

Предлагаем подружиться

(Игры с программами)

Самыми распространенными и доступными в СССР индивидуальными средствами вычислительной техники сегодня являются микрокалькуляторы (ПМК) моделей «Электроника «БЗ-34», «МК-54», «МК-56», «МК-61», «МК-52». Они различаются некоторыми обозначениями на клавишах (см. табл. 1).

Несмотря на некоторые недостатки, ПМК достаточно хорошо зарекомендовали себя при решении многих задач. В частности, они позволяют моделировать некоторые типичные игровые ситуации, и такое их применение многих интересует.

Вычислительные возможности ПМК ограничены: 14—15 регистров для запоминания чисел и 98—105 шагов программной памяти для записи команд. Однако некоторые особенности этих устройств — «польская» (обратная бесскобочная) запись арифметических действий, большое количество встроенных функций (в том числе тригонометрические, показательные и обратные им), возможность косвенной адресации по любому регистру, наличие команд циклов, подпрограмм и переходов, довольно удобный оперативный стек на 4 числа — позволяют создавать по-настоящему сложные программы. Возможности калькуляторов «МК-61» и «МК-52» расширены за счет введения весьма полезных для обработки информации логических операций и некоторых новых функций (модуль и знак числа, выделение целой и дробной частей, встроенный генератор псевдослучайных чисел и так далее).

БЗ-34	МК-54 МК-56 МК-52 МК-61	В книге
ИП	$\Pi \rightarrow x$	ИП
$\frac{\Pi}{xy}$	$x \rightarrow \Pi$	Π
$\frac{\uparrow}{\text{ШГ}}$	\longleftrightarrow	\longleftrightarrow
$\frac{\uparrow}{\text{ШГ}}$	$\text{В} \uparrow$	\uparrow
$\frac{\div}{\text{ШГ}}$	$\frac{\div}{\text{ШГ}}$	$\frac{\div}{\text{ШГ}}$ вправо
$\frac{\div}{\text{ШГ}}$	$\frac{\div}{\text{ШГ}}$	$\frac{\div}{\text{ШГ}}$ влево
$\frac{\div}{\text{FO}}$	$\frac{\div}{\text{FO}}$	$\frac{\div}{\text{FO}}$
$\frac{\div}{\text{Farc sin}}$	$\frac{\div}{\text{sin}^{-1}}$	$\frac{\div}{\text{Farc sin}}$
$\frac{\div}{\text{Farc cos}}$	$\frac{\div}{\text{cos}^{-1}}$	$\frac{\div}{\text{Farc cos}}$
$\frac{\div}{\text{Farc tg}}$	$\frac{\div}{\text{tg}^{-1}}$	$\frac{\div}{\text{Farc tg}}$

Таблица 1.

Нередко ограниченные возможности ПМК заставляют программистов идти на всяческие ухищрения, чтобы «втиснуть» выбранный алгоритм в более чем скромную память машины. Он должен в совершенстве владеть полным арсеналом не только стандартных, но и искусственных приемов, таких, как использование кодов команд в качестве адресов переходов, работа со «сверхчислами» (числа, порядок которых превышает 100), нестандартное применение команд, связанных с образованием чисел (десятичная точка и ввод порядка), манипуляции переключателем углов (градусы — градусы — радианы) для ввода информации с пульта во время работы по программе, использование операторов цикла вместо команд сравнения и так далее. Разумеется, он еще и обязан преобразовывать алгоритм к максимально лаконичному виду. Такая работа дает большое удовлетворение, а ее результат повергает в изумление тех, кто презрительно относится к «игрушечному компьютеру».

Игры с помощником

Что вы знаете и чего не знаете о микрокалькуляторе

Перед вами микрокалькулятор. Тридцать разноцветных клавиш выстроились по пять в линию, на каждой — обозначение, над клавишами — тоже, а в нижнем ряду — и под клавишами.

Некоторые обозначения понятны сразу — это цифры, символ «» для отделения целой части от дробной, знаки арифметических операций, названия элементарных функций. Другие — менее очевидны, о них еще будем говорить.

Над рядами клавиш — два переключателя. Левый включает калькулятор, а правый задает режим работы с тригонометрическими функциями: Р означает радианы, Г — градусы, промежуточная позиция ГРД (на «БЗ-34» она, правда, не обозначена) — градусы, сотые доли прямого угла. На вычислении остальных функций положение этого переключателя не отражается.

Передвигаем левый переключатель вправо. На индикаторе загорается цифра 0. Микрокалькулятор готов к работе.

Надо сказать, что сигналы от нажимаемых клавиш по-разному воспринимаются микрокалькулятором в зависимости от того, в каком состоянии он находится, а их два. В первом выполняются команды, либо отдаваемые непосредственно нажатием на клавиши, либо записанные в память микрокалькулятора в виде программы. Программы вводятся в нашу миниатюрную ЭВМ нажатием на те же клавиши, когда она находится во втором состоянии.

Включая микрокалькулятор, мы автоматически устанавливаем его в первое состояние, при этом его память совершенно чиста, так что сразу после включения можно вести вычисления лишь в непрограммном режиме. Ими и займемся сначала, что поможет понять принципы работы машины, назначение клавиш и значительно облегчит переход к программированию.

Освоим клавиатуру. Если нажать на цифровые клавиши, на экране загораются те же цифры: 1, 2, ..., 7, 8. Цифра 9 не появляется, так как калькулятор рассчитан на работу не более чем с восьмиразрядными числами, поэтому дальнейший ввод блокируется.

Чтобы занести дробное число, нажимаем после цифр целой части клавишу «.» и затем вводим дробную часть. Интересно, что сам символ «.» располагается в той же позиции, что и последняя введенная перед ним цифра. Так что при употреблении десятичной запятой общее число позиций для записи значащих цифр не меняется. Ради тренировки введем в наш калькулятор числа 3,1415926 (отношение длины окружности к диаметру), затем 2,7182818 (основание натуральных логарифмов).

Нетрудно сообразить, что возможности такой записи чисел ограничены. Нельзя, к примеру, таким образом ввести в микрокалькулятор заряд электрона, измеренный в кулонах; он выражается десятичной дробью с девятнадцатью нулями после запятой. И расстояние от Земли до Солнца таким способом в калькулятор не внести: выраженное в километрах, оно содержит девять знаков, а в метрах — еще больше.

Подобные затруднения преодолимы, если записывать числа в так называемом экспоненциальном виде $a \cdot 10^b$, где число a должно начинаться с ненулевой цифры, после которой стоит запятая. Число a называется мантиссой, b — порядком.

Вот, например, как записывается в экспоненциальной форме заряд электрона: $1,6021892 \cdot 10^{-19}$ Кл. А вот как — среднее расстояние от Земли до Солнца: $1,497 \cdot 10^8$ км.

Чтобы ввести в микрокалькулятор число, записанное в экспоненциальном виде, надо сначала набрать мантиссу. Если число отрицательное, вслед за этим следует нажать клавишу «/-/». Затем, чтобы указать порядок числа, надо нажать клавишу «ВП» (ввод порядка). В правой части индикатора тотчас появятся два нуля. Теперь нажатие на любую цифровую клавишу приводит к ее отображению в правом углу индикатора. Набираем порядок, и если он отрицательный, то нажимаем клавишу «/-/».

Диапазон чисел, работа с которыми доступна микрокалькулятору, огромен: максимальное — $9,9999999 \cdot 10^{99}$, минимальное по абсолютной величине — 10^{-99} .

Если при задании порядка числа попытаться ввести более двух цифр, то каждая новая будет занимать правую позицию, бывшая там ранее цифра сместится влево, а стоявшая слева исчезнет.

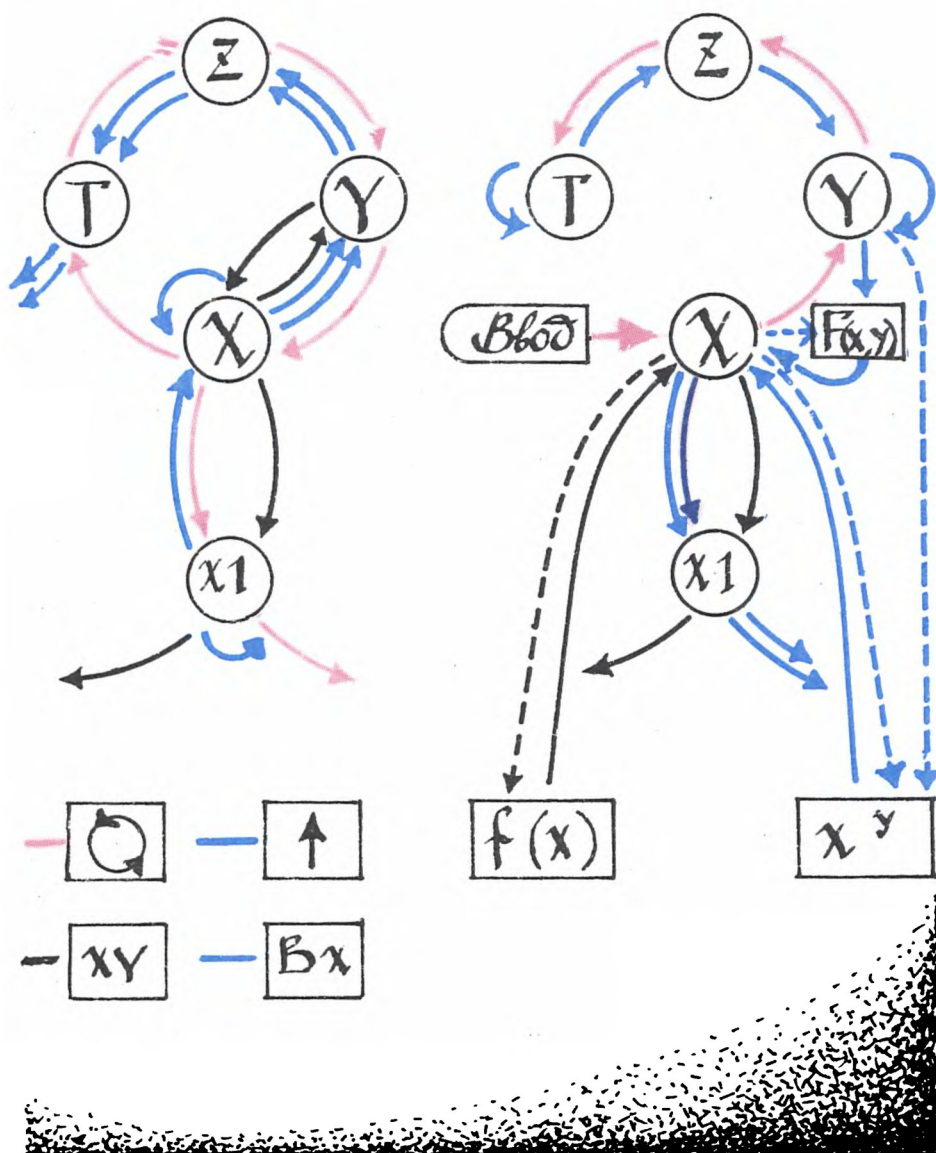
Если вы ошиблись при вводе числа — не беда: клавиша «Сх» «очищает» индикатор. Но дело тут не только в том, что гаснут горевшие на нем цифры. Любое число, введенное в калькулятор с клавиатуры, помещается в ячейку памяти, называемую

«регистром X». Содержимое этого регистра отображается на индикаторе. Действие клавиш и «Сх» как раз в том и заключается, что она очищает регистр X (отсюда и ее название : стереть x). На индикаторе отражается лишь результат, то есть цифра 0, после чего ввод можно повторить.

Теперь перейдем к вычислениям. Надо сказать, что порядок вычислений на нашем калькуляторе отличается от общепринятого.

Записывая любую арифметическую операцию, ставим знак ее между «участниками» операции (их называют операндами). Читая любое арифметическое выражение, надо помнить о приоритете операций: сначала действия в скобках, затем умножение и деление, потом сложение и вычитание. Таким образом, удастся однозначно толковать любую запись. К примеру, вычисляя $a \times b + c \times d$, сначала выполняем два умножения, а затем складываем полученные произведения. Если нужно изменить порядок вычислений, то ставят скобки, например, так: $a \times (b + c \times d)$.

Однако способ этот хоть и удобен для нас, но неэкономичен для микрокалькулятора. Ведь надо где-то хранить промежуточные результаты, запоминать знаки предыдущих операций, проверять их приоритеты. От этих недостатков свободен другой метод записи арифметических выражений, предложенный польским логиком Я. Лукасевичем и получивший название обратной бесскобочной или польской записи. Все операции при такой записи равноценны, а знаки их ставятся не между операндами, а после них, то есть не $a \times b$, а $ab \times$, что избавляет от необходимости использо-



вать скобки и проверять приоритеты. При польской записи первое из приведенных выражений запишется так: $ab \times cd \times +$, а второе — так: $cd \times b + a \times$.

Каждая арифметическая операция для выполнения требует двух исходных чисел и потому называется двуместной.

Для размещения участников двуместной операции нужно два различных регистра, микрокалькулятор предоставляет такую возможность. Регистр X, куда вводятся числа с клавиатуры, — лишь один из четырех взаимосвязанных регистров, обозначенных буквами X, Y, Z, T и образующих вместе так называемый стек (от английского слова

«stack», означающего «стог, скирда, поленница»).

В регистрах X и Y должны располагаться числа, участвующие в любой двуместной операции. Делается это так. Сначала одно из этих чисел набирается на клавиатуре и тем самым записывается в регистр X. Потом нажимается клавиша «↑». Содержимое регистра X при этом копируется в регистре Y. Если теперь набрать на клавиатуре новое число, оно запишется опять-таки в регистр X, стирая прежнее записанное в нем число. После этого можно нажимать клавишу требуемой двуместной операции.

Результат всякой операции записывается в ре-

гистр X, потому он и виден на индикаторе. (В дальнейшем для краткости мы иногда будем писать не «регистр X, регистр Y», а RX, RY).

Порядок расположения чисел в регистрах X и Y безразличен, если числа складываются или перемножаются. Порядок их расположения при вычислении числа в некоторой степени, например, в виде X^Y , задается самой записью операции: в RX — основание, в RY — показатель степени. А вот на случай вычитания и деления порядок расположения операндов следует запомнить.

При вычитании: в RX — вычитаемое, в RY — уменьшаемое. Для лучшего запоминания советуем представлять результат операции вычитания в виде $Y - X$.

При делении: в RX — делитель, в RY — делимое. Советуем представлять результат деления в виде Y/X .

Попробуйте выполнить на микрокалькуляторе вычитание $42 - 17$ и деление $54:18$ и получить соответственно 25 и 3.

Рассказывая об арифметических операциях, мы упоминали только о регистрах X и Y. Между тем и при вводе чисел для выполнения любой операции на микрокалькуляторе, и при ее совершении происходит движение чисел по всему стеку, которое затрагивает еще один регистр, примыкающий к стеку и обозначаемый X1.

После ввода в регистр X некоторого числа мы нажимаем клавишу «↑», при этом содержимое регистра X копируется в регистре Y, прежнее содержимое RY смещается в RZ, содержимое RZ — в RT, а старое содержимое регистра T пропадает.

Когда над числами, находящимися в регистрах X и Y,

3. НАШ НОВЫЙ ДРУГ — КОМПЬЮТЕР

совершается какая-либо арифметическая операция и ее результат записывается в регистр X, то прежнее содержимое регистра X пересылается в регистр X1, содержимое регистра Z смещается в RY, содержимое RT — в RZ, оставляя после себя в регистре T свою копию. То, что было в регистре Y, при этом пропадает.

Когда содержимое регистра X используется для вычисления какой-либо функции, результат записывается в тот же регистр X. Прежнее его содержимое пересылается в регистр X1. Содержимое других регистров не меняется.

Заметим, что названия всех функций, вычисляемых на микрокалькуляторе, написаны над клавишами. Чтобы совершить любую «надклавишную» операцию, нужно предварительно нажать кнопку «F», а уже потом соответствующую клавишу. Заметим также, что вычисление любой функции называется одноместной операцией, поскольку требует для выполнения одного исходного числа.

Если число, находящееся в регистре X, не введено туда с клавиатуры, а является результатом какой-либо операции (не только арифметической, но любой, выполненной по какой угодно команде, кроме команды «Сх»), то ввод нового числа в регистр X смещает его содержимое в регистр Y, содержимое RY — в RZ, содержимое RZ — в RT, а содержимое RT стирается.

Это еще не все возможные перемещения чисел по стеку. Допустим, вам захотелось использовать содержимое регистра X1. Нажмите клавиши «F» и «Vx» (Вверх). Число из регистра X1 переместится в X, а чис-

ла, находившиеся до этого в регистрах стека, сместятся точно так же, как при вводе нового числа: из RX — в RY, из RY — в RZ, из RZ — в RT, из RT — пропадает.

Можно заставить числа двигаться по стеку и в обратном направлении, нажав клавиши «F» и «C» (на клавиатуре микрокалькулятора этот символ образован двумя полукруглыми стрелками, замкнутыми друг на друга). Тогда число из RT перейдет в RZ, из RZ — в RY, из RY — в RX, из RX — в RT и RX1.

Можно переставить числа, находящиеся в RX и RY, нажав клавишу «XY». Содержимое регистра X при этом перейдет не только в RY, но и в RX1.

Естественно, встает вопрос: зачем нужны эти экзотические команды, ведь с первого взгляда их использование напоминает игру с кубиком Рубика? Повернул одну грань — поменялась картинка в трех смежных. Так же и здесь: нажал клавишу, числа в стеке запрыгали, как шарики.

Дальше вы поймете, что описанные команды очень выгодны для программирования на микрокалькуляторе.

