоском

недавней борьбы за кибернетику выглядит абзац, по своему стилю выпадающий из общего стандартного для энциклопедий “академического” тона изложения: “Много дискутировавшийся вопрос о праве кибернетики на существование в качестве самостоятельной научной дисциплины сводится к вопросу о том, насколько существенны общие черты всех процессов связи, управления и контроля, т.е. могут ли общие свойства этих процессов в машинах, живых организмах и их объединениях быть предметом достаточно содержательной единой теории. На этот вопрос следует ответить с полной определенностью утвердительно, хотя в направлении систематического построения кибернетики сделаны лишь первые шаги” [21].

Вскоре появились книги отечественных специалистов [22—25]. В 1958 году вышел “Сигнал” И.А. Полетаева. Как уже говорилось, все авторы первых отечественных книг по кибернетике были военными, работали в учебных заведениях, готовивших кадры для Министерства обороны. Академии имени Жуковского и Дзержинского в Москве, Академия имени Можайского в Ленинграде, Харьковское высшее авиационно-техническое училище и Киевское высшее инженерно-радиотехническое училище были первыми военными учебными заведениями, где преподавание кибернетики стало обязательным.

Важным событием было появление перевода основополагающей для кибернетики книги Норберта Винера. Первое издание [26] вышло с десятилетним опозданием относительно оригинала (с купюрами: все места, которые можно было интерпретировать как критику нашего строя или экономического уклада, были изъяты). Переводчиком книги был И.В. Соловьев, а редактором — Г.Н. Поваров, который в эти годы начал активно заниматься логическими методами анализа и синтеза схем. Позже книга была переиздана с учетом второго, расширенного американского издания, появившегося в 1961 году в более полном объеме [27]. Была издана и вторая книга Н. Винера, наделавшая в США в 1954 году много шума из-за своей направленности против многих сторон жизни капиталистического общества [28].

Затем издаются переводы книг других зарубежных авторов [29—31]. Наконец, появились отечественные научно-популярные книги, пропагандирующие идеи и достижения кибернетики [32, 33].

Не только военные, но и политические руководители страны стали уповать на помощь вычислительных машин. Статья одного из создателей первых отечественных машин, помещенная в главном идеологическом органе партии [34], говорит о необходимости использования компьютеров в задачах планирования народного хозяйства. Это показывает, что недоверие к вычислительной технике и кибернетике в верхних эшелонах власти стало коренным образом меняться.

Но, пожалуй, главным событием этого времени стал выход в 1958 году первого выпуска “Проблем кибернетики” — детища А.А. Ляпунова и его единомышленников. Этот нарядный, в ярко-красной суперобложке том открыл собою серию из 41 сборника, во многом определившую пути

Становление информатики в России

Продолжение. Начало на с. 7—9

развития теоретической кибернетики и того, что потом стало называться информатикой. До переезда Ляпунова в Новосибирск в 1962 году “Проблемы кибернетики” в значительной мере отражали интересы участников семинара по кибернетике, проходившего под его руководством начиная с 1954/55 учебного года на механико-математическом факультете МГУ. Первый выпуск содержал изложение идей доклада, прочитанного М.В. Келдышем, А.А. Ляпуновым и М.Р. Шура-Бурой на октябрьской сессии АН СССР 1956 года [15].

Роль этого семинара в истории отечественной информатики огромна. На протяжении почти двадцати лет (до смерти А.А. Ляпунова в 1973 году) этот семинар во многом определял высокий уровень работ в области кибернетики. Как отмечается в [35], всего было проведено 141 заседание семинара. В работе семинара принимали активное участие математики, физиологи, лингвисты, управленцы и представители других наук. Это был первый в истории нашей науки по-настоящему междисциплинарный семинар. Многие его участники в последующие годы стали крупными учеными в области информатики. Знакомство с темами докладов, прочитанных на семинаре (см. приложение к статье [35]), показывает, сколь широк был спектр интересов его участников.

Семинар был не только чисто научным мероприятием. А.А. Ляпунов использовал его возможности и для осуществления научно-организационной деятельности. На семинаре обсуждались рукописи новых книг отечественных авторов, принимались после обсуждения рекомендации по переводу наиболее интересных зарубежных книг (при этом, как правило, переводчики и редакторы переводов находились среди участников семинара), обсуждались структура и научные задачи учреждений, которые должны были создаваться в области кибернетики. Не без поддержки (в той или иной форме) семинара увидели свет книги [36—39]. С 1960 года начал выходить “Кибернетический сборник”, в котором оперативно публиковались переводы наиболее интересных зарубежных статей по кибернетике. Во главе этого начинания стояли А.А. Ляпунов и О.Б. Лупанов. После переезда Ляпунова в Новосибирск Лупанов продолжил это весьма важное для оторванных от источников зарубежной информации специалистов СССР дело.

По образу и подобию “большого” семинара стали создаваться семинары по кибернетике и в других местах. Наиболее известным стал среди них семинар секции кибернетики при Ленинградском доме ученых. Эта секция была создана в ноябре 1956 года, и ее первым председателем был Л.В. Канторович. Потом его на этом посту сменил Л.П. Крайзмер, оставшийся до настоящего времени бессменным председателем секции и проводимого ею семинара. Это наиболее долговечный кибернетический семинар в СССР.

В середине 50-х годов начал работать семинар по теории автоматов на физическом факультете МГУ [40], бес-

сменным руководителем которого до своего преждевременного ухода из жизни был М.Л. Цетлин. Этот семинар стал центром, вокруг которого со временем сложилась отечественная школа в области коллективного поведения автоматов. Модели такого типа носили ярко выраженный кибернетический характер, а их исследование было немислимо без специальных приемов моделирования на ЭВМ. В семинаре принимали активное участие не только физики, но и физиологи, программисты, математики. Неординарная и многогранная личность руководителя [41] привлекала к участию в работе семинара по поведению автоматов широкие круги научной общественности. Тематика семинара связывала между собой задачи логического анализа и синтеза схем с проблемами машинного моделирования и моделями поведения живых систем.

Чуть позже начал работать семинар в Киеве под руководством В.М. Глушкова. В его работе принимали участие медики, биологи и философы. Со временем от “большого” семинара отпочковался “малый” семинар, получивший название “чайкофского” (часть его заседаний, во время которых молодые участники семинара В.Г. Боднарчук, Ю.В. Капитонова, А.А. Летичевский, М.А. Спивак и другие решали задачи, поставленные перед ними руководителем семинара В.М. Глушковым, проходили на Крещатики, в заведении, называвшемся “Чай — кофе”). На этом семинаре обсуждались вопросы абстрактной теории автоматов, созданием которой в эти годы вместе со своими учениками активно занимался руководитель семинара.

Математическая логика заняла достойное место в математическом образовании. В 1958 году в МГУ была открыта первая в СССР кафедра математической логики. Кафедру возглавил А.А. Марков.

В эти годы было проведено несколько научных мероприятий, в программах которых предусматривались доклады и сообщения, посвященные решению задач в рамках кибернетического подхода. В октябре 1956 года состоялась сессия АН СССР по научным проблемам автоматизации производства. На ней была дана положительная оценка роли кибернетики в решении практических задач автоматизации. Перед этим в том же году, во время проведения III Всесоюзного математического съезда, работала специальная секция кибернетики, на которой был сделан ряд докладов по программированию и теоретическим проблемам новой науки. Важное значение для дальнейшего развития кибернетики имела прошедшая в марте 1956 года в Москве Всесоюзная конференция “Пути развития советского математического машиностроения и приборостроения”. На ней было сделано несколько докладов, связанных с программированием и с использованием вычислительных машин для решения разнообразных задач. В феврале 1958 года в Баку прошло Всесоюзное совещание по вычислительной математике и применению средств вычислительной техники. В работе Всесоюзного совещания по вычислительной математике и вычислительной технике, проходившего в МГУ в ноябре 1959 года, в программе которого была и секция кибернетики, приняли участие свыше 2000 человек.

Хотя и с огромным трудом, отечественные специалисты в области информатики стали выезжать за рубеж для участия в научных мероприятиях в данной области. Из [35] можно узнать, что на

семинаре А.А. Ляпунова о своих зарубежных поездках отчитывались члены советских делегаций, выезжавших на конгресс по кибернетике в Намюре в 1957 году, в США и Великобританию. На симпозиуме “Механизмы мыслительных процессов”, проходившем в ноябре 1958 года в Национальной физической лаборатории в Теддингтоне, А.П. Ершов сделал два доклада: “О работах в ВЦ АН СССР в области теоретического программирования” и “О работах в ВЦ АН СССР в области автоматизации программирования”. Эти доклады вызвали положительные отклики за рубежом. Достижения советской науки в области программирования получили весьма высокую оценку. На основе этих докладов А.П. Ершов подготовил по просьбе редколлегии известного в то время журнала “Datamation” обзор-

ную статью о программировании в СССР [42]. Эта статья была, наверное, второй после статьи [43] публикацией о достижениях отечественных специалистов в области информатики, напечатанной в зарубежном издании.