В самом начале мы упомянули о двух состояниях или режимах работы на калькуляторе. Первый, о нем речь шла выше, называется «Режимом вычислений» или «Автоматическим режимом» и устанавливается сразу после включения калькулятора, второй — «Режимом программирования». Он устанавливается нажатием клавиш «F» и «ПРГ». Для возвращения в «Режим вычислений» нужно нажать клавиши «F» и «АВТ». Так

как клавиша «F» самостоятельной роли не играет, а позволяет лишь выбирать функции, написанные над другими клавишами, то в дальнейшем наборы, подобные «F», «ПРГ», будем обозначать единой записью FПРГ. То же относится к наборам, начинающимся и с другой из служебных клавиш — «K», «П», «ИП».

Итак, включаем микрокалькулятор, нажимаем клавиши FПРГ. В правом углу экрана появляются цифры 00. Программируемый микрокалькулятор готов к приему программы.

Программа для ПМК представляет собой набор команд, следуя которым машина обрабатывает информацию, то есть решает запрограммированную задачу.

Полная совокупность команд вместе с правилами их употребления и толкования образует язык микрокалькулятора. Правда, в отличие, скажем, от русского языка, где слова расчленяются на буквы и могут изменяться при склонении и спряжении, язык микрокалькулятора напоминает скорее китайский или японский. Он состоит из несклоняемых слов — «иероглифов» и только порядком их следования определяется смысл текстов — программ для ПМК. Каждый «иероглиф» — надпись на клавише или над ней (а в нижнем ряду клавиш — и под нею). Две клавиши — «K» и «F» — самостоятельной роли не играют. По своему действию они подобны переключателю пишущей машинки, от которого зависит, будут печататься прописные или строчные буквы. Если требуется выбрать «иероглиф», написанный на клавише, то нажать нужно только ее,

если же выбирается действие, написанное над клавишей, то предварительно нужно нажать клавишу «F», а в некоторых случаях (о них будет сказано особо) клавишу «K».

Не играют самостоятельной роли также клавиши «П» и «ИП». Они образуют команду лишь в сочетании с какой-либо цифровой или буквенной клавишей.

Команды, образующие программу, размещаются в специальном запоминающем устройстве калькулятора, называемом программной памятью. Оно состоит из последовательности занумерованных ячеек, в каждую из которых может попасть только одна команда, точнее, ее код — двузначное число. Именно по кодам и различает калькулятор одну команду от другой. Занумерованы ячейки двузначными числами от 00 до 97. Следовательно, всего их 98. Такова максимальная длина программы, которую можно записать в микрокалькулятор.

Условно все команды микрокалькулятора можно разделить на два класса: команды, используемые в программе, и команды, предписывающие порядок работы микрокалькулятора. Команды второго класса вводятся в режиме вычислений и будут рассмотрены, когда мы будем описывать процесс отладки программ и проведения вычислений по ним.

Первый класс команд можно разделить на такие группы: 1) вычислительные команды; 2) команды обмена информацией; 3) команды управления ходом вычислений; 4) команды, использующие режим косвенной адресации. Последние по

своим функциям не отличаются от команд второй и третьей групп, но используют способ адресации, называемый косвенным. Поэтому они будут рассмотрены отдельно. Так же отдельно будет рассмотрена и команда КНОП.

Так как область чисел, с которыми может работать микрокалькулятор, ограничена, все вычисления проводятся с восемью разрядами. Поэтому некоторые операции могут привести к появлению «аварийного останова» (авоста) — в случае, когда операция не может быть выполнена. Наибольшую осторожность надо соблюдать при умножении и делении. Часто бывает, что конечный результат цепочки операций лежит в пределах возможностей ПМК, а на промежуточном этапе возникает авост.

К арифметическим можно отнести еще одну команду (/—/). Не следует путать ее с вычитанием (—). Она одноместная, использует только регистр X. Работа команды состоит в изменении знака числа, находящегося в этом регистре (плюса на минус или минуса на плюс). Если же она используется после команды ВП и ввода порядка числа, задаваемого в экспоненциальной форме, то меняет она знак порядка, а не числа.

Остальные вычислительные команды используются для расчета значений различных функций. Названия их написаны над соответствующими клавишами, поэтому, чтобы получить значение нужной функции, требуется предварительно нажать клавишу «F».

Какие же функции позволяет вычислить ПМК? Вот они:

- извлечение квадратного корня ($\sqrt{\quad}$);
- возведение в квадрат (x^2);
- получение обратной величины ($1/x$);
- возведение числа 10 в любую степень (10^x);
- возведение в степень числа e — основания натуральных логарифмов (e^x);
- вычисление десятичного и натурального логарифмов (\lg и \ln);
- вычисление тригонометрических функций, аргументы которых могут быть заданы как в градусах, так и в радианах (\sin , \cos , tg), а также обратных тригонометрических функций, (\arcsin , \arccos , \arctg).

Аргумент каждой из этих функций всякий раз берется из регистра X. Туда же записывается результат. Пржнее содержимое этого регистра после выполнения операции запоминается в X1, а числа в других регистрах не меняются.

Особняком среди команд подобного рода стоит возведение числа в степень (x^x). Возводимое число берется из регистра X, а степень — из регистра Y. После выполнения команды результат, как обычно, заносится в RX; то, что было там прежде, переходит в RX1, а вот содержимое RY не стирается, оно остается на месте, так же как и содержимое двух других регистров. Надо сказать, что эту команду «Электроника БЗ-34» выполняет хуже прочих: результата приходится ждать долго, да и точность его ниже, чем при выполнении других команд. Так, 2^2 , вычисленное по этой команде, «равно» 3,9999996. Для сравнения: та же величина, вычисленная по команде x^2 , равна в точности четырем. Калькуляторы первых выпусков при

использовании этой команды иногда дают вообще неправильные результаты. Поэтому рекомендуем по возможности ее избегать.

Перемещением чисел по регистрам стека и регистрам данных занимается вторая группа команд, названных в нашем списке командами обмена. Четыре из них работают только с регистрами стека. Это \uparrow , FBx, XY и Fo. Действия этих команд описаны ранее.

Ряд команд позволяет пересылать содержимое RX в адресуемые регистры, называемые также регистрами данных. Их всего 14. Первые десять обозначаются числами от 0 до 9, последние — буквами A, B, C, D. Для записи в них информации служат команды ПО, П1, ..., П9, ПА, ..., ПД. Обратите внимание, что буквы A, B, C и D написаны под клавишами, но после нажатия клавиши «П» воспринимаются именно они.

Считывается информация из адресуемых регистров в регистр X с помощью команд ИПО, ИП1, ..., ИПА, ..., ИПД.

Условно к этим же командам можно отнести и команду F π . Все действие ее состоит в вызове из постоянного запоминающего устройства записанного там числа π в регистр X.

В языке микрокалькулятора нет специальных команд ввода. Числа просто набираются на клавиатуре и автоматически заносятся в регистр X. Однако набор числа можно поручить и программе. Например, если нужно умножить содержимое R0 на 2, мы пишем: ИПО 2 \times .

Так же можно поступать и с многозначными числами. К примеру, если второй сомножитель будет не 2, а 4,58, то соответствующий

фрагмент программы выглядит так: ИПО 4,58 \times .

При этом для записи числа будут использованы четыре ячейки памяти, по одной на каждый символ. Чаще всего такая запись нецелесообразна, но если все адресуемые регистры заняты, а в программной памяти место есть, то такой способ записи чисел оказывается полезным.

Если число задано в экспоненциальной форме, то для отделения мантиссы от порядка используется команда ВП.

И наконец, последняя команда этой группы — Sx. Она «стирает» содержимое регистра X, или, иначе говоря, засылает в него число 0.

Стоит заметить, что ее действие не идентично засылке нуля в регистр X с помощью цифровой клавиши «0». Команда Sx просто изменяет содержимое регистра X, не затрагивая остальных регистров, при нажатии же клавиши «0» ввод нуля в регистр сдвигает все числа в стеке «снизу вверх», как при вводе любого нового числа.

Рассмотренные выше команды составляют уже достаточный минимум для написания несложных программ, выполняющих расчеты по последовательно записанным формулам.

Однако очень часто встречаются задачи, для решения которых этого набора недостаточно. Ведь даже в таком элементарном случае, как решение квадратного уравнения, сначала требуется проверить знак дискриминанта квадратного трехчлена и в зависимости от него выбрать тот или иной путь решения, то есть вычислять либо действительные корни уравнения, либо действительную и мнимую части корней комплексных. К счастью, в языке микро-

калькулятора есть набор средств для проведения подобных операций. Они реализуются с помощью команд управления программой.

Сначала рассмотрим наиболее часто встречающуюся «команду останова» C/П. Она используется для остановки процесса вычислений, чтобы дать пользователю возможность либо прочесть полученный результат, либо ввести какие-нибудь команды с клавиатуры. В каждой программе обязательно должна быть хотя бы одна команда C/П.

Команда безусловного перехода БП передает управление команде, адрес которой записан сразу после нее. Фактически она занимает две смежные ячейки памяти. В первой — собственно БП, во второй — адрес перехода: две цифры. Кстати, обратите внимание, что две цифры подряд, нажатые после «БП», записываются одним кодом, совпадающим с этими цифрами.

Команды условного перехода также передают управление, но только при выполнении определенных условий. Действие их можно описать так: если условие, записанное в команде, выполнено, то следует отработать команду, написанную после команды перехода, в противном случае — передать управление по указанному в команде адресу. В качестве условия в этих командах используется сравнение RX с нулем.

Например, команда $Fx < 23$ работает так. Проверяется содержимое регистра X. Если оно меньше нуля, то выполняется команда, записанная сразу после команды перехода, в противном случае — команда по адресу 23. Команд условного перехода четыре:



($x < 0$, $x = 0$, $x \neq 0$, $x \geq 0$). Так как названия их написаны над клавишами, то перед ними требуется нажимать клавишу «F».

Есть среди команд перехода четыре команды, специально приспособленные для организации циклов, то есть для многократного выполнения заданной последовательности команд. Эти команды: L0, L1, L2, L3.

Перед ними также надо набирать клавишу «F». После любой из этих команд, как и после предыдущих, записывается адрес перехода. Работают они так. При каждом обращении к команде организации цикла из содержимого соответствующего регистра R0, R1, R2 или R3 вычитается единица. Если при этом получается не нуль, то управление передается команде по адресу перехода, если же результат

вычитания равен нулю, то команде, записанной после адреса перехода. Таким образом, можно, записав в один из регистров R0 — R3 некоторое число n , добиться выполнения определенного куска программы n раз.

Последней из команд перехода рассмотрим ПП — переход на подпрограмму. Структура ее та же, что и остальных команд этой группы: сначала сама команда, потом — адрес перехода. Сначала кажется, команда ПП работает так же, как и БП: и та и другая прерывают последовательность выполнения команд программы и «безусловно» передают управление по адресу, указанному в следующей после нее ячейке. Но если действие команды БП на этом заканчивается, то команда ПП наряду с передачей управ-

ления записывает в программный стек адрес команды, следующей после нее. Поэтому когда подпрограмма выполнена, то по последней ее команде — а это обязательно должна быть команда В/О — в программном стеке восстанавливается «старый» адрес, и программа продолжает работать с него.

Наконец команда КНОП. Набирается она двумя клавишами «К» и «НОП» и является «пустой» командой. Она не совершает никаких действий. Употребляется чаще всего при отладке программ. Если выясняется, что какая-нибудь команда лишняя, то, чтобы не переписывать остальные, на ее место просто записывают «пустую» команду КНОП. При этом адресация всех остальных команд не меняет-

ся. Точно так же работают команды K1 и K2.

Перейдем теперь к командам косвенной адресации. Они незаменимы, когда приходится перебирать информацию, лежащую в разных адресуемых регистрах, варьировать адреса переходов и т. д. Напомним, что служат они для обмена информацией между операционным регистром X и регистрами данных, а также для передачи управления в программах. Для набора их используются те же клавиши, что и для обычных команд, но начинается набор каждый раз с клавиши «К».

Команды косвенной записи в регистры:

КП0, КП1, ..., КП9, КПА, ..., КПД.

Они записывают содержимое регистра X непосредственно в регистр, указанный в команде, а в другой, в тот, номер которого записан в указанном регистре. Аналогично действуют и команды косвенного считывания:

КИП0, КИП1, ..., КИП9, КИПА, ..., КИПД.

Например, если в R7 записано число 5, а в R5 — число 10, то команда КИП7 вызовет в регистр X число 10, то есть содержимое регистра, номер которого записан в R7, — пятого. Иначе говоря, при косвенной адресации регистры, упоминаемые в командах, используются не как хранилища данных, а как хранилища адресов данных. Естественно, возникает вопрос: а как обратиться к регистрам с буквенными «номераами»?

Разработчики ПМК такую ситуацию предусмотрели. Все адресуемые регистры микрокалькулятора имеют сквозную нумерацию. Регистр A числится под номером 10, регистр B — 11. RC — 12 и RD — 13. Поэто-

му использование команды КП9 при содержимом регистра 9, равном 11, передает число из RX в регистр B.

Однако все выше описанное относится лишь к косвенным обращениям к регистрам с именами 7, 8, 9, A, ..., D. При использовании других регистров происходят вещи еще более интересные. Числа, хранящиеся в них, модифицируются, то есть изменяются во время косвенной команды, и обмен информацией происходит по этому модифицированному адресу. Модификация состоит в том, что содержимое регистров 0, 1, 2, и 3 уменьшается на единицу, а содержимое регистров 4, 5 и 6 на единицу же увеличивается.

Проследим этот процесс подробнее. Если в регистр 1 записать число 8, то по команде КИП1 будет сделано следующее. Во-первых, содержимое R1 будет уменьшено на единицу и станет равным 7, и, во-вторых, в регистр X будет передано число, хранящееся в 7-м регистре. Еще пример. Если мы запишем в R4 число 11, то команда КП4 передаст число из регистра X в 12-й регистр, то есть в регистр C.

Иногда при программировании на микрокалькуляторе бывает удобно «забыть» об основной цели косвенно-адресных команд и использовать лишь их «побочный эффект» — модификацию содержимого регистра при обращении к нему с помощью косвенной команды. Например, если мы хотим просто увеличить на единицу содержимое регистра 4, 5 или 6, то вместо команд, скажем, ИП4 1+П4, достаточно записать КИП4. При этом нас может не интересовать, что будет вызываться в операционный регистр X, — нам вполне

достаточно побочного эффекта от этого процесса, то есть увеличения содержимого самого регистра 4 на единицу. Таким же образом при необходимости можно уменьшать содержимое регистров 0, 1, 2 или 3.

Теперь о косвенно-адресных командах передачи управления. От обычных, ранее рассмотренных команд, выполняющих те же функции, эти команды отличаются тем, что адрес, на который нужно передавать управление, должен содержаться в адресуемом регистре, фигурирующем в команде. Например, команда КБПА, если в регистре A записано число 20, эквивалентна команде БП 20. Занимают косвенно-адресные команды такого типа одну ячейку в памяти вместо двух, положенных для хранения их прямых аналогов.

К командам этой группы относятся команды безусловного перехода: КБП0, КБП1, ..., КБП9, КБПА, ..., КБПД команды условного перехода:

$K x < 0$ 0, ..., $K x < 0$ Д

$K x = 0$ 0, ..., $K x = 0$ Д

$K x \neq 0$ 0, ..., $K x \neq 0$ Д

$K x \geq 0$ 0, ..., $K x \geq 0$ Д

и команды перехода на подпрограмму:

КПП0, ..., КППД.

Все, что было сказано о модификации адреса, относится и к этим командам. Используются косвенно-адресные команды передачи управления сравнительно редко, во всяком случае, значительно реже команд косвенно-адресной записи и считывания. Основное поле их действия — некоторое улучшение сервиса и экономия ячеек программной памяти, правда, ценой использования регистров данных.

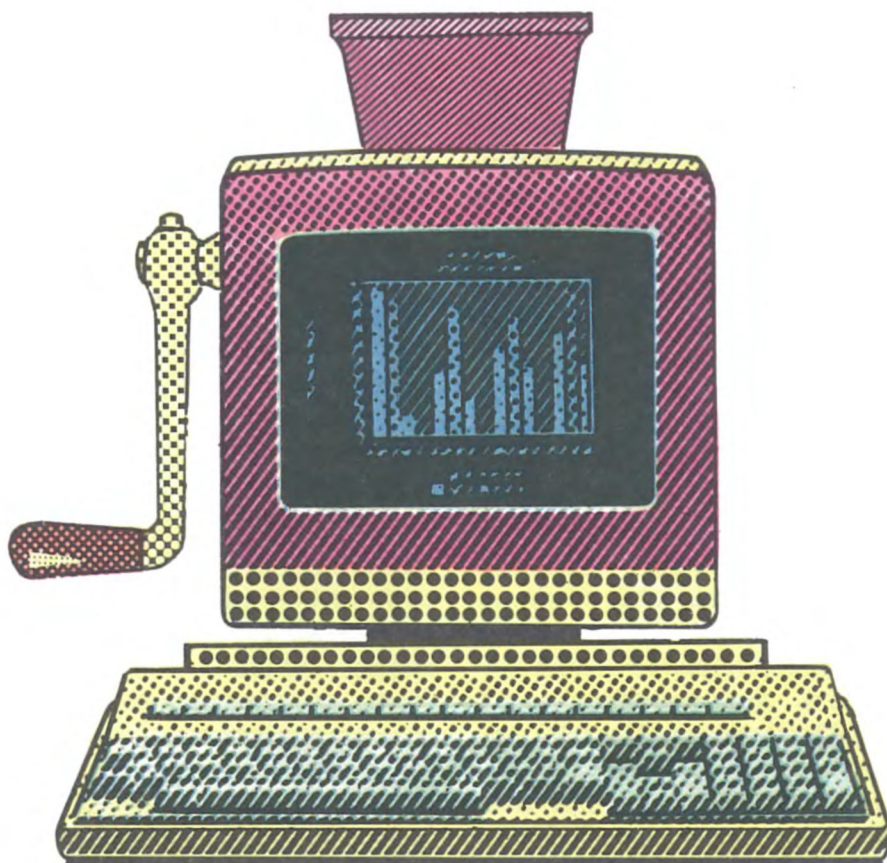
Вы уже, наверное, обратили внимание, что все команды косвенной адресации используют целые числа. А

что будет, если в регистре, с которым оперирует косвенная команда, записать число дробное? В этом случае ПМК «поступает» самым естественным образом. Он просто игнорирует дробную часть числа, отбрасывая ее при выполнении команды. Например, если в регистре 8 было записано число 3,14, то после выполнения команды КИП8 в операционный регистр будет вызвано содержимое 3-го регистра, а в R8 окажется число 3. Это дает, кстати, возможность довольно простым способом выделять целую часть числа, что очень полезно, так как специальной функции для этих целей у ПМК нет. Правда, надо отметить, что для чисел меньше единицы и тем более для отрицательных этот способ непригоден.

При использовании регистров от нулевого до шестого дробная часть числа также отбрасывается, а остаток модифицируется, как обычно.

Познакомившись с кратким описанием команд, можно приступить к созданию программ-текстов, написанных на языке микрокалькулятора.

Каждый, кому приходилось писать школьное сочинение, отчет о проделанной работе, не говоря уже о статьях и книгах, знает, что первый вариант, как правило, нуждается в доработках. Этот процесс называется редактированием. С этой же проблемой сталкивается и любой программист. Но если при редактировании текстов на бумаге достаточно вычеркнуть ручкой какое-нибудь слово или фразу и вписать поверх них новое, сделать вставку на отдельном листе и т. д., то при редактировании программ в памяти микрокалькулятора все это невозможно. Чтобы изменить команду, необ-



ходимо вычеркнуть ее из памяти и на ее место вписать новую. Вставить команду можно, лишь заменив (переписав) весь «хвост» программы вслед за «лишней» командой...

Но первое, с чего следует начать знакомство с редактированием программ — это способы чтения команд, записанных в памяти.

Начальный адрес чтения устанавливается обычно в режиме вычислений. Если вы находитесь в режиме ввода программы (ФПРГ), то нажмите клавиши «FABT». После этого можно ввести либо команду «В/О», тогда просмотр начнется с начала текста, либо БП nm. В этом случае первой появится команда, записанная по адресу nm.

Сами нажатия этих клавиш никак на содержимом индикатора не отразятся. Но после перехода в режим «ПРГ» вы сможете читать программу, записанную в память именно с заданного адреса (00 или nm). Чте-

ние можно продолжить, нажимая клавиши ШГ: со стрелкой вправо — для движения вперед по тексту и со стрелкой влево — для обратного хода. Используя эти две клавиши, нетрудно «подогнать» к индикатору, этому своеобразному «окну» в программу, любой адрес. И когда код команды загорится в левом углу индикатора, вы сможете исправить ее, если она неверна.

Заметим сразу: сделать это так, как исправляют неверно введенные числа, то есть с помощью клавиши «Сх», нельзя. С ее помощью нельзя даже стереть только что введенную команду, если вы сразу убедились, что она неправильна. В режиме программирования команда Сх (стереть x) — это обычная команда, которая будет просто записана в память. Стереть неправильную команду можно только одним способом — записать на ее место новую или записав на

ее место пустую команду КНОП (нет операции).

Несколько сложнее заменять адреса переходов. Дело в том, что адрес перехода состоит из двух цифр и воспринимается как один код только тогда, когда перед ним набрана команда перехода (БП, ПП, FL0, Fx=0). В противном случае каждая цифра записывается в отдельную ячейку, поэтому для изменения адреса перехода нужно вернуться на предыдущую команду, то есть команду перехода, записать ее вновь и уже затем вписать новый адрес. Например, чтобы изменить во фрагменте: 17. БП 18.60 число 60 на 59, недостаточно установить указатель адреса на 18 и ввести 59. В этом случае цифры 5 и 9 будут записаны в две смежные ячейки. При этом адрес перехода станет равен 05, да и команда по адресу 19 окажется испорченной. Если же установить указатель адреса на 17 и повторить ввод всей команды перехода БП 59, то все будет правильно.

С помощью команд \overline{III} и \overline{III} можно добраться до любой ячейки памяти. Заметим, что сами эти команды в память не записываются. Они лишь меняют показание счетчика команд. Также не фиксируется в памяти команда FCF. Она используется для отказа от «полунабранной» команды. Речь идет о командах, требующих нажатия нескольких клавиш, то есть начинающихся с нажатия клавиш «F», «K», «P», «IP». Случайно нажатая клавиша (из перечисленных) еще не отображается на индикаторе — калькулятор «ждет», что будет нажато затем, чтобы правильно интерпретировать всю последовательность

использованных клавиш, например Flп или КИП5. Чтобы отменить этот режим ожидания, и служит команда FCF. После нее интерпретация нажимаемых клавиш начинается «с нуля».

Используя описанные приемы отладки, можно добиться, чтобы текст программы, введенный в память калькулятора, полностью совпал с тем, что написано на бумаге.

И наконец еще об одной — последней команде, применяющейся при отладке вновь созданной программы. Эта команда ПП. В режиме программирования она интерпретируется как переход на подпрограмму и требует ввода адреса этой подпрограммы. Нажатая же в режиме вычислений, она вызывает выполнение очередной команды программы, вывод на индикатор содержимого регистра X и останов. В этом случае аббревиатуру, написанную на клавише, можно расшифровать как потактовый проход.

Когда программа введена и проверена, на счет ее запускают нажатием клавиши С/П. Предварительно не забудьте перейти в режим вычислений «ФПРГ» и установить начальный адрес, с которого должны начать выполняться команды. Чаще всего мы стартуем с «начала», то есть с адреса 00. Для этого нужно нажать клавишу В/О. Если же вы хотите начать выполнение программы с адреса mn , то вместо В/О наберите БП mn . После этих манипуляций калькулятор начнет работать по введенной программе.

Для остановки работающей программы, если вам показалось, что считает она слишком долго, следует на-

жать клавишу С/П. Недаром аббревиатура названия этой команды расшифровывается как СТОП/ПУСК.

Таковы команды языка, редактирования и отладки программируемого калькулятора, с помощью которых можно записывать, изменять, отлаживать и запускать программы на нашей мини-микроЭВМ.

И. Коган (Данилов)

Игры без программ

Правила предлагаемых игр весьма просты и составлены так, что играть можно с непрограммируемым, то есть простейшим или инженерным микрокалькулятором, скажем, «БЗ-36» или «МК-51». Каждый ход в любой из этих игр заключается в том, чтобы набрать на цифровых клавишах какое-то число и нажать еще одну-две клавиши для выполнения тех или иных математических операций.

Во многих развлечениях с микрокалькулятором эта карманная ЭВМ подобна кубуку Рубика или, скажем, коробке для игры в «пятнашки».

По такому же принципу можно разработать целый ряд игр для микрокалькулятора. В каждой из них даются два числа — начальное и конечное. Путем последовательных математических действий, выполнимых на микрокалькуляторе, из начального числа требуется получить конечное. В зависимости от того, какие действия оговорены правилами игры (только сложение и вычитание, только извлечение квадратного корня и т. д.), получаем ту или иную игру.

Конструирование числа сложением и вычитанием.

Пусть начальное число равно, например, 17. Из него путем прибавления и вычитания двузначных чисел надо получить 117.

Один из способов подобного «конструирования» числа 117 может быть таким. К исходному числу 17 прибавляем сначала 98. Для этого производим действия: $17+98=$ на индикаторе 115.

Число 115 меньше целевого числа 117. Прибавляем еще 14:

$+14=$ на индикаторе 129.

Теперь в уме нетрудно оценить, что для получения требуемого числа надо вычесть из уже полученного $12: -12=$ на индикаторе 117.

Число 117 сконструировано за три хода. А сможете ли вы сконструировать то же число за два хода?

У этой игры много возможных решений. Поэтому в ней могут участвовать много игроков. Попробуйте!

Как в любой игре, предполагается, что каждый из игроков честно соблюдает правила. Если забыть о них, то можно, конечно, взять произвольное число, например 78, затем с помощью калькулятора вычислить $17+78=95$, а потом из 117 вычесть 95 и получить 22. Значит, число 117 можно конструировать в виде $17+78+22$. Но это уже не игра. Если из коробки для игры в «пятнадцать» высыпать все шашки и потом их просто положить на свои места, то это тоже не игра. Другое дело, когда вы используете свои умственные способности. Их применение при конструировании чисел и делает игру интересной, а заодно развивает навыки вычислений в уме.

Теперь дадим задачи для упражнения.

Прибавлением и вычитанием двузначных чисел сконструировать:

из числа дней в неделе (7) число дней в году (365);

из года Великой Октябрьской социалистической революции (1917) год начала освоения космоса человеком (1961);

из среднего роста ребенка при рождении (50 см) свой теперешний рост.

Если в игре участвует много игроков, то ход игры желательно протоколировать. Выигрывает тот, кто сконструировал число за наименьшее число ходов.

Конечно, в эту игру могут играть и дети младшего возраста. Ограничение относительно двузначных чисел в таком случае можно снять. Сперва детей надо ознакомить с игрой на таком, скажем, примере: из 4 сконструировать 13 прибавлением и вычитанием чисел, не больших 5:

$4+5=$ на индикаторе 9

$+5=$ на индикаторе 14

$-1=$ на индикаторе 13.

Так пример за примером быстро вырабатываются навыки прикидочных вычислений в уме. Условия игры можно менять в соответствии со знаниями и умениями ребенка. Для детей можно предложить такие задачи:

из 3 сконструировать 10 прибавлением и вычитанием только чисел 2 и 3;

из 0 сконструировать 17 прибавлением и вычитанием только чисел 4 и 5 (или только 3 и 5);

из числа дней в неделе сконструировать число дней в месяце (30, 28 и 31) прибавлением и вычитанием чисел 3 и 5;

из возраста одного из родителей сконструировать свой возраст в годах вычитанием и прибавлением 3 и 7.

Конструирование числа умножением.

Замена сложения и вычитания умножением усложняет конструирование числа. В этой игре вместо одного числа лучше указать интервал чисел. Например, пусть требуется из числа 12 сконструировать число от 137 до 139 включительно. Одно из возможных решений такое:

$12 \times 12 =$ на индикаторе 144
 $\times 0,8 =$ на индикаторе 115,2
 $\times 1,1 =$ на индикаторе 126,72. Дальше продолжайте сами, пока на индикаторе не появится число из интервала (137,139).

Чем больше начальное число и чем уже целевой интервал, тем больше ходов требуется для конструирования числа. Попробуйте с помощью подобного конструирования, исходя из числа 19, попасть в интервал (94, 95), из 23 в (171, 173), из 937 в (402, 404), из 71 в (555, 556).

В эту игру можно играть и вдвоем, и в более широком кругу. Выигрывает тот, кто за меньшее количество ходов попал в заданный интервал.

Эта игра была придумана в США. Там она бытовала под названием «Испорченный микрокалькулятор» и формулировалась так: «У калькулятора не работает клавиша деления, а вам надо найти такой множитель X, который в произведении с заданным числом A давал бы другое заданное число B». Например, надо найти X такое, что $23 \cdot X = 851$.

Попытаемся решить задачу подбором. Пробуем: $23 \times 40 =$ на индикаторе 920
 $23 \times 35 =$ на индикаторе 805.

Экспериментируя далее, можно вскоре определить, что $X = 37$.

Тот вид, в котором эта игра была изложена здесь с

самого начала, придал ей профессор Х. Мейснер из ФРГ. Предложив свой вариант игры людям разных возрастов, он сделал немало любопытных наблюдений.

Так, например, 13-летнему школьнику, который не имел никаких навыков в подобных играх, профессор дал задачу: «Какое число надо умножить на 17, чтобы произведение находилось в промежутке от 560 до 585?» Решение ученик нашел только с 14-й попытки. Когда же он решил еще около 20 задач такого типа, Х. Мейснер в качестве очередного задания поставил такое: «Найти число, которое в произведении с множителем 37 дает число от 960 до 965». С этой задачей ученик справился в три приема. Интересно, что никакие беседы о стратегии поиска решения с ним не проводились. Он сам разработал выигрышную стратегию и продемонстрировал при этом неплохое, как говорят педагоги, чувство числа.

Стоит заметить, что родители и учителя, протестующие против введения микрокалькуляторов в учебный процесс средней школы, нередко заявляют, будто школьники, увлекшись карманными ЭВМ, разучатся считать, утратят чувство числа. Приведенный пример опровергает это предвзятое мнение. Он показывает, что при умелом использовании микрокалькулятор, напротив, обостряет чувство числа — пусть не совсем в том виде, в котором оно понималось ранее.

Можно согласиться, что работа с микрокалькулятором несколько снижает навыки умножения многозначных чисел столбиком и их деления уголком. Но благодаря электронной вычисли-

тельной технике эти навыки сегодня обесцениваются точно так же, как механические часы заставили нас забыть искусство узнавать время по солнцу.

Угадай число.

Это популярная математическая игра, в которую без калькулятора играют так. Один игрок задумывает некоторое число, например 38, а другой должен его отгадать. Второй игрок называет некоторое число, например 20. В ответ он слышит: «Маловато». Допустим, второй игрок вслед за этим называет число 40. Ответ: «Многовато». И так до тех пор, пока число, названное вторым игроком, не совпадет с задуманным числом. Затем отгадывающий и загадывающий меняются ролями, и игра проводится вновь и вновь. Выигрывает тот, кто за меньшее число ходов отгадывает задуманное число.

Эту игру легко перенести на микрокалькулятор. Сперва рассмотрим такой ее вариант, при котором играющему дается сильная подсказка. В этом случае задуманное число используется в качестве константы для автоматического вычитания. Примем, что первый игрок задумал число 387. Он набирает его на микрокалькуляторе, но так, чтобы второй игрок не видел, какое число набирается.

С $387 - 387 =$ на индикаторе 0 (для «БЗ-36») или

С $387 - \text{—} =$ на индикаторе 0 (для «МК-51»). Микрокалькулятор дается в руки второму игроку. Он пробует отгадать загаданное число и набирает, например,

$500 =$ на индикаторе 113. Значит, 500 больше загадан-

ного числа. Предположим, что далее второй игрок набирает

$250 =$ на индикаторе — 137. Это означает, что 250 меньше числа, задуманного первым игроком. Когда же второй игрок наберет $387 =$ на индикаторе 0, то появление нуля оповестит: число отгадано.

Эту игру также можно порекомендовать школьникам младших классов. Она развивает чувство числа, вырабатывает навыки сравнения чисел (больше, меньше, равно).

Для тех, кому легко даются вычисления в уме, отгадывание дву- или трехзначного числа не составит труда, поскольку его нетрудно найти по остатку вычитания. Например, когда на введенное число 250 калькулятор отвечает — 137, то загаданное число быстро определяется суммированием в уме: $250 + 137 = 387$.

Для искусных вычислителей предлагаем вариант более сложной игры.

Лабиринт.

Рассмотрим цифровые клавиши, за исключением 0, как маршрутные точки для путешествия по клавишному полю (см. рис.).

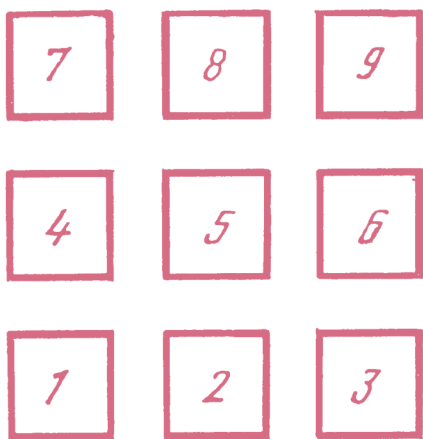
Договоримся:

а) путешествие всегда начинается в точке 1 и кончается в 9;

б) в каждой точке маршрута можно побывать только один раз;

в) в качестве следующей точки маршрута можно выбрать только близлежащую точку. Например, из «1» можно перейти на «4», «5» или «2», но нельзя переходить на «7» или «6».

Предположим, что вы как ведущий игры задаете маршрут 1—2—3—5—9. По-



готовьте микрокалькулятор к игре:

С $12359 - 12359 =$ на индикаторе 0 (для «БЗ-36») или

С $12359 - - =$ на индикаторе 0 (для «МК-51»).

Участнику игры надо отгадать ваш маршрут. Допустим, он набирает $14859 =$ на индикаторе 2500. Появление на индикаторе ненулевого числа означает, что маршрут не отгадан. Но по показанию индикатора видно, что предпоследняя точка маршрута «5» определена верно. Теперь можно попробовать другой маршрут с учетом новой информации: $12659 =$ на индикаторе 300. Результат всего лишь трехзначный. Стало быть, отгадана еще одна точка маршрута, клавиша «2»; теперь мыслимы только два возможных маршрута: 12459 или 12359 . Набрано 12359 , и на индикаторе появляется 0. Маршрут отгадан.

Измерение делением.

Игру «Угадай число» можно усложнить и другим способом — заменив сравнение чисел их делением. Предположим, что ведущий задумал число 38. Он готовит микрокалькулятор к автоматическому делению на 38: $С\ 38 : 38 =$ на индикаторе 1 (для «БЗ-36») или

$С\ 38 : : =$ на индикаторе 1 (для «МК-51»).