Признанием важности информатики стала защита диссертаций в этой области. Первой, по-видимому, была защищена кандидатская работа, написанная Э.З. Любимским на тему “Об автоматизации программирования и методе программирующих программ” (1957). Вскоре за ней последовали защиты других специалистов: “О равносильности и преобразованиях схем программ” (Ю.И. Янов, 1958), “Матричный метод анализа схем и некоторые его приложения” (М.Л. Цетлин, 1958), “Операторные алгоритмы” (А.П. Ершов, 1961). Защиты диссертаций проходили на механико-математическом и физическом факультетах МГУ, в Математическом институте АН СССР и в Институте математики с вычислительным центром СО АН СССР. Высокая марка научных советов этих ведущих учреждений обеспечивала серьезное отношение к выполненным работам.

ЛИТЕРАТУРА

1. А.И. Михайлов, А.И. Черный, Р.С. Гиляревский. Информатика. // Большая советская энциклопедия, 3-е изд., т. 10. М.: Советская энциклопедия, 1972. С. 348—350.
2. Encyclopedia of Computer Science. 3rd Edition. A. Ralston, E.D. Reilly (Eds.). New York: Van Nostrand Reinhold, 1993.
3. И.А. Апокин. Развитие вычислительной техники и систем на ее основе. // Новости искусственного интеллекта, 1994, № 1. С. 26—69.
4. Б.Н. Малиновский. История вычислительной техники в лицах. Киев: Наукова думка, 1995.
5. Л.А. Люстерник, А.А. Абрамов, В.И. Шестаков, М.Р. Шура-Бура. Решение математических задач на автоматических цифровых машинах. Программирование для быстродействующих электронных счетных машин. М.: Издательство АН СССР, 1952.
6. В.П. Тугаринов, Л.Е. Майстров. Против идеализма в математической логике. // Вопросы философии, 1950, № 3. С. 331—339.
7. Краткий философский словарь, под ред. М. Розенталя, П. Юдина, 4-е изд., дополненное и исправленное. М.: Государственное издательство политической литературы, 1954. С. 236—237.
8. Кому служит кибернетика. // Вопросы философии, 1953, № 5. С. 210—219.
9. В. Нескоромный. Человек, который вынес кибернетику из секретной библиотеки. // Компьютерра, 18.11.1996, № 43. С. 44—45.
10. С.Л. Соболев, А.И. Китов, А.А. Ляпунов. Основные черты кибернетики. // Вопросы философии, 1955, № 4. С. 136—148.
11. Э. Кольтман. Что такое кибернетика? // Вопросы философии, 1955, № 4. С. 148—159.
12. А.П. Ершов, М.Р. Шура-Бура. Становление программирования в СССР. Начальное развитие. // Препринт ВЦ СО АН СССР № 12, 1976; А.П. Ершов, М.Р. Шура-Бура. Становление программирования в СССР. Переход ко второму поколению языков и машин. // Препринт ВЦ СО АН СССР № 13, 1976.
13. М. Уилкс, Д. Уилер, С. Гилл. Составление программ для электронных счетных машин. М.: Издательство иностранной литературы, 1953.
14. М.А. Гаврилов. Теория релейно-контактных схем. Анализ и синтез структуры релейно-контактных схем. М. — Л.: Издательство АН СССР, 1950.
15. А.А. Ляпунов. О некоторых общих вопросах кибернетики. // Проблемы кибернетики, 1958, вып. 1. С. 5—22.
16. А.А. Ляпунов, Ю.И. Янов. О логических схемах программ. // Проблемы кибернетики, 1958, вып. 1. С. 46—74.
17. А.П. Ершов. Программирующая программа для быстродействующей электронной счетной машины. М.: Издательство АН СССР, 1958.
18. Л.В. Канторович, Л.Т. Петрова, М.А. Яковлева. Об одной системе программирования. // Пути развития математического машиностроения и приборостроения. Всесоюзная конференция, часть III. М.: ВИНТИ, 1956. С. 30—36.
19. М.Л. Цетлин. Исследования по теории автоматов и моделированию биологических систем. М.: Наука, 1969.
20. В.М. Глушков. О некоторых задачах вычислительной техники и связанных с ними задачах математики. // Украинский математический журнал, 1957, № 4. С. 369—376.
21. А.Н. Колмогоров. Кибернетика. // Большая советская энциклопедия, 2-е изд., т. 51. М.: Большая советская энциклопедия, 1958. С. 149—151.
22. А.В. Китов. Цифровые вычислительные машины. М.: Советское радио, 1956.
23. И.А. Полетаев. Сигнал. М.: Советское радио, 1958.
24. Ю.И. Соколовский. Кибернетика настоящего и будущего. Харьков: Харьковское книжное издательство, 1959.
25. А.И. Китов, Н.А. Криницкий, П.Н. Комолов. Элементы программирования. М.: Издательство Артиллерийской инженерной академии, 1956.
26. Н. Винер. Кибернетика, или Управление и связь в животном и машине. М.: Советское радио, 1958.
27. Н. Винер. Кибернетика, или Управление и связь в животном и машине. 2-е изд. М.: Советское радио, 1968.
28. Н. Винер. Кибернетика и общество. М.: Издательство иностранной литературы, 1958.
29. Цинь Сюэ-сэнь. Техническая кибернетика. М.: Советское радио, 1956.
30. П. Косса. Кибернетика. М.: Издательство иностранной литературы, 1958.
31. У.Р. Эшби. Введение в кибернетику. М.: Издательство иностранной литературы, 1959.
32. Н.Е. Кобринский, В.Д. Пекелис. Быстрее мысли. М.: Молодая гвардия, 1959.
33. Л. Теплов. Очерки о кибернетике. М.: Московский рабочий, 1959.
34. И. Брук. Электронные вычислительные машины — на службу народному хозяйству. // Коммунист, 1957, № 7. С. 124—127.
35. М.Г. Гаазе-Рапопорт. О становлении кибернетики в СССР. // Кибернетика: прошлое для будущего. Этюды по истории отечественной кибернетики. Теория управления. Автоматика. Биобибернетика. М.: Наука, 1989. С. 46—85.
36. О.С. Кулагина. Некоторые теоретические вопросы машинного перевода. М.: Математический институт АН СССР, 1958.
37. Е.А. Жоголев, Г.С. Росляков, Н.П. Трифонов, М.Р. Шура-Бура. Система стандартных подпрограмм. М.: Государственное издательство физико-математической литературы, 1958.
38. Л.Н. Королев. Некоторые вопросы теории машинного словаря. М.: ИТМиВТ, 1959.
39. А.И. Китов, Н.А. Криницкий. Электронные цифровые машины и программирование. М.: Государственное издательство физико-математической литературы, 1959.
40. В.Л. Стефанюк. От автоматов М.Л. Цетлина к искусственному интеллекту (этапы и вехи, или как это было). // Новости искусственного интеллекта, 1995, № 4. С. 56—92.
41. В.В. Иванов. Из истории кибернетики в СССР. Очерк жизни и деятельности М.Л. Цетлина. // Вопросы кибернетики. Кибернетика и логическая формализация. Аспекты истории и методологии. М.: Научный совет по комплексной проблеме “Кибернетика” АН СССР, 1982. С. 166—190.
42. Andrei P. Ershov. Automatic Programming in the Soviet Union. // Datamation, 1959, v. 5, No. 4, p. 14—20.
43. F. Bryzgalin. USSR Struggles with Electronic Computers. // Petroleum Engineer, v. 27, August, 1955, p. A58.

Задачи по программированию

Ю.А. СОКОЛИНСКИЙ

Окончание. См. № 18/99

Тема: "МАССИВЫ"

Указание к задачам 6—10:

а) Все массивы имеют целый тип.
б) Размерность и значения элементов массивов указываются непосредственно в программах как константы.

Рекомендация к задачам 6—8:

Для слабых учащихся решение любой из этих задач целесообразно разбить на два урока. На первом рассмотреть цикл ввода исходных данных с контролем (см. вступление к статье), ввести и отладить соответствующую часть программы. На втором уроке разобрать цикл типа while и написать содержательную часть программы. Подробное пояснение решения этой части, приведенное для задачи 6, относится также к задачам 7 и 8.

Задача 6. Заданы возрастающая последовательность X_0, X_1, \dots, X_n и вещественное число x , $X_0 < x < X_n$, $x \neq X_i$, $i = 0..n$. Тогда число x содержится в некотором интервале X_{i-1}, X_i , т.е. $X_{i-1} < x < X_i$ (см. рисунок). Найти номер (число i) и границы этого интервала.