Микрокалькулятор передается второму игроку, который пробует

$20 =$ на индикаторе 0,526.

Так как на индикаторе число меньше единицы, то 20 — маловато. Игрок вводит большее число:

$40 =$ на индикаторе 1,05.

Результат больше единицы, стало быть, 40 — многовато. Когда игрок наберет $38 =$, то на индикаторе появится 1. Число отгадано.

Попутно отметим одну деталь, которая может удивить начинающего. Если выполнить деление 20 на 38 с помощью разных микрокалькуляторов, то ответ 0,526... может появиться на индикаторе некоторых из них в не совсем привычной форме (выпишем полностью все знаки частного):

5.263 1578—01.

Расшифровывается эта форма следующим образом: выписывается число, знаки которого идут от левого края индикатора, и умножается на десятку в степени, равной числу, стоящему с правого края индикатора: $5,2631578 \cdot 10^{-1} = 0,52631578$.

Это так называемая экспоненциальная форма представления чисел. Если калькулятор «владеет» ею, то так, в частности, в нем выводятся на индикатор числа, не превосходящие единицу по абсолютной величине.

Отыщи корень.

До сих пор, играя с микрокалькулятором, мы использовали лишь арифметические действия: сложение и вычитание, умножение и деление. Между тем многие микрокалькуляторы, даже непрограммируемые, способны выполнять и

более сложные операции: нахождение числа, обратного данному, вычисление синуса и косинуса от заданного числа, возведение числа в квадрат и т. д. Эти операции найдут свое применение в дальнейших играх, а пока в порядке разминки опробуем какую-нибудь из них.

Наберем на клавиатуре некоторое число, скажем 25, и нажатием клавиши x^2 возведем его в квадрат:

$25x^2 =$ на индикаторе 625.

На микрокалькуляторах, не имеющих клавиши x^2 , возведение в квадрат можно проделать так:

$25 \times =$ на индикаторе 625.

Предположим, что у нас есть один из таких калькуляторов, не имеющих, впрочем, клавиши $\sqrt{\quad}$. В случае такого неудобства корни можно отыскивать подбором, притом с любой точностью.

Требуется, например, отыскать $\sqrt{784}$. Иными словами, надо найти число, при умножении которого на себя получится 784.

Экспериментируем:

$30 x^2$ на индикаторе 900

$27 x^2$ на индикаторе 729

$28 x^2$ на индикаторе 784,

то есть искомым корень.

Интересно, за сколько ходов вы сможете найти квадратные корни из чисел 1369, 1936, 1024?

Справившись с этими задачами, попытайтесь найти приближенные значения квадратных корней из чисел 500, 700, 2000.

Эту игру можно предложить младшим школьникам, еще не знакомым с понятием квадратного корня. Игра поможет получить хорошее представление об этом понятии. Сначала же, разумеется, задачу надо ставить так: найти число, которое, будучи умножено само на себя, дает заданное число, например, 169 или 361, 256 или 196.

Отыщи корень другим способом.

Закройте одной рукой индикатор, а другой наберите $1369\sqrt{\quad} - 1369\sqrt{\quad} =$ (для «БЗ-36») или $1369\sqrt{\quad} - \quad =$ (для «МК-51»).

Снимите руку с индикатора. Если вы все сделали правильно, то на индикаторе должен быть 0. Теперь наберите пробное число, по вашему мнению, равное квадратному корню из набранного вначале числа, и нажмите клавишу $=$. Если на индикаторе положительный результат, то ваше число больше искомого корня, если отрицательный — то меньше. В случае успеха на индикаторе появится 0.

Подобным образом можно проверить навыки возведения числа в квадрат. Наберите, например, $23x^2 - \quad =$ (для «МК-51»).

Оцените в уме квадрат числа 23, наберите его и нажмите клавишу $=$. Больше или меньше ваш ответ, нежели истинный?

Можно потренироваться в оценке обратной величины числа. Наберите $0,0016 \ 1/x - 0,0016 \ 1/x =$ (для «БЗ-36») или $0,0016 \ 1/x - \quad =$ (для «МК-51»).

Нетрудно догадаться, что обратное значение числа 0,0016 меньше 1000. Наберите некоторое трехзначное число и нажмите клавишу $=$. Как близка ваша оценка к точному значению $1/0,0016$?

Если вы желаете, чтобы калькулятор определял не абсолютную, а относительную погрешность ваших оценок, готовьте калькулятор к игре иначе. При отыскании корня:

С $1369\sqrt{\quad} : 1369\sqrt{\quad} =$ (для «БЗ-36») или $С 1369\sqrt{\quad} ::=$ (для «МК-51»).

При возведении в квадрат:



С $23x^2 ::=$ (для «МК-51»).

Далее игра идет так же, как в предыдущих вариантах. Если после вашего хода на индикаторе появится число, большее единицы, — ваша оценка завышена. Если меньше — занижена.

Стрельба по цели.

Представьте себе, что у вас имеется пушка, стреляя из которой вам надо забросить трос помощи на маленький островок среди большого озера. Остров находится на расстоянии 0,650—0,665 км от пушки. Дальность полета определяется углом выстрела α и скоростью V , с которой из пушки вылетает метательный снаряд, увлекающий за собой трос.

Человек, знакомый с механикой, без труда определит, что дальность полета равна

$$(v^2/g) \sin 2\alpha.$$

Примем для простоты: скорость v , с которой вылетает снаряд из пушки, постоянна и притом такова, что дробь v^2/g равна единице. Тогда дальность стрельбы выражается величиной $\sin 2\alpha$ и отыскивается совсем просто: набираем на клавиатуре удвоенное значение приня-

того угла стрельбы и нажимаем клавишу \sin .

Установим для начала угол выстрела равным 10° : $20 \sin$ на индикаторе 0,342.

При таком значении угла трос упадет в озеро — недолет.

Увеличим угол выстрела до 20° . Набираем $40 \sin$ на индикаторе 0,642. Снова недолет — трос опять упадет в озеро. Дальше экспериментируйте сами. Если вам удалось найти угол выстрела, при котором трос упадет на остров, то попытайтесь найти еще одно значение угла, при котором трос также достигнет цели.

Удивительные числа.

Эту игру придумал Х. Гутцер из ГДР. На клавиатуре микрокалькулятора цифровые клавиши размещаются обычно так:

7	8	9
4	5	6
1	2	3

Составим четырехзначные числа, которые набираются по часовой или против часовой стрелки из цифр, расположенных в углах любого прямоугольника, образованного цифровыми кла-

вишами. Например: 7931, 2365, 4631. Попробуем поделить эти числа на 11. Чтобы облегчить деление и каждый раз не набирать «делить на одиннадцать», подготовим микрокалькулятор к автоматическому делению:

C 11 : 11 = на индикаторе 1 (для «БЗ-36») или

C 11 : : = на индикаторе 1 (для «МК-51»).

Теперь набираем указанные числа и производим деление на 11:

7931 = на индикаторе 721

2365 = на индикаторе 215

4631 = на индикаторе 421.

Поразительно, но оказывается, что все эти числа делятся на 11 без остатка.

Продолжайте экспериментировать с другими числами: 1782, 3289 и т. д. Поищите числа, которые набираются указанным способом, но не делятся на 11.

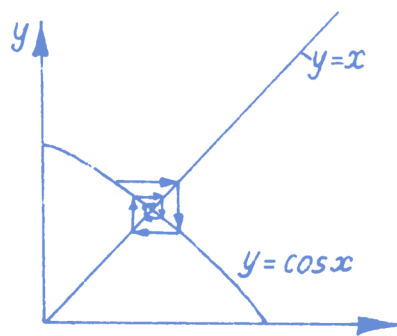
В чем же секрет чисел, набравшихся прежде и разделившихся на 11 без остатка? Калькулятор зовет пытливых к размышлению, к исследованию...

Упрямый косинус.

Возьмите произвольное число, большее нуля и меньшее единицы. Представьте, что это радианная мера некоторого угла. Возьмите микрокалькулятор, переведите переключатель $R=\Gamma$ в положение R (радианы) и вычислите косинус взятого угла, нажав соответствующую клавишу. От полученного числа вновь возьмите косинус, с полученным на сей раз проделайте ту же операцию... и так далее.

Числа, последовательно загорающиеся на индикаторе, будут постепенно приближаться к некоторому числу.

Выберите другое число, большее нуля и меньшее



единицы, и проделайте с ним ту же процедуру. Числа на индикаторе вновь будут приближаться к тому же пределу, что и прежде.

Новые попытки убедят вас: каждый раз наблюдается стремление к пределу и этот предел не зависит от начального числа. Что же представляет собой этот предел? Какова его природа, его смысл?

Попытаемся разобраться в этих вопросах с помощью графика (см. рис.). На нем каждая вертикальная черточка, соответствующая n -му нажатию на клавишу \cos , означает: вычисляется значение косинуса от аргумента x_{n-1} , горизонтальная: полученное число принимается за новое значение аргумента $x_n = \cos x_{n-1}$.

Стрелочки складываются в ломаную линию, спирально стягивающуюся к точке пересечения двух графиков: $y = x$ и $y = \cos x$. Это означает, что предел, к которому приближаются числа, появляющиеся на индикаторе, есть корень уравнения $x = \cos x$.

Вот так протекает этот сходящийся процесс, если за начальное приближение взять семь десятых:

$$\begin{aligned}x_0 &= 0,70000000; \\x_1 &= 0,76484222; \\x_2 &= 0,72149167; \\x_{11} &= 0,73958447; \\x_{12} &= 0,73874869.\end{aligned}$$

В пределе $x = 0,739085$.

Метод, по которому в этих задачах отыскивается корень уравнения типа $x =$

$f(x)$, называется методом итераций. Примеры его применения нетрудно умножить. Так с помощью вычислительных машин решаются не только отдельные уравнения, но и системы линейных и дифференциальных уравнений.

Симметричные числа.

Возьмите любое двузначное число в интервале от 11 до 99. Прибавьте к нему зеркальное число, то есть прочитанное справа налево исходное число. Например: $39 + 93 = 132$.

К результату прибавьте его зеркальное число: $132 + 231 = 363$. Вы получите симметричное число, то есть такое, чье зеркальное число равно ему самому. В нашем примере для получения симметричного числа потребовалось два сложения.

Таким способом можно получить симметричное число из любого от 11 до 99. Для большинства чисел при этом потребуется от одного до четырех сложений. Лишь четыре числа в этом диапазоне несколько более упрямы — для получения из них симметричных чисел нужно шесть сложений. А два числа очень упорны — каждое из них превращается в симметричное лишь после 24 сложений! В результате получается симметричное число из 13 знаков.

Разрядная сетка нашего калькулятора вмещает всего 8 знаков, так что, если вы желаете получить такой результат, вам придется получать его вручную. Желаем успеха!

Для опытных любителей математических развлечений добавим, что эту игру можно вести и с трехзначными или даже с более длинными числами. Для подавляющего большинства чисел при этом удастся достичь сим-

метрии. Но есть несколько орешков, которые до сих пор не поддаются никаким усилиям, хотя над некоторыми уже произведено более тысячи сложений.

Математикам пока не удалось установить, для всех ли чисел можно получить симметричный итог путем конечного числа сложений или существуют такие числа, которые для этой игры не годятся.

Для более увлекательных игр требуются калькуляторы классом повыше — те, что могут работать по вводимым в них программам.

Подготовлено по материалам книги: Микрокалькуляторы в играх и задачах. М., «Наука», 1986.

Истинная правда

«Громадный метеорит врезался с космической скоростью в наш звездолет и пробил его насквозь, оставив в обшивке дыру размером с человеческую голову. Воздух со свистом хлынул наружу».

«Пилот наконец решился и нажатием кнопки отправил в реактор последние остатки топлива. Тысячетонная громадина корабля дрогнула и медленно двинулась вверх. Люди были спасены».

Подобными эпизодами изобилуют поступающие в редакции рассказы начинающих фантастов. Оценивать такие произведения затруднительно. Интуитивно ясно, конечно, что после столкновения с «громадным метеоритом» от звездолета ничего не останется, а «последних остатков топлива» не хватит, чтобы даже при «десятикратной перегрузке» обеспечить взлет «тысячетонной громадины» со сколько-нибудь приличной планеты. Но

какими аргументами подкрепить интуитивные соображения? Не будешь же каждый раз проделывать громоздкие вычисления по соответствующим формулам... А компьютер — где его взять?

К счастью, несколько лет назад, когда промышленность освоила массовый выпуск программируемых микрокалькуляторов, положение коренным образом изменилось. Ошибки начинающих авторов легко подразделить на несколько четко выраженных классов, и никто не мешает составить для ПМК десяток программ, в которые заложены наиболее типичные фантастические ситуации. Так что, введя в машину различные соображения относительно размеров «громадного метеорита» и величины его «космической скорости», нетрудно через минуту получить число, подсказывающее такой, например, ответ: «К сожалению, при самых оптимистических соображениях насчет размеров и скорости придуманного Вами метеорита диаметр проделанной им дыры в обшивке значительно превысил бы длину звездолета, то есть последний попросту превратился бы в пар, так что увлекательные приключения Ваших героев после такого столкновения никоим образом не могли иметь места. Рукопись возвращаем...»

Одна из наиболее распространенных ситуаций — взлет с безатмосферных планет и посадка на их поверхность. Соответствующая проверочная программа называется «Лунолет-1»:

00.ИПД 01. $Fx < 0$ 02.09
03. ↑ 04.ИП8 05. ÷ 06.ХУ
07.ПП 08.90 09.ИПА 10. $Fx \neq 0$
11.43 12. $Fx < 0$ 13.33 14.2
15.× 16. ↑ 17.ИП4 18.ИП3
19.—20× 21.ИПВ 22. Fx^2
23.+24. $F\sqrt{\quad}$ 25.ИПВ 26.—

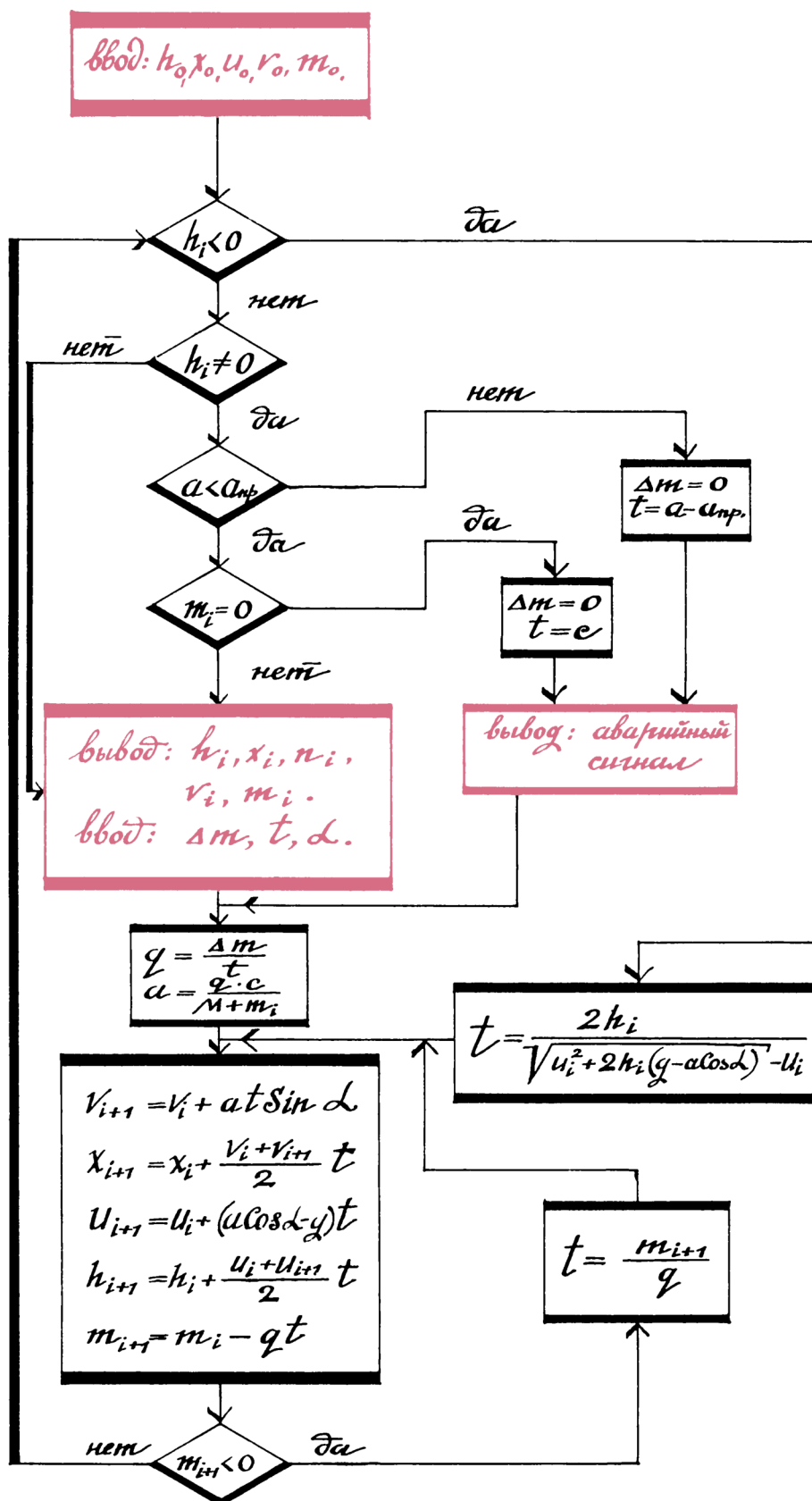
27. ÷ 28. ↑ 29.ИП8 30.×
31.БП 32.90 33.ИПД 34. $Fx \neq 0$
35.86 36.ИПЗ 37. Fx^2 38. $F\sqrt{\quad}$
39.ИП7 40.—41. $Fx < 0$ 42.87
43.ИПВ 44.ИПА 45.С/П
46.П1 47.П2 48. $Fx \neq 0$ 49.43
50. ÷ 51.П8 52.ИП5 53.ИПД
54.+ 55. ÷ 56.ИП6 57.×
58.ПЗ 59.ИП4 60.— 61.ИП2
62.× 63.ИПВ 64.+ 65.ПВ
66. FVx 67.+ 68.2 69. ÷
70.ИП2 71.x 72.ИПА 73.+
74.ПА 75.ИПС 76.ИП2
77.ИПО 78.× 79.— 80.ПС
81.ИПД 82.ИП1 83.— 84.ПД
85.В/О 86.ИП6 87.ИП9
88.С/П 89.Сх 90.П1 91.ХУ
92.П2 93. $Fx < 0$ 94.50
95.ИПЗ 96.БП 97.59

Надо сказать, что, помимо многочисленных писем, сюда довольно часто приходят посетители. Как правило, это весьма необычные люди. Один несет новый проект или действующую модель вечного двигателя либо безопорного движителя; другой рассказывает о встречах со «снежным человеком» и чудом дотянувшими до наших дней мезозойскими динозаврами. А года два назад в редакции одного из научно-популярных журналов появился человек, который категорически утверждал, что он якобы... провалился к нам из будущего, из конца ХХI века!

Этот человек бывал у нас на протяжении двух недель. И каждый раз рассказывал что-нибудь о будущем. Его внимательно выслушивали (так полагается по долгу службы), записывали его рассказы на магнитофон. Ничего, впрочем, особенного в них не было — любой при желании мог придумать и не такое. Да и как проверить? Потом он куда-то пропал, и вскоре все забыли о нем.

А недавно возникла мысль: проанализировать с помощью ПМК и рассказы этого человека. Ведь все

* * *



записано на пленку, а пленки хранятся в архиве!

И вот на столе расшифровка старой магнитофонной записи. Рядом — готовая к вычислениям «Электроника».

Приготовьтесь и вы. Введе-

дите в свой ПМК программу «Лунолет-1» и переведите машинку в режим вычислений. Будем работать вместе.

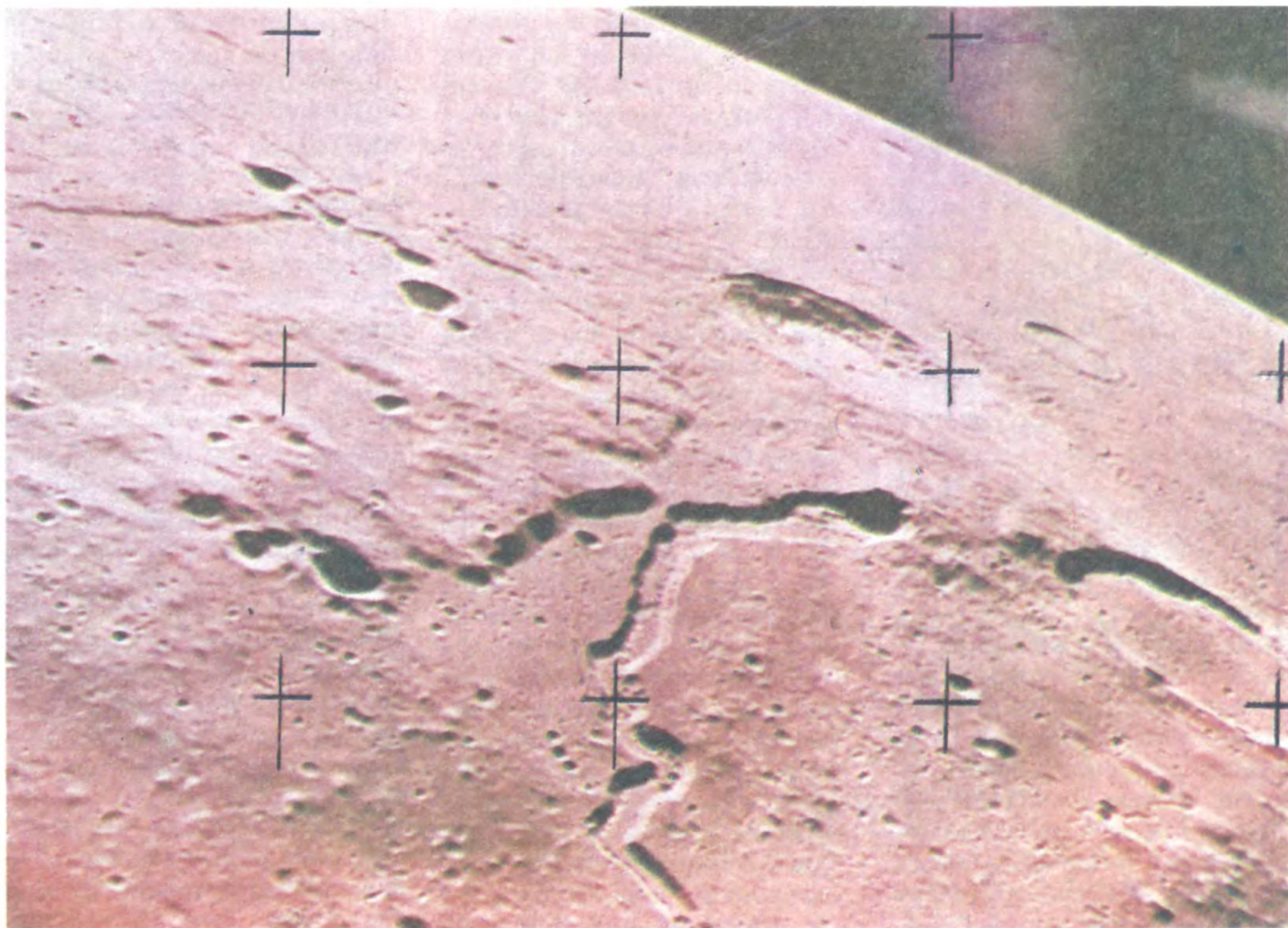
Вот одна из его историй. (В скобках — наши краткие комментарии.)

Я управлял ракетой один-единственный раз в жизни.

Конечно, в юности, как и многие мои сверстники, я мечтал стать космонавтом. Но мечты эти рассеялись на первой же медкомиссии: при перегрузках больше трех «же» мне становилось плохо. А тех, кто не выдерживал пятикратной, к дальнейшим испытаниям не допускали. Волей-неволей пришлось забирать документы. Я подал на вычислительную технику, через шесть лет благополучно защитил диплом и — ирония судьбы! — был направлен по распределению на Луну, в Центр имени С. П. Королева. Там я работаю до настоящего времени.

После того как по соседству нашли неорганическую нефть, Центр сильно разросся. Теперь это настоящий город с населением порядка трех тысяч человек. Прикрывающие его купола соединены туннелями. Как в метро, только стены прозрачные. Это, грубо говоря, большие трубы, протянутые прямо по поверхности. Один из туннелей ведет к аэровокзалу. Космопорт Центра — обширный комплекс, он обслуживает всю Солнечную систему. По роду работы я часто бываю в порту, потому что корабли напичканы электроникой, рано или поздно что-нибудь выходит из строя, а чинить вычислительную технику — это моя специальность и прямая обязанность.

Случай, о котором я упомянул, произошел летом 2087 года. Работа у нас строится циклически: четыре месяца трудимся, два отдыхаем. Как правило, на Земле. Родные тоже прилетают иногда погостить — пассажирская линия Земля — Лу-



на открылась давно. В то лето ко мне прилетел сын Сергей. Я не видел его несколько месяцев, за это время он сильно подрос. Ему скоро двенадцать, и он бредит космонавтикой. Мы с женой надеялись, что самостоятельное путешествие на Луну очень его обрадует. Да и сам он, как она сообщила, в ночь накануне вылета совсем не спал.

Но когда я встретил его на аэровокзале, он выглядел разочарованным.

— Ерунда! — сказал он. — Сидишь в кресле, стюардесса носит конфеты и воду в тубиках. Как в самолете. Никаких перегрузок. Хоть и невесомость, но плавать по воздуху запрещают. Заставляют сидеть в кресле да еще и пристегиваться. Вот если бы самому в рубке сидеть за штурвалом и нажимать рычаги...

Он горько вздохнул и

грустил еще минут пять, пока мы добирались домой. Потом отправился погулять. Вернулся через полчаса, разочарованный еще больше: на поверхность не выпускают, скафандр не дают, все кругом самое обычное, деревья и люди как на Земле. Никакой Луны нет. Разве что тяжесть поменьше, но это ему неинтересно — после двух-то суток невесомости на борту лайнера. Было уже поздно, и я уложил его спать. А потом и сам лег — завтра с утра на работу. Я пообещал взять его с собой: там интересно — вычислительные машины, манипуляторы и прочее.

Но наутро мне позвонили — появилось срочное дело. Я все записал, потом сварил кофе. Сергей был уже на ногах. Когда мы позавтракали, я сообщил, что планы изменились, так что пусть пока посидит дома. Он

сначала возмутился, но потом смирился с необходимостью. Я пояснил, что буду в отсутствие каких-нибудь полтора часа.

— Папа! — сказал он. — А куда ты пойдешь?

— На космодром.

— А, — он разочарованно махнул рукой. — Я там уже был. Там скучно. Никуда не пускают.

— Мне не в порт, — объяснил я. — Мне на местные линии. Это небольшая площадка в стороне от главного поля. Лайнеров, на каком ты летал, там нет. Только лунолеты местного сообщения.

— Настоящие? — поинтересовался он.

— Разумеется, не игрушечные. Но они маленькие — всего две тонны сухого веса. Вернее, сухой массы. Здесь ведь все весит в шесть раз меньше, чем на Земле.

— Знаю, — отмахнулся

Сергей.— А что ты там будешь делать?

— Работать,— пожал я плечами.— На одной из этих машин отказал киберпилот. Надо посмотреть. Если какой-нибудь пустяк, сделаю на месте. А если что-то серьезное...

— Ты пойдешь в рубку? — Глаза у него расширились.

Я невольно расхохотался.

— Какая там рубка! Кабина, два кресла. Не повернешься.

— Но приборы там? — продолжал он допрос.— Рычаги управления там?

— Конечно,— простодушно признался я.— Где же им быть еще?

— Возьми меня с собой,— потребовал он.

— Но это не игра,— попробовал я объяснить.— Это работа...

— Папа! — сказал он.— Ты мне обещал.

— Только не говори маме,— попросил я, сдаваясь.

Несколько минут спустя электрокар мчал нас по направлению к космопорту. Улицы в этот час были пусты — все на работе. Незадолго до аэровокзала мы свернули в боковой туннельчик, ведущий к служебному выходу. Сергей был в приподнятом настроении, что-то напевал.

В проходной я показал удостоверение вахтенному.

— А это что за гражданин?

— Он со мной,— сказал я.— Это мой сын.

— А где его документ? — спросил вахтенный.

— Ему не положено. Он еще маленький.

Минуту вахтенный размышлял. Ситуация, очевидно, была для него новой.

— Ладно, пускай идет. Под личную ответственность.

Скафандры нам выдали без проблем. Сережке, ко-

нечно, он был великоват, но только чуть-чуть — не такой уж он у меня маленький.

— Баллоны стандартные,— предупредил выпускающий.— На два часа. Справитесь?

— Конечно. Мне и часа хватит с гарантией. Двадцать минут туда, двадцать обратно, десять на месте. Ну и десять на всякие осложнения.

— Ясно,— добродушно сказал выпускающий.— Только пусть лучше осложнений не будет. Хорошо?

Я опустил забрало, и мы прошли в воздушный шлюз. А спустя короткое время уже шагали под голубым светом Земли. Была середина лунной ночи — на Земле в эти дни новолуние.

Сказать, что Сергей был восхищен,— значит, ничего не сказать. Он был ошеломлен. Похоже, он никак не ожидал, что ему выдадут скафандр. Настоящий, с индивидуальной системой жизнеобеспечения. На Земле я бы в таком весил килограммов сто пятьдесят, да и Сергей потянул бы на добрую сотню. Мы осторожно шагали по ровному рголиту — искусственного покрытия здесь никто не прокладывал, убрали крупные камни, и все. Не для важных персон. Но дорожка была ухоженной. До нас здесь ходили многие — все, кто работал на дальних лунных базах и опорных пунктах. Строители, ученые, инженеры...

— А там что такое? — Сергей вновь обрел способность задавать вопросы. Его голос в моем шлемофоне звучал непривычно.

Он показывал направо. Там выступали из-за горизонта массивные строения промышленного блока.

— Это тебе неинтересно, Сережа,— сказал я.— Там

качают нефть и гонят из нее керосин. Для двигателей.— Я махнул в сторону космопорта.— И еще вырабатывают жидкий кислород.

— Для скафандров? — Он произнес слово «скафандр» с особенным выражением. Я усмехнулся:

— И для скафандров, конечно. Но в основном его везут на те же заправочные станции. Горючее ведь не будет гореть без окислителя.

Некоторое время мы шагали в молчании. Идти стало чуть труднее — дорожка поднималась к вершине обычного для Луны плоского холма. Еще сотня шагов — и мы достигли места своего назначения. Нашим глазам открылась стоянка лунолетов.

Их было десятка два — старые, но надежные машины. В первом приближении это скругленный конус высотой метра три с половиной, опирающийся на три амортизатора типа «паучья нога». Вся верхняя часть прозрачна, для облегчения обзора. Это и есть та самая «рубка», в которую так рвался мой сын.

На боку каждого лунолета стоял опознавательный номер — две цифры, начертанные светящейся краской и обведенные черной каймой. Наш был с краю. Без дополнительных приключений мы забрались внутрь. При виде многочисленных циферблатов у Сергея разгорелись глаза.

— Это настоящий корабль? — спросил он.

Я понял. На Земле похожие машины стоят в каждом парке отдыха. Атракционы. Влезай в люк и испытывай всякие ощущения.

— Самый что ни на есть,— сказал я.

— А как им управлять? Я усмехнулся.

— Проще простого. Вот

этот ящик, — я указал пальцем, — называется киберпилот. Если тебе нужно попасть, допустим, на базу «Циолковский», ты набираешь на пульте задание, потом нажимаешь вот эту кнопку, и киберпилот благополучно доставляет тебя куда надо. Но сейчас именно он-то и неисправен.

Это Сергея на время утихомирило. Он, видимо, ожидал чего-нибудь в духе земных аттракционов, когда сам даешь вводные и тебя швыряет в разные стороны. Я аккуратно снял с киберпилота пломбу и сдвинул лицевую панель. На вид все в порядке. Дал на схему напряжение. Циферблаты на пульте ожили.

Где же искать повреждение?..

— А это что за рычаги? — услышал я голос Сергея.

Я обернулся. Оказывается, он устроился во втором кресле. Играл в космонавта. Указывал он на рычаги ручного управления двигателем.

— Это для посадки, — объяснил я. — Киберпилот всегда доставит тебя куда надо, но он не знает местность. Вдруг там трещина, скажем, или какой-нибудь камень. Тогда нужно дать небольшой импульс, чуть притормозить спуск, чтобы машина проскочила опасное место.

Я опять повернулся к киберпилоту. Однако не тут-то было.

— А как дать импульс? — спросил Сергей.

— Ты мне мешаешь, — сказал я. — Откуда мне знать? Впрочем, гляди: написано «Расход топлива» и цифры, ага, в килограммах. Рычаг стоит здесь, значит, ты собираешься истратить 65 килограммов топлива. А возле правого рычага — «Время» и тоже цифры. Это,

видимо, время, за которое ты собираешься свое топливо израсходовать. Меньше время — больше тяга, а если время больше — тяга соответственно меньше. Сейчас рычаг стоит на цифре три. Значит, если ты подашь эту команду на двигатель, он израсходует 65 килограммов за три секунды.

— А это много?

— Не знаю, — сказал я. — По-моему, все равно что ничего.

— А как подать команду на двигатель?

— Откуда я знаю? — огрызнулся я. — Ты мне мешаешь. Пульт наверняка заблокирован, а баки пусты. Полюбуйся, — я ткнул пальцем в индикатор. — Видишь? «Топливо», четверста. Всего-навсего! А шкала на три с половиной тонны.

— А как... — продолжал он допрос.

— Отстань от меня! — скомандовал я. — Ты мне мешаешь. Скажешь еще хоть слово, тут же идем домой. И вообще, отошлю тебя к маме.

Он обиженно умолк, а я занялся киберпилотом. Набрал контрольный тест — он прошел нормально. Набрал второй — тоже полный порядок. Что они там, спятили? Совершенно исправная машина. Я набрал третий тест. И тут началось...

(Пока рассказчик работает с вычислительной техникой, поработаем немного и мы. Не будем забывать о своей главной задаче — вывести его на чистую воду. Кульминационный момент, судя по всему, приближается, самое время нажимать клавиши ПМК. Программа введена, формируем и отправляем в регистр 9 аварийный сигнал Г: Сх ÷ ВП ВП П9. Теперь исходные данные. Дело происходит на Луне, ускорение свободного падения $1,62 \text{ м/с}^2$.