Решение. Последовательно просматриваем интервалы начиная с первого, пока не встретим интервал, содержащий x . Реализуем этот процесс с помощью цикла пока:

```
i:=1
нц пока x > Xi
| i:=i+1
кц
```

Если последовательность X_0, \dots, X_n в программе задать массивом $aX[0..N]$, то этот цикл будет иметь следующий вид:

на Паскале: while (x>aX[i]) do i:=i+1;
на Бейсике: DO WHILE x > aX(i) i = i + 1: LOOP
на Си: while (x>aX[i]) i++;

Приводим программы (напомним, что при вводе числа x контролируется условие $X_0 < x < X_n$).

Язык Паскаль

```
{Указать номер и границы интервала, содержащего заданное число}
const
  N=5;{число интервалов}
  aX:array[0..N] of integer=(1, 3, 7, 10, 12, 18);
var
  x:real; i:integer;
begin
  repeat {Начало цикла контроля при вводе x}
    writeln('Введите число x такое, что x>',aX[0],' и x<',aX[N]);
    readln(x);
  until (aX[0]<x) and (x<aX[N]); {Конец цикла контроля при вводе x}
  i:=1;
  while (x>aX[i]) do i:=i+1;
  writeln('Номер интервала, содержащего заданное число,=',i);
  writeln('Границы интервала: ',aX[i-1],' и ',aX[i]);
end.
```

Язык Бейсик

```
'Указать номер границы и интервала, содержащего заданное число
DIM N AS INTEGER, i AS INTEGER, x AS SINGLE
N = 5
DIM aX(0 TO N) AS INTEGER
aX(0) = 1: aX(1) = 3: aX(2) = 7: aX(3) = 10: aX(4) = 12: aX(5) = 18
DO 'Начало цикла контроля при вводе x
PRINT "Введите число x такое, что x>"; aX(0); " и x<"; aX(N)
INPUT x
LOOP UNTIL aX(0) < x AND x < aX(N)
'Конец цикла контроля при вводе x
i = 1
DO WHILE x > aX(i) i = i + 1
LOOP
PRINT "Номер интервала, содержащего заданное число, ="; i
PRINT "Границы интервала: "; aX(i - 1); " и "; aX(i)
END
```

Язык Си

```
/*Указать номер и границы интервала, содержащего заданное число*/
#include<stdio.h>
#define N 6 /*Число элементов массива*/
void main()
{int i,aX[N]={1, 3, 7, 10, 12, 18};
float x;
do /*Начало цикла контроля при вводе x */
{printf
("\nВведите число x такое, что x>=%d и x<=%d\n",aX[0],aX[N-1]);
scanf("%f",&x);
}
while (x<=aX[0]||x>=aX[N-1]); /* Конец цикла контроля при вводе x */
i=1; while (x>aX[i]) i++;
printf("Номер интервала, содержащего заданное число=%d",i);
printf("\nГраницы интервала: %d и %d",aX[i-1],aX[i]);
}
```

Задача 7. Заданы возрастающая последовательность X_0, X_1, \dots, X_n и вещественное число x , находящееся в промежутке X_0, X_n , т.е. $X_0 < x < X_n$, $x \neq X_i$, $i = 0..n$. Указать элемент последовательности, ближайший к x .

Решение. Как и в предыдущей задаче, находим интервал X_{i-1}, X_i , содержащий x , с помощью цикла пока:

```
i:=1
нц пока x > Xi
| i:=i+1
кц
```

Пусть $Near$ — значение элемента последовательности, ближайшего к x , а X_m — середина найденного интервала. Тогда $Near$ находится таким образом:

если $x < X_m$ то $Near = X_{i-1}$ иначе $Near = X_i$

Программы, реализующие этот алгоритм:

Язык Паскаль

```
{Элемент последовательности, ближайший к заданному числу}
const
  N=5;{число интервалов}
  aX:array[0..N] of integer=(1, 3, 7, 10, 12, 18);
var
  Xm,x:real; i,Near:integer;
begin
  repeat
    writeln('Введите число x такое, что x>',aX[0],' и x<',aX[N]);
    readln(x);
  until (aX[0]<x) and (x<aX[N]);
  i:=1;
  while (x>aX[i]) do i:=i+1;
  Xm:=(aX[i-1]+aX[i])/2;
  if x<Xm then Near:=aX[i-1] else Near:=aX[i];
  writeln('Элемент последовательности, ближайший к x: ',Near);
end.
```

Язык Бейсик

```
'Элемент последовательности, ближайший к заданному числу
DIM N AS INTEGER, i AS INTEGER, Near AS INTEGER
DIM x AS SINGLE, Xm AS SINGLE
N = 5 'Число интервалов
DIM aX(0 TO N) AS INTEGER
aX(0) = 1: aX(1) = 3: aX(2) = 7: aX(3) = 10: aX(4) = 12: aX(5) = 18
DO
PRINT "Введите число x такое, что x>"; aX(0); " и x<"; aX(N)
INPUT x
LOOP UNTIL aX(0) < x AND x < aX(N)
i = 1
DO WHILE x > aX(i): i = i + 1: LOOP
Xm = (aX(i - 1) + aX(i)) / 2
IF x < Xm THEN Near = aX(i - 1) ELSE Near = aX(i)
PRINT "Элемент последовательности, ближайший к x: "; Near
END
```

Язык Си

```
/*Элемент последовательности, ближайший к заданному числу*/
#include<stdio.h>
#define N 6 /*Число элементов массива*/
void main()
{int Near,i,aX[N]={1, 3, 7, 10, 12, 18};
float x,Xm;
do
{printf
("\nВведите число x такое,что x>%d и x<%d\n",aX[0],aX[N-1]);
scanf("%f",&x);
}
while (x<=aX[0] || x>=aX[N-1]);
i=1; while (x>aX[i]) i++;
Xm=(aX[i-1]+aX[i])*0.5;
if (x<Xm) Near=aX[i-1]; else Near=aX[i];
printf("Элемент последовательности, ближайший к x: %d",Near);
}
```

Задача 8. Ступенчатая функция $Fstep(x)$ определяется следующим образом. Заданы "узлы", т.е. возрастающая последовательность точек на оси X : X_0, X_1, \dots, X_n .

Промежуток X_0, X_n — область определения функции. В каждом узле значение функции известно: $Fstep(X_i) = Y_i$. Для каждой точки i -го интервала, кроме его правого конца, значение функции одно и то же и, следовательно, равно Y_{i-1} :

$Fstep(x) = Y_{i-1}$ при $X_{i-1} \leq x < X_i$

Задана точка x , входящая в область определения функции. Найти значение $Fstep(x)$.

Решение. Как и в двух предыдущих задачах, сначала находим интервал X_{i-1}, X_i , содержащий x , с помощью цикла пока:

```
i:=1
нц пока x > Xi
| i:=i+1
кц
```

Продолжение на с. 12

11

ЗАДАЧИ

1999 № 19 ИНФОРМАТИКА

Задачи по программированию

Продолжение. Начало на с. 11

Теперь находим значение функции:

если $x < X_i$ то $Fstep = Y_{i-1}$ иначе $Fstep = Y_i$

Приведем программы вычисления ступенчатой функции.

Язык Паскаль

```
{Значение ступенчатой функции}
const
N=5;{число интервалов}
aX:array[0..N] of integer=(1, 3, 7, 10, 12, 18);
aY:array[0..N] of integer=(21, 13, 5, 15, 17, 25);
var
x:real; i,Fstep:integer;
begin
  repeat
    writeln('Введите число x такое, что x>=',aX[0],' и x<=',aX[N]);
    readln(x);
  until (aX[0]<=x) and (x<=aX[N]);
  i:=1;
  while (x>aX[i]) do i:=i+1;
  if x<aX[i] then Fstep:=aY[i-1] else Fstep:=aY[i];
  writeln('Fstep(x)=',Fstep);
end.
```

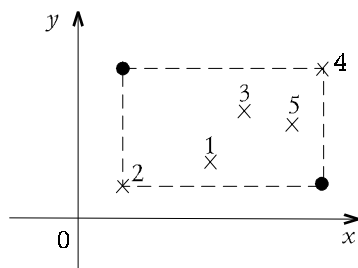
Язык Бейсик

```
'Значение ступенчатой функции
DIM N AS INTEGER, i AS INTEGER, Fstep AS INTEGER, x AS SINGLE
N = 5
DIM aX(0 TO N) AS INTEGER, aY(0 TO N) AS INTEGER
aX(0) = 1: aX(1) = 3: aX(2) = 7: aX(3) = 10: aX(4) = 12: aX(5) = 18
aY(0) = 9: aY(1) = 5: aY(2) = 4: aY(3) = 15: aY(4) = 17: aY(5) = 25
DO
PRINT "Введите число x такое, что x>="; aX(0); " и x<="; aX(N)
INPUT x
LOOP UNTIL aX(0) <= x AND x <= aX(N)
i = 1
DO WHILE x > aX(i): i = i + 1: LOOP
IF x < aX(i) THEN Fstep = aY(i - 1) ELSE Fstep = aY(i)
PRINT "Fstep(x)="; Fstep
END
```

Язык Си

```
/*Значение ступенчатой функции*/
#include<stdio.h>
#define N 6 /*Число элементов массивов aX и aY*/
void main()
{int i, Fstep; float x;
int aX[N]={1, 3, 7, 10, 12, 18}, aY[N]={21, 13, 5, 15, 17, 25};
do
{printf ("\nВведите число x такое, что x>=%d
и x<=%d\n",aX[0],aX[N-1]);
scanf("%f",&x);
}
while (x<aX[0] || x>aX[N-1]);
i=1; while (x>aX[i]) i++;
if (x<aX[i]) Fstep=aY[i-1]; else Fstep=aY[i];
printf("Fstep(x)=%d",Fstep);
}
```

Задача 9. На плоскости даны n точек $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$. Рассмотрим прямоугольники, содержащие эти точки, причем стороны прямоугольников параллельны координатным осям. Возьмем наименьший из них. Определить координаты его противоположных углов — левого верхнего и правого нижнего (см. рисунок).



Решение. Ясно, что надо найти максимальный и минимальный элементы двух последовательностей:

X_{max}, X_{min} для x_1, x_2, \dots, x_n , а также Y_{max}, Y_{min} для y_1, y_2, \dots, y_n .

Тогда координаты левого верхнего угла — это (X_{min}, Y_{max}) , а правого нижнего — (X_{max}, Y_{min}) . Поиск четырех экстремальных элементов двух последовательностей можно выполнить с помощью одного цикла для:

```
Xmax = x1; Xmin = x1; Ymax = y1; Ymin = y1
нц для i от 2 до n
  если xi > Xmax то Xmax = xi
  если xi < Xmin то Xmin = xi
  если yi > Ymax то Ymax = yi
  если yi < Ymin то Ymin = yi
кц
```

Осталось привести программы.

Язык Паскаль

```
{Наименьший прямоугольник, содержащий заданные точки}
const
n=5;{Число точек}
x:array[1..n] of integer=(23, 10, 38, 60, 47);
y:array[1..n] of integer=(12, 7, 25, 55, 31);
var i, Xmax, Xmin, Ymax, Ymin:integer;
begin
  Xmax:=x[1]; Xmin:=x[1]; Ymax:=y[1]; Ymin:=y[1];
  for i:=2 to n do begin
    if Xmax<x[i] then Xmax:=x[i];
    if Xmin>x[i] then Xmin:=x[i];
    if Ymax<y[i] then Ymax:=y[i];
    if Ymin>y[i] then Ymin:=y[i];
  end;
  writeln('Координаты левого верхнего угла: ',Xmin,', ',Ymax);
  writeln('Координаты правого нижнего угла: ',Xmax,', ',Ymin);
end.
```

Язык Бейсик

```
'Наименьший прямоугольник, содержащий заданные точки
DEFINT I, N, X-Y
n = 5 'Число точек
DIM x(1 TO n), y(1 TO n)
x(1) = 23: x(2) = 10: x(3) = 38: x(4) = 60: x(5) = 47
y(1) = 12: y(2) = 7: y(3) = 25: y(4) = 55: y(5) = 31
Xmax = x(1): Xmin = x(1): Ymax = y(1): Ymin = y(1)
FOR i = 2 TO n
  IF Xmax < x(i) THEN Xmax = x(i)
  IF Xmin > x(i) THEN Xmin = x(i)
  IF Ymax < y(i) THEN Ymax = y(i)
  IF Ymin > y(i) THEN Ymin = y(i)
NEXT i
PRINT "Координаты левого верхнего угла: "; Xmin; ", "; Ymax
PRINT "Координаты правого нижнего угла: "; Xmax; ", "; Ymin
END
```

Язык Си

```
/*Наименьший прямоугольник, содержащий заданные точки*/
#include<stdio.h>
#define N 5 /*Число точек*/
void main()
{int x[N]={23, 10, 38, 60, 47}, y[N]={12, 7, 25, 55, 31};
int i, Xmax, Xmin, Ymax, Ymin;
Xmax=x[0]; Xmin=x[0]; Ymax=y[0]; Ymin=y[0];
for (i=1; i<N; i++)
{if (Xmax<x[i]) Xmax=x[i];
if (Xmin>x[i]) Xmin=x[i];
if (Ymax<y[i]) Ymax=y[i];
if (Ymin>y[i]) Ymin=y[i];
}
printf("\nКоординаты левого верхнего угла: %d, %d",Xmin,Ymax);
printf("\nКоординаты правого нижнего угла: %d, %d",Xmax,Ymin);
}
```

Задача 10*. (В этой задаче сложным для учащихся представляется алгоритм, основанный на методе индукции. Программа же получается достаточно простой.)

Найти два наибольших элемента последовательности x_1, x_2, \dots, x_n , где $n \geq 3$.

Решение. Пусть $Max1$ — наибольший элемент последовательности, а $Max2$ — следующий за ним по величине. Предположим, что мы нашли эти величины для последовательности, состоящей из первых $i-1$ элементов. Добавим i -й элемент. Тогда новые значения $Max1, Max2$ можно найти так:

если $x_i \leq Max1$ то нач $Max2 = Max1$; $Max1 = x_i$ все
иначе если $x_i > Max2$ то $Max2 = x_i$

Чтобы начать этот процесс, надо упорядочить первые два элемента:

если $x_1 > x_2$ то нач $Max1 = x_1$; $Max2 = x_2$ все
иначе нач $Max1 = x_2$; $Max2 = x_1$ все

Теперь выполняем описанный процесс начиная с $i = 3$ и кончая n . В результате получим два наибольших элемента всей последовательности.

Программы, реализующие этот алгоритм.

Язык Паскаль

```
{Два максимальных элемента массива}
const
n=5; {Число элементов}
aX:array[1..n] of integer=(10, 47, 23, 38, 60);
var i,Max1,Max2:integer;
begin
  if aX[1]>aX[2] then
    begin Max1:=aX[1]; Max2:=aX[2] end
  else begin Max1:=aX[2]; Max2:=aX[1] end;
  for i:=3 to n do
    if aX[i]>=Max1 then begin Max2:=Max1; Max1:=aX[i] end
    else if aX[i]>Max2 then Max2:=aX[i];
  writeln('Два максимальных элемента массива: ',Max1,', ',Max2);
end.
```

Язык Бейсик

```
'Два максимальных элемента массива
DEFINT A, I, M-N
n = 5 'Число элементов
DIM aX(1 TO n)
aX(1) = 47: aX(2) = 10: aX(3) = 23: aX(4) = 38: aX(5) = 60
IF aX(1) > aX(2) THEN
  Max1 = aX(1): Max2 = aX(2)
ELSE
  Max1 = aX(2): Max2 = aX(1)
END IF
FOR i = 3 TO n
IF aX(i) >= Max1 THEN
  Max2 = Max1: Max1 = aX(i)
ELSE
  IF aX(i) > Max2 THEN Max2 = aX(i)
END IF
NEXT i
PRINT "Два максимальных элемента массива: "; Max1; ", "; Max2
END
```

Язык Си

```
/*Два максимальных элемента массива*/
#include<stdio.h>
# define N 5 /*Число элементов*/
void main()
{int i,Max1,Max2,aX[N]={10, 47, 23, 38, 60};
  if (aX[0]>aX[1]) {Max1=aX[0]; Max2=aX[1];}
  else {Max1=aX[1]; Max2=aX[0];}
  for (i=2; i<N; i++)
    if (aX[i]>=Max1) {Max2=Max1; Max1=aX[i];}
    else if (aX[i]>Max2) Max2=aX[i];
  printf("\nДва максимальных элемента массива: %d, %d",Max1,Max2);
}
```

Тема: “СИМВОЛЬНЫЕ СТРОКИ”

Указание к задачам 11—15. Строки, являющиеся исходными данными, указываются непосредственно в программе.

Задача 11. Имеется строка. Добавить к ней в начале и в конце заданное число звездочек, т.е. символов “*”.