Набираем на клавиатуре 1,62 П4. Масса корабля без горючего две тонны, сюда нужно добавить массу скафандром (150 кг) и его сына (100 кг). Набираем 2250 П5. Двигатель, очевидно, работает на керосине и жидком кислороде, скорость истечения 3660 м/с. Набираем это число на клавиатуре и нажимаем П6. Очередь за предельным ускорением. По словам рассказчика, ему становится плохо уже при трех «же». Набираем на клавиатуре 9,81 З х П7. Скорость и высота равны нулю — нажимаем О ПА ПВ. Запас топлива 400 кг. 400 ПД. Вводим в регистр С ресурс жизнеобеспечения в секундах. Воздуха в баллонах было на два часа, двадцать минут герои повествования шли до стоянки, возьмется минут двадцать, да надо еще накинуть двадцать на обратный путь. «На всякие осложнения» им остается ровно час. Набираем 3600 ПС и соответственно 1 ПО. Исходные данные введены. Нажимаем В/О С/П. Через секунду на экране загорается высота — ноль. Нажимаем ХУ. На экране скорость — тоже ноль. Все правильно. Можно во всеоружии ждать грядущих событий. А они, несомненно, вот-вот последуют.)

... Ни один контрольный тест не проходил. Я бросил взгляд на часы: с момента, когда мы покинули воздушный шлюз, прошло уже сорок минут. Пора возвращаться. Я потянулся к рубильнику — снять с оборудования напряжение — и посмотрел на сына. Он продолжал играть в космонавта: нажимал какие-то кнопки, созерцал пляшущие на экранах кривые. Рычаги управления тягой стояли в прежнем положении. Он положил указательный палец на

большую красную кнопку. Неясное предчувствие шевельнулось у меня в голове.

— Не смей! — крикнул я.

Но было поздно. Под нами загрохотало, за прозрачным колпаком взметнулось пламя, чудовищной силы удар швырнул меня в кресло, и у меня потемнело в глазах...

(Значит, пульт все-таки заблокирован не был и команда прошла на двигатель. Не особенно, конечно, убедительно, но нас интересует фактическая сторона дела. Двухтонная машина, жалкие 60 кг топлива, и вдруг — «чудовищной силы удар»! К счастью, проверить данный эпизод нетрудно. Подадим ту же команду и на свой пульт: 65 ПП 3 С/П. На экране мелькают цифры, потом... загорается аварийный сигнал Г! Как это ни удивительно, перегрузка действительно превысила допустимую, рассказчик потерял сознание и какое-то время после отсечки двигателя не сможет управлять лунолетом. Снова нажимаем С/П.)

... Когда я очнулся, кругом было только небо. Сколько я был без сознания? Не знаю. Но мы падали, падали со страшной скоростью! Очевидно, за время моего беспометства ракета прошла вершину траектории и теперь стремительно неслась вниз. Сергей тянулся к рычагам управления. Но игры кончились. Какую-то секунду я не мог опомниться, но еще через секунду был у пульта. Что я мог успеть в такой ситуации? Заметил лишь показания индикаторов — скорость во семьдесят, высота триста с чем-то.

— Папа! — крикнул Сергей.

Что я мог успеть? Не

меня режима двигателя, я ударил по красной клавише. На нас снова обрушилась перегрузка...

(На нашем же индикаторе высота полета 169 м — везде округляем до целых. Нажимаем ХУ. Скорость 84 м/с. Ну что ж, будем считать, проверка закончена. Скорость еще более-менее, но высота в рассказе завышена вдвое. Можно откладывать ПМК в сторону... Впрочем, пока он добирался до пульта, прошло еще две секунды. Две секунды свободного падения с выключенным двигателем. Но куда можно упасть за две секунды? Разве что на Луну — не в небо же! Ладно, для очистки совести засылаем соответствующую команду: О ПП 2 С/П. На экране зажигаются высота 334, скорость 81. Невероятно, но цифры совпали! Почему же он утверждает, что лунолет падал? Непонятно. Но набираем новую команду 65 ПП 3 С/П. Опять сигнал Г — запредельные перегрузки! — вновь нажимаем С/П и ждем результата.)

... Когда я очнулся снова, мыг опять падали. Я рванул к красной клавише, но взгляд мой упал на индикатор высоты. Почти километр! И цифры росли! Значит, мы вовсе не падали — мы неслись вверх со скоростью реактивного истребителя! И в прошлый раз мы, конечно же, тоже поднимались! Меня ввела в заблуждение невесомость. Мы действительно падали, но падали вверх! И я, болван, включив двигатель, только усугубил наше и без того тяжелое положение. Зато теперь появилось время, чтобы сойтись с мыслями...

(Не так-то просто, оказывается, поймать его на слове! У нас очередной останов. Высота 916 м — действительно, почти кило-

метр, — скорость 166 м/с. Примерно 600 км/ч. Мало-вато, конечно, для истребителя, но... Можно ли считать это серьезной ошибкой? Будем объективны, оставим рассказчику право на художественное преувеличение. Как бы то ни было, включать двигатель он вроде пока не собирается, так что нашему ПМК можно тоже дать передышку.)

— Папа! — сказал Сергей, выбираясь из-под меня. — А почему ты говорил, что 65 килограммов — это все равно что ничего?

— Отстань от меня! — приказал я. — Марш во второе кресло и пристегнись!

Я проследил, как он это выполнил, и пристегнулся сам. Цифры на индикаторе высоты увеличивались, но все медленнее и медленнее. На киберпилот надежды нет, придется выкручиваться самому. Но пока, пожалуй, лучше не делать ничего. Гнать вверх бессмысленно. Вниз (а рядом с красной клавишей я углядел другую, «Реверс тяги») — еще хуже. Вот начнем падать, тогда... Я заранее установил рычаги в положение 25 кг и 2 с и ждал. Да, я сильно ошибся насчет этих килограммов. Похоже, в них большая сила...

— Папа! — подал голос Сергей. — А мы сможем улететь в космос?

— Отстань! — рявкнул было я, но вдруг у меня защемило сердце. Ребенок не понимал, что мы на волосок от гибели, для него это было игрой! — Сереженька, — сказал я как можно ласковее, — в космос мы с тобой еще слетаем. Но сейчас, пожалуйста, помолчи...

Мы взлетели уже почти на десять километров. Цифры на указателе скорости дошли до нуля и начали медленно расти, теперь уже с отрицательным знаком.

3. НАШ НОВЫЙ ДРУГ — КОМПЬЮТЕР

Когда скорость достигла примерно тридцати метров в секунду, я нажал красную клавишу. Высота к этому времени уменьшилась почти до девяти километров. Свободный полет продолжался ровно две минуты...

(Наконец-то появилась новая цифра, которую можно проверить. Две минуты с нулевой тягой. Команда: О ПП 120 С/П. После останова высота 9175, скорость — минус 28. Опять все сходится! Новая команда: 25 ПП 2 С/П. Машина рассчитывает маневр.)

... На сей раз перегрузка была терпимой. На индикаторах мелькали цифры. Высота почти не изменилась, но мы опять поднимались! Что ж, мы так и будем болтаться на этой высоте, пока не кончится все топливо? И весь кислород?! В отчаянии я установил рычаги в положение 10 и 10 и дал реверс тяги. Лунолет кувыркнулся двигателем вверх. Далеко внизу я увидел постройки Центра, обширные поля космодрома и крошечное пятнышко площадки, с которой мы так неосмотрительно стартовали...

(У нас после останова высота 9151, скорость — чуть меньше пяти метров в секунду и действительно снова направлена вверх. Вводим команду 10 ПП 10 ПП /—/ С/П. После останова высота равна 9044 м, скорость — 26 м/с со знаком минус. Снова падаем, и довольно быстро.)

... Скорость падения увеличивалась быстрее, чем я рассчитывал. Я решил притормозить: убрал реверс, установил 25 кг и 5 с и надавил на красную клавишу. Но она не поддалась. Видимо, эти пульты устроены так, что новая команда блокируется, пока не исполнена прежняя. И лишь когда де-

сят секунд истекли, корабль вновь крутанулся двигателем вниз. Но когда тот выключился, мы оставались все на тех же девяти километрах и опять, хоть и очень медленно, поднимались!..

(Команда: 25 ПП 5 С/П. Результат: высота 8984, скорость два с половиной метра в секунду, снова со знаком плюс. Он что, действительно собирается провести здесь всю оставшуюся жизнь?)

... Нет, решил я, так дело не пойдет. Если мне даже удастся установить приемлемую скорость спуска — допустим, пять метров в секунду, — и поддерживать ее до самого прилунения, то сколько времени на это уйдет? Полчаса? Час? Топлива не хватит наверняка. Да и кислород...

— Папа! — вновь подал голос сын. — Топлива всего двести...

— Двести десять! — рявкнул я. — Но помолчи же!

Мы уже снова падали — все быстрее и быстрее. Топлива осталось чуть больше половины. Но если, черт побери, на половине топлива мы ухитрились добраться сюда, то оставшейся половины должно хватить для возвращения! Если, конечно, его разумно тратить... Только как это — разумно?

Я решил выждать сколько возможно, а потом дать резкий тормозной импульс. Заранее установил рычаги в положение 100 и 3. На этот раз свободный полет продолжался полторы минуты. Мы уже опять неслись со скоростью истребителя, но только вниз. До поверхности оставалось чуть больше двух с половиной километров, когда я надавил красную клавишу. На нас вновь обрушились перегрузки...

(Полторы минуты свободного падения. О ПП 90 С/П. Высота 2652, скорость падения 143. Около 500

км/ч. Но сравнение с истребителем, как мы договорились, не ошибка, а всего лишь гипербола. Вводим 100 ПП 3 С/П. На экране, естественно, буква Г. Перегрузки снова превысили допустимую величину. Не слишком ли часто? Но нажимаем С/П. Высота 2123, скорость минус 32.)

... Когда вернулось сознание, высота упала на полкилометра, но скорость снизилась до тридцати метров в секунду. Маневр удался! Можно попробовать идти дальше с этой же скоростью, а где-нибудь ближе к поверхности повторить маневр. Только как удержать скорость? Я решил экспериментально подобрать нужную тягу. Для начала дал 10 кг за 20 с. К нам снова вернулся вес — правда, поменьше, чем на Луне, — но скорость все-таки росла. К окончанию маневра она достигла 50 м/с. Я увеличил тягу: те же десять килограммов, но теперь за 15 с...

(Повторяем оба маневра. 10 ПП 20 С/П. Высота 1314, скорость спуска 49. 10 ПП 15 С/П. Высота 515, скорость 58. Многовато!)

... Скорость все равно увеличивалась, а до лунной поверхности оставалось каких-нибудь полкилометра. Запас топлива — 90 килограммов. Делать было нечего. Я дал 35 кг за полторы секунды — и, конечно же, вновь отключился...

(Команда: 35 ПП 1,5 С/П. На экране Г — перегрузки. С/П. Результат: высота 390, скорость спуска 17. Примерно 60 км/ч. Это уже полегче.)

... Когда я пришел в себя, то понял, что маневр удался. До поверхности было еще почти 400 метров, а скорость упала больше чем втрое! Семнадцать метров в секунду — скорость электрокара! Я знал уже, как

ее сохранить. Я заслал 22 кг за 22 с. Сам я родился 22 марта. Сергей родился, когда мне было 22 года. И вообще, 22 — число для меня счастливое...

(Команда: 22 ПП 22 С/П. На экране мелькают цифры.)

... Я действительно угадал! Скорость почти не менялась, только в десятых долях. Высота равномерно уменьшалась: 250... 200... 150... 100... 50... И вдруг до меня дошло, что мы вот-вот врежемся в поверхность Луны, амортизаторы не удержат! Топлива оставалось 33 кг. Оставив рычаг расхода на прежней отметке, я рванул второй вниз до упора — 0,7 с — и давил, давил на красную клавишу. Но она поддалась, лишь когда до губительного удара осталось меньше секунды. Опять перегрузки...

(На экране: высота 13,5, скорость 17,5. Команда: 22 ПП 0,7 С/П. Г — перегрузки! Снова С/П.)

... Как ни удивительно, но я, видимо, отключился лишь на ничтожную долю секунды. Когда сознание вернулось ко мне, скорость была прежней, а от поверхности нас отделяли всего 7 метров. Двигатель молчал. Впоследствии я не раз задумывался, как такое могло случиться. Неужели сбой двигателя? Не знаю. Но в тот момент мне было не до размышлений. Палец лежал на клавише. Я повторил команду, послав в двигатель последние капли топлива. Я сделал это в тот же миг, как открыл глаза. Новый удар ускорений...

(Высота 7, скорость 17. Все сходится! Но не могли же одновременно подкачать и двигатель лунолета, и наша «Электроника»! Попробуем разобраться в ситуации — у нас-то время есть. Проверяем запас топлива. ИПД.

На индикаторе 11 кг. Значит, двигатель свои 22 кг отработал. Почему же тогда прежняя скорость? Смотрим время, на которое выключался двигатель. ИПД. На экране 21 с. Ничего себе, «ничтожная доля секунды»! Впрочем, и это нельзя считать неточностью рассказчика: выглядело-то все именно так, и откуда ему знать, сколько он был без сознания? У него ведь нет такой аппаратуры, как наша! А произошло следующее: расход был задан слишком большой, двигатель не только полностью погасил скорость, но и вновь разогнал лунолет вверх. И только после этого выключился на 21 секунду, за это время Луна вновь подтянула корабль к себе. А он думает, «сбой двигателя»! Но ладно, вводим последнюю команду: 22 ПП 0,7 С/П. Да, но ведь топлива осталось всего 11 кг, а задано 22! На индикаторе загорается буква Г. Перегрузки? На этот раз машина сигнализирует о более важном происшествии: команда на двигатель подана с превышением наличного запаса топлива. Когда оно иссякнет — а это случится ровно через 0,35 с, — он выключится окончательно. Нажимаем С/П. На экране мелькают цифры, и вдруг загорается ноль. Поверхность! Смотрим скорость: 3,7 м/с. Отличная посадка!

... Двигатель молчал. Я лежал в кресле в ласковых объятиях привычного лунного тяготения. Лунолет, покачиваясь на амортизаторах, стоял невдалеке от того места, откуда мы стартовали. На индикаторах застыли скорость — меньше четырех метров в секунду — и время — 350. Значит, мы летали неполные шесть минут...

Я повернул голову — как там мой Сергей? Он лежал

неподвижно, глаза его были закрыты.

— Сережа, — позвал я. Он не шелохнулся.

— Сережа! — заорал я. Он оставался недвижим.

Я рванул из кресла — меня не пустили ремни. Не помню, как я расстегнул пряжки, как очутился с ним рядом. Я тряс его, дергал — безрезультатно. Не знаю, сколько это продолжалось. И вдруг...

В моем шлемофоне раздался его громкий счастливый смех!

Он продолжал играть! Он играл в космонавта, убитого перегрузками!..

Потом, конечно, я много рассказывал об этом приключении. Все, само собой, изумлялись, как это мне, впервые оказавшемуся за пультом, удалось выполнить столь успешную посадку. Только один приятель, по профессии селенолог, выслушал все внимательно и произнес: «Неплохо! Но мне, я думаю, в такой ситуации хватило бы и шестидесяти!» Он имел в виду, что затратил бы на посадку не триста с чем-то килограммов топлива, как я, а всего шестьдесят. Я не понимаю, как это можно сделать — ведь на старте было сожжено шестьдесят пять, значит, и на финиш должно уйти минимум столько же! Однако в подробности он вдаваться не стал. Хватило бы, и точка! Эти селенологи лихие ребята — гоняют на своих лунолетах по всей Луне. Им виднее.

Сережка, разумеется, тоже хвастался всем подряд. Его версия происшедшего звучала примерно так: «Папа посадил корабль на Луну, зато в космос поднял его я!» Друзья, конечно, сильно ему завидовали.

И только маме он ничего не сказал. Потому что пообещал.

(Рассказ подошел к концу. Но, собственно, и наша работа закончена. Отодвигаем ПМК в сторону. Какие неточности удалось обнаружить? Никаких. Так, парочку преувеличений. А проверить такие факты, как последний, наша программа не в состоянии...

Если же кому-нибудь захочется выяснить, прав ли был тот лихой селенолог, сделайте это сами. Закачайте топливо в баки и держайте. Но только, пожалуйста, не забудьте перед стартом уменьшить массу лунолета на 100 кг. Вот так: ИП5 100 — П5. Пока не приобретете опыта, не берите с собой ребенка!)

Охота на инопланетных чудовищ

В недрах любой вычислительной системы, в частности и нашей «Электроники», обитают не менее диковинные создания, чем те, которые населяют, скажем, глубины инопланетных океанов. На индикатор ПМК, как известно, выводятся числа, не превышающие по величине 9,9999999 ВП 99 (9,9999999 — мантисса, 99 — порядок числа). Они для нас столь же привычны, как и обычные рыбы земных водоемов. Однако «Электроника БЗ-34» (и последующие модели ПМК) способна формировать числа гораздо большие (с порядком до 1000!), причем при соответствующем навыке каждое из них можно «изловить» (записать в регистр), проанализировать, а затем как-то использовать. Конкретный вид и свойства этих «арифмитических чудовищ» зависят от глубин, где они водятся (точнее, от величины поряд-

ка). «Охота» на них — занятие увлекательное и в ряде случаев небезопасное.

Вот краткая классификация «глубоководной фауны» ПМК. Глубины (порядки) до 100 заселены обычными числами. Следующий «этаж» (от 100 до 200) принадлежит ЕГГОГам; еще глубже (от 200 до 300) обитают ЗГГОГи — создания, вопреки своему зловещему виду, в высшей степени полезные, их легко приручить и заставить служить человеку.