Решение. Пусть S — исходная строка, n — число звездочек. Сначала образуем строку Star, состоящую из n звездочек. Для этого сначала присвоим Star пустую строку “”, а затем n раз выполним сцепление строк Star и “*”. Теперь осталось сцепить строки Star, S, Star и результат присвоить S. В Паскале и Бейсике все это делается с помощью одного оператора. В Бейсике он записывается как

$$S = Star + S + Star$$

и аналогично в Паскале. В языке Си для выполнения указанных операций потребуется больше усилий. Дело в том, что в Си все действия со строками выполняются с помощью функций. В частности, функция strcat сцепляет две строки и результат присваивает первой из них, а функция strcpy копирует одну строку в другую. Поэтому первая из этих операций — получение копии строки Star, которую назовем StarR. Затем сцепляем строки Star и S, результатом чего становится строка Star. Ее копируем в S и сцепляем S и StarR, получая наконец исходную строку, обрамленную звездочками.

Приводим программы.

Язык Паскаль

```
{Обрамление строки звездочками}
var i,n:byte; S,Star:string;
begin
S:='Информатика';
write('Укажите число звездочек ');readln(n);
writeln('Исходная строка: ',S);
Star:='';
for i:=1 to n do Star:=Star+'*';
S:=Star+S+Star;
writeln('Строка, обрамленная звездочками: ',S);
end.
```

Язык Бейсик

```
'Обрамление строки звездочками
DIM i AS INTEGER, n AS INTEGER, S AS STRING, Star AS STRING
S = "Информатика"
INPUT "Укажите число звездочек "; n
PRINT "Исходная строка: "; S
Star = ""
FOR i = 1 TO n: Star = Star + "*": NEXT i
S = Star + S + Star
PRINT "Строка, обрамленная звездочками: "; S
END
```

Язык Си

```
/*Обрамление строки звездочками*/
#include<stdio.h>
#include<string.h>
void main()
{int i,n;
  char S[80]="Информатика",Star[80]="", StarR[80];
  printf("\nУкажите число звездочек "); scanf("%d",&n);
  printf("\nИсходная строка: %s",S);
  for (i=1; i<=n; i++) strcat(Star,"*");
  strcpy(StarR,Star);/*StarR — копия Star*/
  strcat(Star,S); /*Сцепление строк Star и S.
                    Результат присвоен Star */
  strcpy(S,Star); strcat(S,StarR);
  printf("\nСтрока, обрамленная звездочками: %s",S);
}
```

Задача 12. Строка представляет собой десятичную запись целого положительного числа, причем слева находится несколько “незначащих” нулей. Убрать левые незначащие нули. Например, строку “000480” заменить на строку “480”.

Решение. Рассмотрим алгоритмы решения этой задачи применительно к каждому языку программирования в отдельности.

а) В Паскале имеется процедура Delete, предназначенная для удаления подстроки. В частности, можно уничтожить подстроку единичной длины, т.е. символ. Использование указанной процедуры позволяет сформулировать весьма краткий алгоритм: пока первый символ заданной строки — ноль, уничтожать его.

б) В Бейсике можно сначала найти позицию первой слева цифры, отличной от нуля. Для этого начиная с первой позиции строки, пока символ в очередной позиции — ноль, переходим к следующей позиции. Подстрока, начинающаяся с найденной позиции вплоть до последней, и есть запись числа без левых нулей. Выделяем ее с помощью функции MID\$ и присваиваем первоначальной строке.

в) Напомним, что в Си индексация массивов начинается с нуля, а строка — это массив символов, содержащий признак конца строки. Прежде всего находим длину строки L с помощью функции strlen. Позиция (L-1) содержит последний символ строки, а позиция L — признак конца строки. Как и в предыдущем случае, находим позицию первой слева цифры, отличной от нуля, причем поиск начинаем с нулевой позиции. Обозначим ее i1. Теперь символы строки, начиная с позиции i1 и кончая позицией L, сдвигаем на i1 позиций влево. При этом символ, находившийся в позиции i1, окажется в нулевой позиции, следующий за ним — в первой и т.д. Другими словами, строка содержит запись числа без левых нулей.

Приводим соответствующие программы.

Язык Паскаль

```
{Убрать левые нули}
var
S:string; i:byte;
begin
  S:='000480';
  writeln('Исходная строка: ',S);
  i:=1;
  while S[i]='0' do delete(S,1,1);
  writeln('Строка без левых нулей: ',S);
end.
```

Язык Бейсик

```
'Убрать левые нули
DIM S AS STRING, i AS INTEGER
S = "000480"
PRINT "Исходная строка: "; S
i = 1
DO WHILE MID$(S, i, 1) = "0": i = i + 1: LOOP
S = MID$(S, i)
PRINT "Строка без левых нулей: "; S
END
```

Язык Си

```
/*Убрать левые нули*/
#include<stdio.h>
#include<string.h>
void main()
{int i,i1,L; char S[80]="000480";
  printf("\nИсходная строка: %s",S);
  L=strlen(S);
  i1=0; while (S[i1]=='0') i1++;
  for (i=i1; i<=L; i++) S[i-i1]=S[i];
  printf("\nСтрока без левых нулей: %s",S);
}
```

Задача 13. Строка представляет собой десятичную запись вещественного числа с ненулевой дробной частью, причем справа в конце строки находится несколько “незначащих” нулей. Убрать правые незначащие нули. Например, строку “12.35000” заменить на строку “12.35”.

Решение. Как и в предыдущей задаче, будем рассматривать алгоритмы решения этой задачи применительно к каждому языку программирования в отдельности.

а) Для Паскаля опять используем процедуру удаления подстроки Delete.

б) В Бейсике прежде всего находим длину строки. Затем ищем позицию первой справа цифры, отличной от нуля. Для этого начиная с последней позиции строки, пока символ в очередной позиции — ноль, переходим к предыдущей позиции. Подстрока, начинающаяся с первой позиции и заканчивающаяся в найденной позиции, есть запись числа без правых нулей. Выделяем ее с помощью функции LEFT\$ и присваиваем первоначальной строке.

Задачи по программированию

Продолжение. См. с. 11–13

в) В Си сначала действуем как и в Бейсике: находим длину строки, а затем ищем позицию первой справа цифры, отличной от нуля. Теперь в следующую позицию заносим признак конца строки — символ '\0'. Получаем запись числа без правых нулей.

Приводим программы.

Язык Паскаль

```
{Убрать правые нули}
var S:string; i:byte;
begin
  S:='12.35000';
  writeln('Исходная строка: ',S);
  while S[Length(S)]='0' do delete(S,Length(S),1);
  writeln('Строка без правых нулей: ',S);
end.
```

Язык Бейсик

```
'Убрать правые нули
DIM S AS STRING, i AS INTEGER
S = "12.35000"
PRINT "Исходная строка: "; S
i = LEN(S)
DO WHILE MID$(S, i, 1) = "0": i = i - 1: LOOP
S = LEFT$(S, i)
PRINT "Строка без правых нулей: "; S
END
```

Язык Си

```
/*Убрать правые нули*/
#include<stdio.h>
#include<string.h>
void main()
{int i; char S[80]="12.35000";
 printf("\nИсходная строка: %s",S);
 i = strlen(S)-1;
 while (S[i]!='0') i--;
 S[i+1]='\0';
 printf("\nСтрока без правых нулей: %s",S);
}
```

Задача 14. Строка содержит арифметическое выражение, в котором используются круглые скобки. Проверить “правильность” расстановки скобок: если имеются лишние правые (закрывающие) скобки “)”, то выдать сообщение с указанием позиции первой такой скобки; если имеются лишние левые (открывающие) скобки “(”, то выдать сообщение с указанием количества таких скобок; если скобки расставлены правильно, то сообщить об этом.

Замечание: такая проверка не гарантирует, что выражение с “правильно” расставленными скобками действительно является правильным. Последний пример демонстрирует это.

ПРИМЕРЫ

Скобочное выражение

```
(a+b)/(c*d(
(a+b)/(c*d(
(a+b)/(c*d)
(a+b(/)c*d)
```

Сообщение

```
Лишняя правая скобка в позиции 7
Число лишних левых скобок – 2
Скобки расставлены правильно
Скобки расставлены правильно
```

Решение. Будем использовать *счетчик*, предварительно его занулив. Просматриваем строку слева направо начиная с первой (для Си — нулевой) позиции. Встретив левую (открывающую) скобку “(”, прибавляем к счетчику единицу. Встретив правую (закрывающую) скобку “)”, отнимаем от счетчика единицу. Просмотр заканчиваем, если счетчик имеет отрицательное значение или символ — последний. По окончании просмотра анализируем значение счетчика. Если он отрицательный, то выдаем сообщение о лишней правой (закрывающей) скобке. Она находится в последней просмотренной позиции. Если счетчик имеет положительное значение, то выдаем сообщение о лишних левых (открывающих) скобках. Их число определяется значением счетчика. Наконец, если счетчик равен нулю, то скобки расставлены правильно.

Приводим программы.