Далее (от 300 до 400) располагается вотчина диких и неукротимых чудовищ, норвящих при малейшей оплошности со стороны охотника привести программу в негодность и заставить его выключить ПМК. Следующий этаж (от 400 до 500) заселен ОС-оборотнями — существами очень полезными, но, в свою очередь, подразделяющимися на многочисленные семейства. Еще ниже (от 500 до 600) располагаются владения Тьмы, при любом контакте с этой таинственной и грозной субстанцией индикатор гаснет, и приходится отключать ПМК. (Отметим, что с Тьмой можно случайно столкнуться и на других этажах.) Глубже, за пределы Тьмы, можно проникнуть лишь с помощью специального «водолазного оборудования» (соответствующих программ): глубины от 600 до 700 заселены медлительными С-ЕГГОГаами, еще ниже (от 700 до 800) обитают неповоротливые монстры, чьи повадки тем не менее заставляют вспомнить безудержных чудовищ 4-го этажа и «охота» на которых протекает аналогично. На предпоследнем этаже (от 800 до 900) безраздельно властвует Ноль (самый обычный, насколько удалось выяснить), дальше (от 900 до 1000) начинается зона

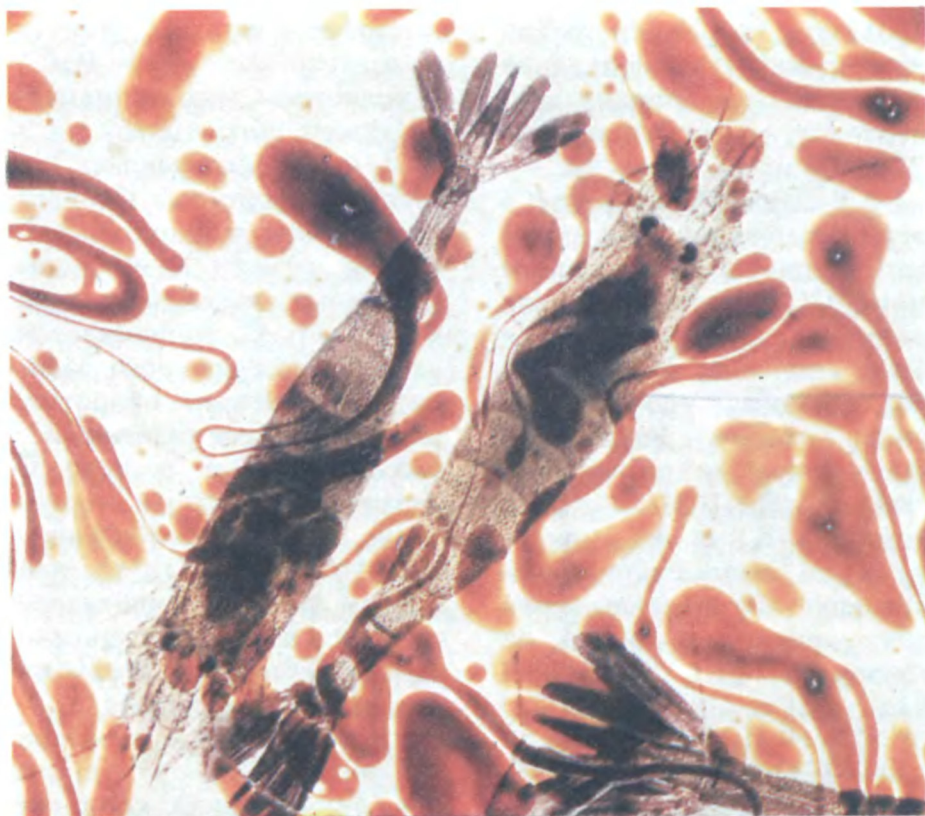
обычных чисел с постепенно уменьшающимися отрицательными порядками, наконец, после 1000 круг замыкается — на сцену вновь выступают числа с положительными порядками, затем ЕГГОГи, и все повторяется. А теперь познакомимся ближе с населением каждого этажа.

1-й этаж. Здесь, как уже отмечалось, обитают обычные числа. У них, конечно, много всяких любопытных свойств (как и у самых обыкновенных земных животных), но к предмету нашего разговора они не относятся.

2-й этаж. ЕГГОГи, населяющие глубины (порядки) от 100 до 200, — самые неинтересные из обитателей нашего «электронного океана». В общем-то, это обычные числа, которые можно делить, умножать, складывать, записывать в регистры, но которые не выводятся на индикатор в силу своей чрезмерной величины. Изловить ЕГГОГа проще простого: достаточно, например, отдать команду 1 ВП 50 Fx² ПО Сх, и ЕГГОГ (десять в сотой степени) сидит в регистре О! Если теперь разделить его, допустим, на 10, то на индикаторе появится совершенно обыденная единица с порядком 99.

3-й этаж. Если возвести ЕГГОГа из предыдущего примера в квадрат (или иным способом получить число с показателем степени между 200 и 300), на индикаторе появится ЗГГОГ. Эти числа также можно умножать, складывать, записывать в регистры и так далее. Однако, помимо этого, ЗГГОГ обладает целым рядом присущих только ему и весьма полезных свойств.

1) Десятичная точка при появлении на индикаторе ЗГГОГа сохраняет свое положение, как бы «наследу-



ет» его от предыдущего числа. Запишите какого-нибудь ЗГГОГа в произвольный регистр. Наберите на индикаторе любое число (в его состав, естественно, обязательно входит десятичная точка — если число целое, она его замыкает) и вызовите ЗГГОГ на индикатор. Точка осталась на прежнем месте. Это свойство позволяет использовать ЗГГОГов в электронных играх для визуальной индикации положения объекта.

2) Всякий ЗГГОГ выполняет операцию безусловного перехода на адрес, совпадающий с первыми двумя цифрами порядка «зашифрованного» под ним числа. Так, полученный нами ЗГГОГ равен 10 в двухсотой степени; если при его появлении на индикаторе отдать команду FПРГ, убедимся, что справа горит 20.

3) Всякого ЗГГОГа, появившегося на индикаторе, легко «расшифровать» с помощью следующей процедуры: нажать FАВТ, затем десятичную точку — справа

на индикаторе загорится трехзначный порядок числа, которое прячется под личной ЗГГОГа. Снова нажмите FАВТ — слева на индикаторе появится мантисса числа, справа — некий новый показатель, весьма причудливый, зависящий от способа появления данного ЗГГОГа на индикаторе и для дешифровщика бесполезный. Применение этой процедуры к нашему ЗГГОГу дает порядок 200 и мантиссу 1, как, очевидно, и должно быть.

4) Предыдущее свойство подсказывает эффективный прием формирования показательных видеосообщений (чисел с буквенными порядками). Вызвав нашего ЗГГОГа из регистра, куда он был записан, и применив к нему процедуру «расшифровки», получим показатель — L. Если теперь отдать команду ВП 99 FАВТ, появится еще одно новое показательное сообщение (справа на индикаторе горит «чистая» буква E). Из этих двух сообщений с помощью команд ВП /—/ 1 и ВП /—/

10 легко получить все остальные мыслимые показательные шифры.

5) ЗГГОГ, записанный в регистр 9 либо 0, может использоваться как анализатор состояния программного счетчика. Убрав ЗГГОГа с индикатора, отдайте, например, команду БП 58. Вызовите ЗГГОГа и нажмите десятичную точку. Справа на индикаторе загорится 580. Данное свойство ЗГГОГа позволяет использовать его для «дешифровки» некоторых других «чудовищ», населяющих глубинные этажи нашего «числового моря».

4-й этаж. Перейдем к «охоте» на глубинах 300—400. Выберем в качестве объекта, например, число 10 в трехсотой степени. Отдаем команды 1 ВП 50 Fx² Fx² П9 (записываем ЗГГОГа для последующего использования в качестве анализатора) FВх. Все готово: в регистре Y сидит ЗГГОГ (10 в двухсотой степени), в регистре X — ЕГГОГ (10 в сотой степени). Остается их перемножить...

Караул! На экране мелькают цифры — ПМК самопроизвольно перешел в режим счета! Чудовище вырвалось на свободу и мчится по нашей пустой программе, как по бесконечному коридору! Срочно нажимаем С/П. На индикаторе ноль. Это естественно — программа пуста, она состоит из нулей, вот ноль и считался в регистре X, оттеснив чудовище в регистр Y. Чтобы взглянуть на «добычу», нужно нажать XY...

Нас ждет новое потрясение! Вместо ожидаемого чудовища мы видим перед собой лишь следы его деятельности — испорченный фрагмент программы. ПМК самопроизвольно перешел в режим программирования! Слева на индикаторе горит 0, затем две пары 00, в пра-

вом углу — 31. Значит, программа остановилась на адресе 30. По аналогии со ЗГГОГами заключаем, что это опять-таки первые две цифры порядка изловленного числа. Точка, как и у ЗГГОГа, унаследовала свое положение от предыдущего числа (только что на индикаторе горел ноль, естественно, с точкой). Наконец, левый ноль — это вторая цифра порядка (300). Если бы порядок был, скажем, 384, то слева на индикаторе горело бы .8, справа — 39.

Что делать дальше? Грубейшей ошибкой будет естественное FАВТ — ПМК зациклится на поврежденной команде и не ответит ни на один приказ с пульта, придется его отключить. Нажимаем FПРГ. Точка исчезает. Теперь ШГ влево. Какой командой заменить испорченную? Наша задача — поймать чудовище, поэтому впишем сюда, например, ПА. Затем Сх (чтобы очистить стек) и С/П. Вот теперь можно и FАВТ. На индикаторе тут же загорается 0 — стек чист, а чудовище сидит в регистре А! Самое время проанализировать его с помощью ЗГГОГа из регистра 9. Трижды нажимаем ШГ влево (для компенсации вписанных в программу команд), ИП9, точку (на индикаторе появляется порядок 300) и FАВТ (слева загорается мантисса 1). Забив на всякий случай нулями вписанные в программу команды, можно начинать «охоту» на следующее чудовище (только не надо забывать, что первое все еще томится в регистре А, ожидая команды ИПА, чтобы оттуда вырваться!). Вся эта процедура может пригодиться и для получения совершенно конкретных практических результатов. Например, она позволяет определять факториалы чисел вплоть до

210 (воспользуйтесь любой программой, вычисляющей факториал, и проанализируйте результат с помощью ЗГГОГа из регистра 9).

5-й этаж. Для охоты в этом регионе нашего «числового моря», в таинственном мире ОС-оборотней (числа с порядками между 400 и 500), полезно обзавестись подходящим «водолазным снаряжением». Введите в ПМК такую, например, программу: 00.КНОП (кстати, команды К1 и К2 ничуть не хуже выполняют функции «пустой» команды, хотя в инструкции о них и не говорится) 01.1 02.ВП 03.5 04.0 05.Fx² 06.Fx² 07.Fx² 08.× 09.ПА 10.0 11.× 12.С/П. Она умножает набранное нами число на 10⁴⁰⁰, формируя «чудовище», включает его в «клетку» — регистр А (можно использовать и любой другой) — и уничтожает все его следы в стеке. Легко видеть, что, подавая на вход различные числа с положительными порядками, мы перекрываем весь диапазон ОС-оборотней. Начнем охоту с самого «меньшего» — 10⁴⁰⁰. Команда: 1 В/О С/П. На индикаторе ноль, но оборотень в «клетке»! Не торопитесь выпускать его на свободу — просмотрите содержимое остальных регистров. Все спокойно, нигде ничего нет. Теперь ИПА. На индикаторе по-прежнему ноль. Охота, судя по всему, не удалась... Но не спешите с выводами — загляните в регистр С.ИПС. На индикаторе — предиковиннейшее создание, «хвост оборотня» (20,000000Е). Избавляемся от порядка: ВП 7 КНОП. Перед нами 2Е, причем двойка занимает «законное» место знака «минус». Если нажать клавишу (—), она сменится девяткой. Прделаем операцию 0 ПС ИПС. На индикаторе, естественно, ноль.

А что, если опять заглянуть в регистр А? ИПА ИПС. В регистре С вновь появился «хвост оборотня»!

Мы познакомились с главным свойством ОС-оборотней: при всяком их вызове в регистр Х на индикаторе появляется ноль, зато в регистр С записывается «хвост», вид которого зависит от величины оборотня. Если в качестве «клетки» использовать сам регистр С (заменить в нашей «водолазной» программе команду 09.ПА на 09.ПС), то при первом ИПС на индикаторе появится ноль, при втором — «хвост» оборотня, а сам он безвозвратно исчезнет.

Второе важное свойство ОС-оборотней — их этаж в искаженном виде копирует структуру всего «числового океана». При вводе в нашу программу чисел от 1 до 9,9999999 ВП 9 включительно в регистре С появляются «числа-мутанты», начинающиеся какой-либо цифрой на месте минуса (она на единичку больше старшей цифры введенной мантиссы; если мантисса начинается с девятки, здесь стоит просто минус) и заканчивающиеся буквой Е, затем — мутантные формы ЕГГОГов (при входных числах вплоть до 9,9999999 ВП 19), ЗГГОГов (при входных числах до 9,9999999 ВП 29; можете сами их исследовать на предмет отличия от обычных ЗГГОГов; процедура их «расшифровки» приводит к показательным шифрам с довольно интересными основаниями), затем знакомых уже нам диких чудовищ 4-го этажа... Но самое любопытное начинается при вводе чисел от 1 ВП до 9,9999999 ВП 49 — при вызове оборотня из «клетки» в регистр С записываются опять-таки оборотни (назовем их оборотнями второго порядка)!

Образуем, например, чис-

ло, равное 10^{440} . 1 ВП 40 В/О С/П. На индикаторе ноль, но оборотень — в регистре А. ИПА. На индикаторе снова ноль, но теперь в регистр С записался оборотень второго порядка. ИПС — на индикаторе по-прежнему ноль, зато в регистр С, по идее, записался «хвост». ИПС — действительно, на индикаторе 00,0000ЕЕ. Если снова скомандовать ИПА, в регистр С опять запишется оборотень второго порядка, и команда ИПС — при первом нажатии — выдаст на индикатор ноль...

Но главный сюрприз впереди. Оказывается, уровень оборотней второго порядка также копирует структуру «числового океана». При вводе в нашу программу чисел от 1 ВП 40 до 9,9999999 ВП 40 команды ИПА ИПС ИПС приводят к «числам-мутантам», завершающимся ЕЕ, затем наступает очередь мутантных форм ЕГГОГов, ЗГГОГов («расшифровка» последних, кстати, приводит к показательным шифрам с ЕЕ в левой части), затем чудовищ 4-го этажа, а потом... мы вновь сталкиваемся с ОС-оборотнями, на этот раз уже третьего порядка!

Их «экологическая ниша» — это узкая щель между 10^{444} и 10^{445} (на вход нашей программы, стало быть, для их формирования нужно подавать числа от 1 ВП 44 до 9,9999999 ВП 44). «Хвост» оборотня третьего порядка представляет собой «число-мутант», завершающееся комбинацией ЕЕЕ, этакое «длинношее» (к счастью, дальнейшего копирования структуры нашего «океана» не происходит, иначе нам пришлось бы заниматься ОС-оборотнями до бесконечности), появляющееся на индикаторе лишь после третьей команды

ИПС. Пример: 1 ВП 44 В/О С/П ИПА ИПС ИПС ИПС (на индикаторе мантисса 0,0000ЕЕЕ и порядок 10).

Но пора и остановиться. Дальнейшее увеличение вводимых в программу чисел переносит нас в искаженные ОС-оборотнями миры еще незнакомых нам глубоководных созданий, в первую очередь Тьмы, встреча с которой небезопасна (индикатор гаснет, и приходится временно отключать ПМК). Отметим, что применение ОС-оборотней в игровых программах довольно перспективно: их можно использовать для получения различных наглядных шифров, временного либо постоянного зануления регистра С, а также в некоторых других целях. С соответствующими примерами мы еще встретимся.

6-й этаж. Самым, пожалуй, неприятным обитателем глубин нашего «числового океана» является Тьма — при любом контакте с ней индикатор гаснет. Основные владения Тьмы располагаются между порядками 500 и 600 (таким образом, всякое число от 10^{500} до 10^{600} — это Тьма). Для первого знакомства с ней можно в режиме АВТ набрать на клавиатуре такую, например, последовательность команд: 1 ВП 70 Fx² Fx² Fx². Индикатор гаснет — наши действия привели к числу 10^{560} , а это, конечно же, Тьма. Легко убедиться, что ПМК не отзывается теперь ни на один приказ с пульта. Однако если его выключить на несколько секунд, а затем включить снова, он будет работать как ни в чем не бывало.

Чтобы упрятать Тьму в «клетку» (адресуемый регистр), можно воспользоваться простой программой: 00.Fx² 01.Fx² 02.Fx² 03.ПА 04.Сх 05.С/П. Команда:

FAVT В/О 1 ВП 70 С/П. После останова на индикаторе горит ноль, но в регистре А сидит Тьма! Если вы рискнете и выпустите ее оттуда (ИПА), то индикатор погаснет, придется отключить калькулятор и ввести программу снова.

Как вы помните, для количественного анализа чудовищ 4-го этажа использовался ЗГГОГ из регистра 9. Однако для расшифровки как ОС-оборотней, так и Тьмы такой анализатор непригоден. Чтобы дешифровать Тьму (да и любые другие «суперчисла»), полезен логарифмический анализатор: 00.Fx² 01.Fx² 02.Fx² 03.Flг 04.1 05.0 06.0 07.0 08.— 09./—/ 10.П9 11.КИП9 12.ХУ 13.ИП9 14.— 15.FVх 16.ХУ 17.F10^x 18.С/П (стрелки в командах ХУ по техническим соображениям опущены). Программа логарифмирует сформированное командами 00—02 «чудовище» и вычисляет его мантиссу и порядок, так что после останова в регистре Х оказывается мантисса (с небольшой ошибкой в последних десятичных знаках), в регистре У — порядок. Обратите внимание на фрагмент 04—09 — вычисленный логарифм «сверхчисла» вычитается из тысячи; легко убедиться, что такая коррекция необходима для всех обитателей «числового океана», вплоть до Нуля (10^{800}). Фрагмент 10—13 использует для выделения целой части числа команду косвенного вызова; это наиболее простой путь выполнения такой операции на «Электронике БЗ-34» (в новых моделях ПМК для нее предусмотрена специальная команда).