Язык Паскаль

```
{Расстановка круглых скобок}
var i:byte; Count:integer; S:string;
begin
  S:='(a+b)/(c*d(';
  writeln('Скобочное выражение: ',S);
  Count:=0; i:=0;
  repeat
    i:=i+1;
    if S[i]='(' then Count:=Count+1; if S[i]=')' then Count:=Count-1;
  until (Count<0) or (i=Length(S));
  if Count<0 then writeln('Лишняя правая скобка в позиции ',i);
  if Count=0 then writeln('Скобки расставлены правильно');
  if Count>0 then writeln('Число лишних левых скобок ',Count);
end.
```

Язык Бейсик

```
'Расстановка круглых скобок
DIM i AS INTEGER, Count AS INTEGER, S AS STRING
S = "(a+b)/(c*d("
PRINT "Скобочное выражение: "; S
Count = 0: i = 0
DO
  i = i + 1
  IF MID$(S, i, 1) = "(" THEN Count = Count + 1
  IF MID$(S, i, 1) = ")" THEN Count = Count - 1
LOOP UNTIL Count < 0 OR i = LEN(S)
IF Count < 0 THEN PRINT "Лишняя правая скобка в позиции "; i
IF Count = 0 THEN PRINT "Скобки расставлены правильно"
IF Count > 0 THEN PRINT "Число лишних левых скобок "; Count
END
```

Язык Си

```
/*Расстановка круглых скобок*/
#include<stdio.h>
#include<string.h>
void main()
{int i=-1,Count=0; char S[80]="(a+b)/(c*d(";
 printf("\nСкобочное выражение: %s",S);
 do
  {i++;
   if ( S[i]=='(' ) Count++; if ( S[i]==')' ) Count--;
  }
  while (Count>=0 && i<strlen(S)-1);
  if (Count<0) printf("\nЛишняя правая скобка в позиции %d",i+1);
  if (Count==0) printf("\nСкобки расставлены правильно");
  if (Count>0) printf("\nЧисло лишних левых скобок %d",Count);
 }
```

Задача 15*. Назовем *разделителями* символы: пробел “ ”, запятую “,”, точку с запятой “;”, двоеточие “:”. Слово определим как группу символов, стоящих в строке подряд и среди которых нет разделителей. Тогда любая строка состоит из слов, между которыми находятся один или несколько разделителей. Определить число слов в заданной строке.

Указание. Объединить разделители в строку. Для определения, является ли некоторый символ разделителем, использовать функции, проверяющие вхождение подстроки или символа в строку. В Паскале это Pos, в Бейсике — INSTR, в Си — strchr.

Решение. Следуя указанию, организуем строку разделителей. Тогда символ является разделителем, если он входит в эту строку, что проверяют перечисленные функции. Введем счетчик числа слов, предварительно его занулив. Будем просматривать строку слева направо и искать ситуации, соответствующие концу очередного слова. Ясно, что такая ситуация соответствует случаю, когда очередной символ — разделитель, а его предшественник к таковым не относится. Зафиксировав эту ситуацию, наращиваем счетчик слов. Закончив этот процесс, мы найдем все слова, кроме, возможно, последнего. Это произойдет, если конец последнего слова совпадает с концом строки. Поэтому, окончив просмотр, надо проверить, является ли последний символ строки разделителем или нет. В последнем случае к счетчику слов добавляем единицу.

Приводим программы, реализующие описанный алгоритм.

Язык Паскаль

```
{Число слов в строке}
var i,Nword,L:byte; S:string;
Const Sep=' ,;:';
begin
  S:='aa bb,cc;dd:ee';
  Nword:=0; L:=Length(S);
  for i:=2 to L do
    if (Pos(S[i],Sep)<>0) and (Pos(S[i-1],Sep)=0) then
      Nword:=Nword+1;
  if (Pos(S[L],Sep)=0) then Nword:=Nword+1;
  writeln('Строка: ',S);
  writeln('Число слов в строке: ',Nword);
end.
```

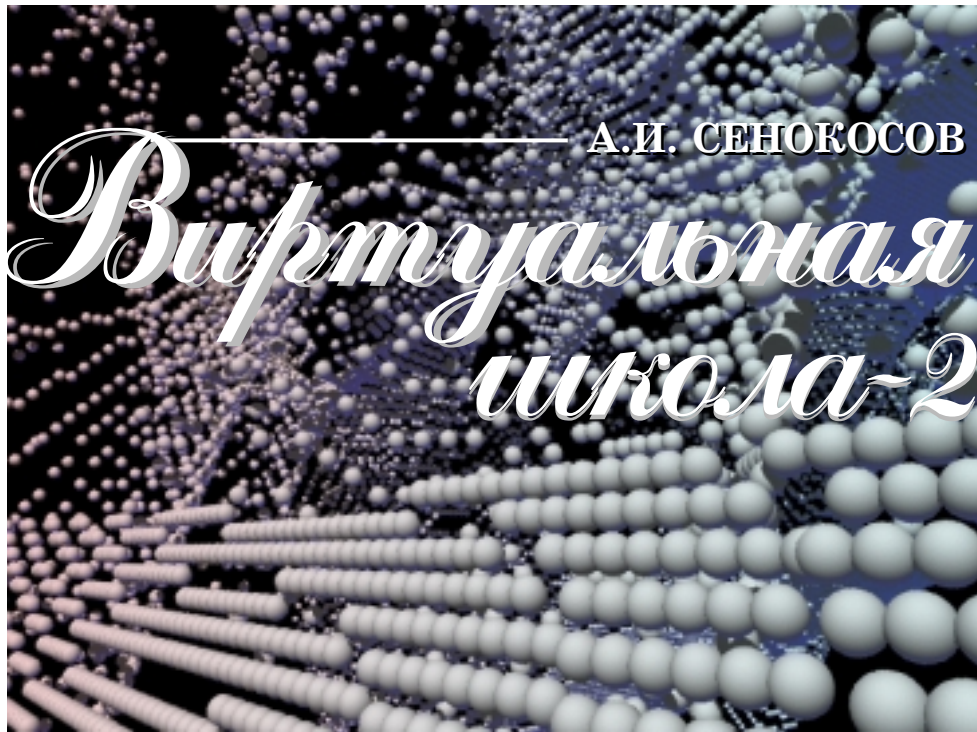
Язык Бейсик

```
'Число слов в строке
DIM i AS INTEGER, Nword AS INTEGER, L AS INTEGER
DIM C AS STRING, C1 AS STRING
DIM Sep AS STRING: Sep = " ,;:"
DIM S AS STRING: S = "aa bb,cc;dd:ee"
Nword = 0: L = LEN(S)
FOR i = 2 TO L
  C = MID$(S, i, 1): C1 = MID$(S, i - 1, 1)
  IF INSTR(Sep, C) <> 0 AND INSTR(Sep, C1) = 0 THEN
    Nword = Nword + 1
  END IF
NEXT i
C = MID$(S, L, 1)
IF INSTR(Sep, C) = 0 THEN Nword = Nword + 1
PRINT "Строка: "; S
PRINT "Число слов в строке: "; Nword
END
```

Язык Си

```
/*Число слов в строке*/
#include<stdio.h>
#include<string.h>
void main()
{int i,Nword=0,L;
 char Sep[5]=" ,;:", S[80]="aa bb,cc;dd:ee";
 L=strlen(S);
 for (i=1; i<L; i++)
  if (strchr(Sep,S[i])!= NULL && strchr(Sep,S[i-1])==NULL)
    Nword++;
 if (strchr(Sep,S[L-1])==NULL) Nword++;
 printf("\nСтрока: %s",S);
 printf("\nЧисло слов в строке: %d",Nword);
}
```

14



Окончание. Начало в № 18/99

Напомним вкратце, о чем шла речь в "Многоотрудной истории..." (см. № 3, 5, 6, 9, 11/98). В школе № 104 г. Екатеринбург создана локальная сеть, а в большинстве предметных кабинетов стоят мультимедийные комплексы на базе компьютера, TV-кодера и телевизора.

Поначалу комплексы использовались в трех вариантах:

1. (Самый ранний и самый простой.) Просмотр слайдов из общешкольной коллекции, насчитывающей их несколько тысяч. Коллекция постоянно пополняется за счет сканирования новых иллюстраций.
2. Использование текстового редактора IBM WORKS или графического CorelDRAW! как "электронной доски" с заранее подготовленными текстами и иллюстрациями, которые можно совершенно свободно передвигать по полю "электронной доски" во время урока.
3. Использование на уроке специализированных обучающих программ, рассчитанных именно на вариант мультимедиа-кабинета в нашем понимании (т.е. один компьютер с демонстрационным телевизором в классе).
4. ...

Вот как раз про четвертый вариант в "Многоотрудной истории..." не было рассказано ничего. Постараемся хотя бы отчасти восполнить этот недостающий фрагмент.

Было бы смешно, имея общешкольную компьютерную сеть, не попробовать организовать в ней публикацию HTML-документов хотя бы в рамках изучения сетевых технологий в курсе информатики.

Собственно говоря, "Виртуальная школа" создавалась исключительно как учебное средство по теме "Гипертекстовый редактор". Ею предполагалось заменить лабораторные работы для 7-го класса, использующие гипертекстовый редактор "LinkWay" (см. учебник "Информатика-7" — "Информатика" № 33/97).

Школа не имеет выхода в Интернет, изучать CGI-программирование изначально никто не собирался, поэтому "Виртуальная школа" использует протокол NetBEUI без установки Интернет-сервера. Говоря более простым языком, если у вас уже есть локальная сеть, достаточно указать программе-просмотрщику (Microsoft Internet Explorer или Netscape Navigator) в качестве основной странички HTML-файл, расположенный на сервере, чтобы система web-публикации заработала.

Главный HTML-файл содержал поначалу лишь ссылки на странички классов, а странички классов, в свою очередь, содержали ссылки на странички учеников. Кроме того, исключительно для оживления ра-

боты на сервер была скопирована примерно тысяча картинок на самые различные темы: "Фантастика", "Животные", "Пейзажи", "Фоновые рисунки", "Машины", "Виды города" и т.п.

Согласитесь, было бы несправедливо изучать этот материал только с седьмыми классами, обойдя вниманием старшие лишь потому, что в их "семиклассное" время никакой школьной сети не существовало.

Поэтому были изысканы необходимые резервы учебных часов в рамках учебной программы для обучения технологии web-публикации всех классов.

Это дало начало процессу, который иначе как тихим массовым помешательством назвать нельзя. В дополнительное компьютерное время, которое всегда существовало в нашей школе, кабинет буквально ломился от желающих поработать, чего раньше никогда не было. Как легко понять, все занимались созданием собственных страничек, насыщенных картинками, фотографиями, текстами любимых песен, стихами...

Конечно, информатика в нашей школе и раньше-то была предметом, который не требовал постоянной "апелляции к мотивации" (т.е. заклинаний типа "Учите, а не то...!!!"). С появлением же "Виртуальной школы" ситуация существенно изменилась. И не просто в лучшую сторону. Стало возможно говорить о принципиально новом уровне в изучении предмета.

Во-первых, информатика оказалась чуть ли не единственным школьным предметом, в котором шаблонное мышление оказалось не только неприемлемым, но и общественно-осуждаемым. Попробуйте-ка создать свою страничку, почти не отличающуюся от страничек одноклассников, и приготовьтесь выслушать все, что они об этом думают. Если списывание на других предметах воспринимается, скажем так, не всегда резко негативно, то на информатике оно становится просто невозможным.

В общем-то это дело психологов — разобраться в неожиданном и буквально взрывном интересе к информатике при появлении "Виртуальной школы", но вполне разумным объяснением можно считать резко возросшую возможность самовыражения, что фактически эквивалентно расширению области приложения творческого начала.

Конечно, можно считать творческое начало уделом избранных. Ребенок, скажем, рисуя домик, самовыражается, а Микеланджело, рисуя Мадонну, творит. Или, скажем, Пушкин творил свою поэзию, автор же справочного руководства по Windows опять же всего лишь самовыражался. Представляется достаточно бессмысленным спорить с такой точкой зрения, поэтому в качестве рабочей модели,

вполне адекватно описывающей происходящие события, примем следующее объяснение:

Успех "Виртуальной школы" базируется на врожденной тяге любого человека к творчеству.

Имея локальную сеть и компьютер почти в каждом кабинете, было очень несложно направить бьющую через край творческую энергию школьников на создание HTML-страничек по заказам учителей.

Разумеется, учителям пришлось объяснить, что такое гипертекст, как его проектировать, а затем использовать в качестве наглядного материала на уроках. И на это потребовалось чуть ли не полчаса времени. По общему мнению, гипертекст оказался тем самым долгожданным визуальным, внешне привлекательным и эстетически состоятельным средством организации урока, которого до сих пор так не хватало.

Иллюстрации оказалось возможно снабжать не просто подписями, но и развернутыми комментариями, а тестовые задания — ответами. Добавим к этому очень большую гибкость в подаче компьютерных слайдов, возможность включать непосредственно в гипертекст файлы, подготовленные в редакторах Microsoft Word и CorelDRAW! с помощью механизма ActiveX, и вы получите чуть ли не идеальную систему организации компьютерного иллюстративно-текстового, видео- и звукового материала для различных уроков.

В отличие от специализированных обучающих программ, которые по плечу далеко не каждому ученику, создание гипертекстовых HTML-страниц вкуче с несложными JavaScript-программами доступно вообще любому. Мало того, элементы JavaScript-оформления достаточно прозрачны, и относительно сложные программы, обеспечивающие, например, анимационные эффекты появления слайдов, можно напрямую копировать, изменяя лишь их отдельные строчки.

Вернемся к делам нашим школьным. К концу учебного года намечилось явное нежелание учащихся расставаться с любимой игрушкой в виде "Виртуальной школы". Справедливости ради заметим, что она стала не только игрушкой для учащихся, но и чуть ли не самым необходимым рабочим инструментом многих преподавателей.

Поэтому вполне логичной представлялась идея "вечного" продления этого удовольствия, разработав курс информатики

и, соответственно, учебник, ориентированный именно на обслуживание "Виртуальной школы".

Такой подход, как легко видеть из всего предыдущего рассуждения, позволяет в самой обычной школе в рамках самого обычного курса решить все задачи, стоящие перед школьной информатикой, кроме, разумеется, подготовки профессионалов в области ПОД, что является прерогативой углубленного курса.

Интернет-технология, таким образом, выступает не только как средство создания школьной Единой Информационной Среды, но и в качестве чрезвычайно гибкого инструмента, обеспечивающего, помимо всего прочего, и общеразвивающую функцию информатики. Безусловно, при этом в полной мере реализуются и требования "Образовательного минимума".

Феномен "Виртуальной школы" в чем-то сродни феномену персонального компьютера. За счет использования очень простых, доступных любому ученику средств самими школьниками создается до предела визуализированная красочная учебно-игровая среда, что производит буквально революционный эффект в восприятии предмета "информатика" как такового и его места в школе.

Решение же "пятой задачи А.Ю. Уварова" (см. первую часть статьи) методами "Виртуальной школы" отличается от решения методом разработки обучающих программ на языках программирования, о котором часто говорят и пишут, в том числе и в нашей газете, абсолютно адекватно различию между мини-компьютерами и современными им первыми персональными.

"Если мини-компьютеры были созданы для профессионалов, то уже самые первые персональные компьютеры ориентировались на самого обычного человека. Конечно, поначалу никто не воспринимал всерьез "персоналки", и крупнейшие фирмы-производители продолжали выпускать мини и супермини для профессионалов. Чем все это кончилось, мы прекрасно знаем". (Впрочем, желающие могут прочитать об этом в книге Г.Р. Громова. (Громов Г.Р. Национальные информационные ресурсы: проблемы промышленной эксплуатации. М.: Наука, 1985).)

Окончание на с. 16

1999 № 19 ИНФОРМАТИКА

ПРОВЕРЬТЕ ПРАВИЛЬНОСТЬ ОФОРМЛЕНИЯ АБОНЕМЕНТА!

На абонементах должен быть проставлен отпечаток кассовой машины.

При оформлении подписки (переадресовки) без кассовой машины на абонементах проставляется отпечаток календарного штемпеля отделения связи. В этом случае выдается подписчику с квитанцией об уплате стоимости подписки (переадресовки).

Для оформления подписки на газету или журнал, а также для переадресования издания бланк абонемента с доставочной карточкой заполняется подписчиком чернилами, разборчиво, без сокращений, в соответствии с условиями, изложенными в каталогах Роспечати.

Заполнение месячных клеток при переадресовании издания, а также клетки "ПВ—МЕСТО" производится работниками предприятий связи и Роспечати.

ВОКРУГ ПРЕДМЕТА

Обратившись к школьным проблемам, можно заметить, что скоро исполнится 20 лет попыткам внедрить компьютер в другие предметы с помощью полупрофессиональных программ, написанных школьниками (опять же полупрофессионалами) по заказам учителей. И могут ли даже самые "продвинутые" в этом направлении школы похвастаться тем, что хотя бы треть уроков по предметам ведется с использованием компьютеров? В Екатеринбургской же школе № 104 уже сейчас 10—15% всех уроков проводится с использованием мультимедийных комплексов, а в ближайшие год-два планируется довести это число до 50—70%. И эти планы более чем реальны.

Напомним при этом, что по вполне корректным расчетам эффективность каждого такого урока на 15—20% выше обычного. Видимо, в первом приближении это можно считать очень неплохим результатом при решении задачи интенсификации учебного процесса.

Заметим, что эта интенсификация идет отнюдь не за счет увеличения физических и умственных нагрузок на школьников или учителей. Напротив, серьезное повышение уровня наглядности преподавания приводит к заинтересованности и включению эмоционального сопричастия. Показательно, что ученики с нетерпением ждут мультимедийных уроков, а недавнее "затопление" школы талой водой с крыши стало всеобщей трагедией еще и потому, что привело к отключению комплексов в кабинетах математики и истории.

Учителя, разумеется, поначалу вынуждены тратить довольно много времени на подготовку "электронных досок", разработку эскизов HTML-страниц, но, во-первых, Единая Информационная Среда сразу делает каждую разработку всеобщим достоянием, а во-вторых, уже на второй-третий годы работы с таким комплексом происходит значительное облегчение нелегкого учительского труда за счет наработанных ранее материала-

лов. Впрочем, чисто психологически довольно легко затратить изрядное время на подготовку какой-либо темы, понимая, что

- 1) это делается раз и навсегда;
- 2) материал усваивается учащимися значительно лучше;
- 3) на уроке практически полностью отпадает вопрос дисциплины.

Тут уже начинает играть роль феномен не школьников, а учителей. Те, кому довелось хоть раз в своей учительской жизни увидеть горящие глаза детей, ловящих каждое твое слово, поймут, что ради этого не жалко никаких дополнительных временных затрат.

Пора подводить хотя бы первоначальные итоги.

Несмотря на умеренный скептицизм уже упомянутой статьи А.Ю. Уварова, объединение 1-, 2-, 4- и 5-й задач представляется вполне реальным без всякого головоломного дележа зон ответственности между предметами естественно-научного цикла и, соответственно, без пересмотра программ и безумно дорогостоящей разработки новых учебников.

Для этого даже не надо пересматривать "Обязательный минимум" по информатике, но совершенно необходимо разработать новый курс и новый учебник, ориентированные на создание Единой Школьной Информационной Среды.

Разумеется, требуются и некоторые чисто технические предпосылки в виде кабинета информатики, оснащенного компьютерами хотя бы на базе 486-го процессора, локальная школьная сеть и хотя бы один предметный кабинет с мультимедийным комплексом, подключенным к такой сети.

Наш опыт показывает, что это не такая уж и утопическая задача. Чтобы не быть голословными, в летних номерах "Информатики" вашему вниманию будет предложен учебник для седьмого класса "Информатика в виртуальной школе-7".

Три пары монет

Условие задачи опубликовано на с. 2

Решение

Хорошие монеты тяжелее фальшивых. Поэтому назовем их тяжелыми, а фальшивые — легкими. Золотые монеты (и их веса) обозначим (a_1, a_2) , серебряные — (b_1, b_2) , бронзовые — (c_1, c_2) .

На первом взвешивании сравним

$$(a_1 + b_1) ? (a_2 + c_2).$$

1. Если окажется равенство, т.е.

$$a_1 + b_1 = a_2 + c_2,$$

то все просто: вместе с тяжелой из монет (a_1, a_2) лежит легкая из (b_1, c_2) , а вместе с легкой — тяжелая. Сравнив вторым взвешиванием a_1 и a_2 , мы все и выясним.

2. Сложнее, если первое взвешивание покажет неравенство. Например,

$$a_1 + b_1 < a_2 + c_2.$$

Тогда заметим, что $a_1 < a_2$ и с парой (a_1, a_2) мы разобрались.

Вторым взвешиванием сравним

$$(a_1 + a_2) ? (b_1 + c_2).$$

Если правая чашка перевесит, то обе монеты (b_1, c_2) — тяжелые;

Если перевесит левая — то обе они легкие. Если же выйдет равенство:

$$a_1 + a_2 = b_1 + c_2,$$

то одна из монет (b_1, c_2) легкая, а другая — тяжелая. Значит, легкая — b_1 , а тяжелая — c_2 (если наоборот, то в первом взвешивании было бы равенство).

3. Остается случай, когда первое взвешивание показывает, что $a_1 + b_1 > a_2 + c_2$. Его можно рассмотреть так же, как предыдущее. Не делайте этого! Проще переменить обозначения a_1, b_1, c_1 на a_2, c_2, b_2 (соответственно), и этот случай сведется к предыдущему.

Сводка результатов:

1. Если $a_1 + b_1 = a_2 + c_2$, то: если $a_1 < a_2$, то хорошие a_2, b_1, c_1 ; если $a_1 > a_2$, то хорошие a_1, b_2, c_2 .
2. Если $a_1 + b_1 < a_2 + c_2$, то: если $a_1 + a_2 < b_1 + c_2$, то хорошие a_2, b_1, c_2 ; если $a_1 + a_2 = b_1 + c_2$, то хорошие a_2, b_2, c_2 ; если $a_1 + a_2 > b_1 + c_2$, то хорошие a_2, b_2, c_1 .
3. Если $a_1 + b_1 > a_2 + c_2$, то: если $a_1 + a_2 > b_1 + c_2$, то хорошие a_1, b_2, c_1 ; если $a_1 + a_2 = b_1 + c_2$, то хорошие a_1, b_1, c_1 ; если $a_1 + a_2 < b_1 + c_2$, то хорошие a_1, b_1, c_2 .

16

ЗАДАЧА

1999 № 19 ИНФОРМАТИКА

©ИНФОРМАТИКА 1999
выходит четыре раза в месяц
При перепечатке ссылка
на ИНФОРМАТИКУ
обязательна, рукописи
не возвращаются.
Регистрационный номер 012868

121165, Москва,
Киевская, 24
тел. 249 4896
Отдел рекламы
тел. 249 9870

ИНДЕКС ПОДПИСКИ
для индивидуальных подписчиков 32291
для предприятий и организаций 32591
комплекта приложений 32744

Internet: inf@1september.ru
Fidonet: 2:5020/69.32
WWW: http://www.1september.ru



тел./факс (095)249 3138, факс (095)249 3184, тел. 249 3386

ОБЪЕДИНЕНИЕ ПЕДАГОГИЧЕСКИХ ИЗДАНИЙ "ПЕРВОЕ СЕНТЯБРЯ"

Первое сентября
А.С. Соловейчик
индекс подписки — 32024

Английский язык
Е.В. Громушкина
индекс подписки — 32025

Биология
Н.Г. Иванова
индекс подписки — 32026

Воскресная школа
монах Киприан (Яценко)
индекс подписки — 32742

География
О.Н. Коротова
индекс подписки — 32027

Здоровье детей
А.У. Лекманов
индекс подписки — 32033

Информатика
С.Л. Островский
индекс подписки — 32291

Искусство
Н.Х. Исмаилова
индекс подписки — 32584

История
А.Ю. Головатенко
индекс подписки — 32028

Литература
Г.Г. Красухин
индекс подписки — 32029

Математика
И.Л. Соловейчик
индекс подписки — 32030

Начальная школа
М.В. Соловейчик
индекс подписки — 32031

Немецкий язык
Gerolf Demmel
индекс подписки — 32292

Русский язык
Л.А. Гончар
индекс подписки — 32383

Спорт в школе
Н.В. Школьникова
индекс подписки — 32384

Управление школой
Н.А. Широкова
индекс подписки — 32652

Физика
Н.Д. Козлова
индекс подписки — 32032

Химия
О.Г. Блохина
индекс подписки — 32034

Школьный психолог
М.Н. Сартан
индекс подписки — 32898

Гл. редактор
С.Л. Островский
Зам. гл. редактора
Е.Б. Докшицкая

Редакция:
Л.Н. Картвелишвили,
Ю.А. Соколинский,
Н.Л. Беленькая,
Н.П. Медведева
Дизайн и компьютерная верстка:
Н.И. Пронская
Корректоры:
Е.Л. Володина,
С.М. Подберезина

Отпечатано с готовых диапозитивов редакции в ОАО ПО «Пресса-1», 125865, ГСП, Москва, ул. Правды, 24.

Тираж 7000 экз.
Заказ №

Ф. СП-1		Министерство связи Российской Федерации "Роспечать"	
АБОНЕМЕНТ на газету		32291	
Информатика		(индекс издания)	
наименование издания		Количество комплектов	
на 1999 год по месяцам			
1	2	3	4
5	6	7	8
9	10	11	12
Куда		(почтовый индекс) (адрес)	
Кому		(фамилия, инициалы)	
ДОСТАВочНАЯ КАРТОЧКА			
ПВ	место	ли-тер	на газету 32291
			(индекс издания)
Информатика			
(наименование издания)			
Стои-мость	подписки	_____ руб.	Количество комплек-тов
	пере-адресовки	_____ руб.	
на 1999 год по месяцам			
1	2	3	4
5	6	7	8
9	10	11	12
Куда		(почтовый индекс) (адрес)	
Кому			