Испробуем наш анализатор на Тьме: В/О 1 ВП 70 С/П. После возведения в восьмую степень должно, очевидно, получиться число



10^{560} . На индикаторе загорается приближенное значение мантиссы (1,0002303), в регистре Y оказывается совершенно правильная величина порядка (560).

Можно ли вызвать Тьму в регистр X? Казалось бы, странный вопрос... Но введите в ПМК программу: 00.Fx² 01.Fx² 02.Fx² 03.K— (подойдет и любая другая «неправильная» команда, начинающаяся с K). Перейдите в режим АВТ и скомандуйте: В/О 1 ВП 70 С/П. На индикаторе загорается

сообщение ЕГГОГ (результат «неправильной» команды), но под ним скрывается Тьма — если отдать сейчас одну из команд КНОП, К1, К2, стрелка вверх (ввод в стек) или FАВТ, индикатор погаснет. Тьма, «замаскированная» сообщением ЕГГОГ, находится в регистре X, и с нею можно обращаться как с любым нормальным числом — умножить на что-нибудь, разделить, прологарифмировать вручную, используя приведенную выше процедуру... А

что, если попробовать вычислить число, обратное Тьме? Команда: F 1/x. На индикаторе — ноль. Казалось бы, ничего удивительного — что же еще могло получиться в результате такой операции? Однако не будем спешить с выводами, заглянем в регистр С. ИПС ИПС. На индикаторе — уже знакомый нам «хвост» (00,0000ЕЕ) оборотня, равного 10^{440} . Итак, разделив единицу на 10^{560} , мы получим 10^{440} ; впрочем, если вспомнить, что наш «чис-

ловой океан» характеризуется периодом в 1000 по величине порядков, в этом опять-таки нет ничего удивительного: единица в «арифметике» ПМК тождественно равна 10^{1000} (вспомните коррекцию логарифма, о которой только что шла речь). Отсюда следует важный вывод: числа, обратные Тьме, это ОС-оборотни; следовательно, числа, обратные ОС-оборотням, — это Тьма; значит, во избежание неприятностей не стоит производить над ОС-оборотнями такой операции... Кроме того, возникает подозрение, что в наш «числовой океан» можно проникнуть и с «черного хода» — через числа с отрицательными порядками; это действительно так.

Кроме своего «законного» этажа, Тьма занимает и две «ниши» в мире ОС-оборотней: от 10^{450} до 10^{470} (оборотни первого порядка) и от 10^{445} до 10^{447} (оборотни второго порядка); легко видеть, что в этих мирах Тьма «оккупирует» еще и соседний этаж, где, по идее, должны были бы располагаться С-ЕГГОГи (числа с порядками между 600 и 700), с которыми мы еще познакомимся. Отдайте, например, такую команду (в ПМК введена последняя из приведенных программ, завершающаяся К—): В/О 1 ВП 58 С/П. На индикаторе — сообщение ЕГГОГ, под ним скрывается ОС-оборотень, равный 10^{464} . Нажимаем КНОП, на индикаторе гаснет, Тьма...

Казалось бы, Тьму можно использовать лишь в электронных играх со «смертельным исходом»: записав ее в регистр, нетрудно добиться того, чтобы при ошибке со стороны играющего пришлось бы вводить программу заново. Однако это далеко не так, практиче-

ские применения Тьмы гораздо шире. Они связаны с «тайными адресами» программной памяти «Электроники».

Работа наших ПМК характеризуется 160-шаговым циклом (к сожалению, инструкция о нем не упоминает): если в программе нет переходов, выполняются сначала команды, записанные по адресам 00—97 (главная ветвь), затем по адресам 00—13 (короткая побочная ветвь), потом по адресам 00—47 (длинная побочная ветвь), после чего управление вновь передается на начало главной ветви программы. Побочные ветви имеют собственную систему адресации: в короткой ветви адресам 00, 01 и т. д. соответствует 98, 99, А0... А9, В0, В1; в длинной — В2... В9, С0... С9, Д0... Д9, Е0... Е9, 0... 9. Букве Е на клавиатуре соответствует стрелка вверх (ввод в стек); «пустышке», стоящей на первом месте в последней десятке адресов, соответствия нет — команды переходов по ним можно записать в программу лишь с помощью довольно «хитрых» искусственных приемов. В новых ПМК («МК-61» и «МК-52») главная ветвь простирается до адреса А4, поэтому первая побочная ветвь укорочена. Начиная с адреса С0, в длинной побочной ветви начинается «темная зона»: коды команд, записанных по соответствующим адресам главной ветви, при выходе в режим ПРГ на индикатор не выводятся, однако в режиме счета эти команды исправно выполняются. Побочные ветви 160-шагового цикла можно использовать на практике для весьма замысловатых операций. Например, при безусловном переходе на адрес Е9 выполняются сначала действия, записанные по адресам 37—

47, затем, без всякой дополнительной команды, произойдет возврат на адрес 00 главной ветви.

Таким образом, в 160-шаговый цикл входят все адреса, заканчивающиеся на какую-либо цифру. А куда передается управление при команде перехода на адрес, завершающийся буквенным символом?

Отдайте в режиме АВТ, например, команду БП ОА и посмотрите, куда передалось управление ФПРГ. Знакомая картина, индикатор гаснет, мы столкнулись с Тьмой... Отключите ПМК на несколько секунд, включите его снова и повторите эксперимент: БП ОА. Только теперь перед ФПРГ нажмите ШГ влево и ШГ вправо. Казалось бы, ничто не должно измениться... ФПРГ. Справа горит 10, следовательно, управление передалось на этот адрес! Таким образом, «явному» адресу 10 соответствует «тайный» ОА; самое важное — при работе по программе команды переходов по ним дают совершенно тождественные результаты.

«Тайные» двойники есть у многих адресов главной и побочных ветвей 160-шагового цикла. Адресу 11 соответствует ОВ, 12 — ОС, 13 — ОД, 14 — ОЕ, 20 — 1А и так далее (рекомендуем составить для себя табличку «тайных» и «явных» адресов всех ветвей цикла). Это позволяет, в частности, использовать хранящиеся в регистрах буквенные сообщения в качестве адресов перехода при косвенной адресации. Сформируйте, например, символ Е и зашли-те его в регистры 9 и 0: 1 К— ВП П9 ПО. Отдайте теперь команду КБП9. Управление должно перейти на адрес, хранящийся в регистре 9, а там у нас находится символ Е! Куда перейдет управление? Логично предпо-

ложить, что на «тайный» адрес 0Е, которому, как мы уже знаем, соответствует «явный» адрес 14 основной ветви программы. Действительно, если сделать ШГ влево, ШГ вправо и перейти в режим ПРГ, убедимся, что это так. А что произойдет при команде КБПО? Адрес, хранящийся в регистре 0, модифицируется (уменьшится на единичку), и управление перейдет на модифицированный адрес. Легко убедиться, что «модификация» в данном случае — это преобразование символа Е в Г, а управление передается на «тайный» адрес 0Д, которому соответствует «явный» 13... (К слову сказать, буквенные символы могут использоваться и при командах косвенного вызова и косвенной записи. Если сейчас, например, отдать команду КИПО, Г преобразуется в L, а на индикаторе появится содержимое регистра В.)

«Тайные» адреса, как и «явные», можно использовать и в качестве кодов команд. Одна из игровых программ, опубликованных в журнале «Техника — молодежи», содержала довольно любопытный блок выдачи видеосообщений: 60.Fx < 0 61.61 62.Fx ≥ 0 63.63 64.С/П. Как он работает? Находящееся в регистре X число сравнивается с нулем и, если условие по адресу 60 не выполняется, управление вторично передается на адрес 61. Эта комбинация цифр воспринимается теперь как код команды ИП1, и вызывается одно из двух видеосообщений, характеризующих складывающуюся в игре ситуацию. Оно, в свою очередь, сравнивается с нулем, благополучно проходит эту проверку (оба видеосообщения воспринимаются как положительные числа), и управление передается на

команду останова С/П. Если же входное число меньше нуля, то оно переправляется на вторую проверку, не выдерживает ее, управление передается на адрес 63, эта комбинация воспринимается как код команды ИПЗ, вызывается второе сообщение и происходит программный останов. В результате экономятся две ячейки программной памяти: оба адреса перехода служат одновременно и командами вызова. Достаточно очевидно, что наличие «тайных» адресов расширяет возможности использования рассмотренного приема.

Как видим, охота на Тьму завершилась успешно: трофеи взяты немалые. Можно отправляться дальше.

7-й этаж. Прежде чем вступить в контакт с его обитателями, ознакомимся с одной особенностью «БЗ-34» («МК-54»). Вопреки тому, что утверждается в заводской инструкции, адресуемых регистров в памяти этого ПМК вовсе не 14, а 15 — имеется еще один буквенный регистр Е: на клавиатуре ему соответствует стрелка вверх (ввод в стек). Нетрудно проверить, что по этому регистру можно осуществить полный набор команд (прямые запись и вызов, а также косвенные: запись, вызов, обращение к подпрограмме и переходы — четыре условных и один безусловный). Эти команды имеют собственные коды (все они завершаются буквой Е) и исправно выполняются как при ручных вычислениях, так и при расчетах по программе. Лишь одна особенность отличает регистр Е от остальных: он постоянно связан с регистром 0! Иными словами, содержимое обоих регистров всегда совпадает.

Казалось бы, что толку от такого дополнительного регистра? Какая разница, 0

или Е, если числа в них все равно одинаковы? Действительно, команды прямой записи и прямого вызова по этим регистрам, несмотря на то, что коды их отличаются (40 и 4Е, 60 и 6Е), абсолютно взаимозаменяемы. А вот при косвенном обращении к регистру Е (как и к другим буквенным, а также «старшим» цифровым регистрам 7, 8 и 9) не происходит «модификации» находящегося в нем числа — оно, попросту говоря, не меняется; соответственно остается прежним и содержимое регистра 0. При косвенном же обращении к регистру 0 его содержимое «модифицируется» (уменьшается на единичку, как и в случае регистров 1, 2 и 3) — соответственно меняется и число в регистре Е. Эту постоянную связь удобно использовать в циклах по регистру 0, с этим мы еще встретимся. К слову сказать, в «МК-61» и «МК-52» связь Е—0 разорвана, поэтому обладателям этих ПМК придется в некоторых ситуациях искусственно ее вводить, что потребует минимум двух команд ИПО ПЕ (а чаще трех: ИПО ПЕ F0). Кроме того, придется учитывать сдвиг программы и менять некоторые адреса переходов.

Столь обширное отступление потребовалось потому, что обитатели 7-го этажа «электронного океана», С-ЕГГОГи, позволяют устанавливать подобную (правда, одностороннюю) связь между регистром С и любым другим. Введите в регистр С какое-нибудь число, например, 22, перейдите в режим ПРГ, наберите стандартную «водолазную» программу: 00.Fx² 01.Fx² 02.Fx² 03.ПА 04.Сх 05.С/П, вернитесь в режим АВТ и скомануйте, допустим, 1 ВП 80 В/О С/П. На индикаторе

ноль, но в регистре А сидит С-ЕГГОГ. ИПА. На индикаторе — 22, содержимое регистра С! Это главное свойство «сверхчисел» с порядками между 600 и 700 (сейчас в регистре А записано 10^{640}) — при их вызове в регистр Х сами они тут же «отстают» в регистр Y, вытаскивая на индикатор число из регистра С (почему именно этому буквенному регистру такое предпочтение, никому не известно). С числом, которое горит сейчас на индикаторе, можно осуществлять различные операции. Например, $2 \times /44/ 4 \div /11/ 1 - /10/ 15 + /25/ FV /5/ F1/\times /0,2/$ и т. д. Но в регистре Y по-прежнему находится «сверхчисло». Попробуем XY. На индикаторе вновь появляется 22 — «чудовище», вызванное в регистр X, незамедлительно отползло в свою «пещеру» (регистр Y), прикрывшись «добычей» (содержимым регистра С)...

Если нажать теперь знак сложения, после томительной паузы на индикаторе появится сообщение ЕГГОГ. «Сверхчисло», замаскированное под ним, собственной персоной явилось на индикаторе! Это легко проверить, например, отдав команду Flg: на экранчике загорится 359,99998 — логарифм «сверхчисла» (с учетом периода в 1000 по величине порядков).

С-ЕГГОГи обладают и многими другими, еще не вполне понятными свойствами. Например, их можно использовать для дешифровки других «сверхчисел». Если извлечь из регистров А или С заранее записанного туда ЕГГОГа, а сразу за ним — С-ЕГГОГ (из другого регистра), то на индикаторе загорится расшифровка числа в регистре С. Если теперь отдать команду /—/, то на индикаторе

появится расшифровка содержимого регистра 0. Использование же этих «суперчисел» в электронных играх представляется проблематичным.

8-й этаж. То же самое можно сказать и о числах с порядками от 700 до 800. Для знакомства с ними пригодится ЗГГОГ-анализатор: 1 ВП 50 Fx² Fx² П9 Сх. Подадим на вход нашей «водолазной» программы, допустим, 1 ВП 90 В/О С/П. На индикаторе 0. ИПА. На экранчике появляется нечто несообразное (00,10000000 2). Это «длинный монстр», типичный обитатель данного этажа. Справиться с ним нетрудно: FАВТ ИП9 ИП9. На индикаторе — ЗГГОГ-анализатор. Нажимаем десятичную точку. Справа загорается трехзначный порядок — 720; нажимаем FАВТ — слева появляется мантисса 1. ЗГГОГ, как всегда, не подвел (кстати, при некотором навыке расшифровать «длинного монстра» легко по его внешнему виду; предлагаем в этом потренироваться самостоятельно). На этот этаж можно проникнуть и без «водолазной» программы, «вплыв» (то есть в режиме АВТ), со стороны чисел с отрицательными порядками. Например, так: 0,01 (количество нулей после запятой может быть произвольным) ВП /—/ 99 Fx². На индикаторе — 80,10000000 9.

9-й этаж. Эта область (порядки между 800 и 900, а также между —100 и —200) безраздельно принадлежит Нулю. Проверьте это сами. Для электронных игр наиболее интересны его «воплощения» в мире ОС-оборотней (числа с порядками между 480 и 490, а также между 448 и 449). Записав такое число, допустим, в регистр А, получаем возможность обнулять

регистр С одной-единственной командой ИПА. Например, сейчас в регистре С записано 22. Подадим на вход «водолазной» программы число 1 ВП 60 В/О С/П. На индикаторе 0. ИПС /22/ ИПА /0/ ИПС/0/. Легко убедиться, что такое зануление исправно выполняется и при расчетах по программе. В результате появляется возможность сэкономить одну команду — практика показывает, что именно ее-то очень часто и не хватает.

*Игру придумал и описал
М. Пухов*

Школа гравилетчиков

Универсальные ЭВМ (в том числе и наши ПМК) позволяют строить математические модели не только существующих, но и фантастических механизмов и ситуаций. В частности, перенестись в систему нейтронной звезды с труднопроизносимым названием и вместе с героями рассказа совершить полет на инопланетном летательном аппарате с гравитационным двигателем. Соответствующая игровая программа называется «Гравилет»:

00.БП 01.01 02.ИПА
03.ИП8 04.÷05.+06.П9
07.ИПС 08.ИПД 09.× 10.
ИП5 11.÷ 12.ИПА 13.ИПД
14.— 15.КИП9 (КНОП) 16.
С/П 17.П9 18.Fx² 19.F10²⁰..
21.ИП4 22.× 23.ИПО 24.
ИПА 25.× 26.П7 27.÷28.
ИП9 29.ИПА 30.÷ 31.↑ 32.
Fx² 33.ИПВ 34.× 35.—
36.ИПО 37.× 38.ПА 39.ИП4
40.ИПД 41.÷ 42.FV 43.ИПД
44.÷ 45.ИП9 46.× 47.—
48.ИП5 49.× 50.ИПС 51.+
52.1 53.8 54.0 55.— 56.Fx<0
57.64 58.FVx 59.+ 60.FVx



61.+ 62.Fx \geq 0 63.58 64.FBx
 65.— 66.ПС 67.Fo 68.ИПО
 69.— 70.П9 71.ИПА 72.Fcos
 73.× 74.— 75.ИПВ 76.ИПА
 77.Fsin 78.ПА 79.× 80.—
 81.ПО 82.. 83.Fcos 84.ИПВ
 85.× 86.ИП9 87.ИПА 88.×
 89.— 90.ПВ 91.ИП7 92.ИПО
 93.÷ 94.ПА 95.ИП6 96.—
 97.FV

По адресам 20 и 82 записана десятичная точка (код 0—), по адресу 31— стрелка вверх (OE), по адресу 67—

круговое перемещение стека (25), по адресам 42 и 97— извлечение квадратного корня (21). После ввода программы надо сформировать и заслать в регистры 1—3 наглядные видеосообщения о положении гравилета в пространстве. Например, такие : 1 ВП 50 Fx² Fx² П9 Сх ИП9 FАВТ точка К— ВП /—/ 54 П1 (слева Е, справа минус; это значит «Гравилет вблизи звезды») ВП /—/ 10П2 (правее мину-

са появился нолик; это означает «Гравилет на подходе к астероиду») ВП 9 П3 (минус и нолик поменялись местами; «Гравилет за орбитой астероида»). Минус на всех изображениях означает космический аппарат, буква Е — звезду, 0 — астероид. Конечно, ничто не мешает использовать любые другие видеосообщения; скажем, если в приведенной последовательности заменить К— на КНОП, звезда изобразится единичкой, а если эту команду вообще опустить — тройкой.

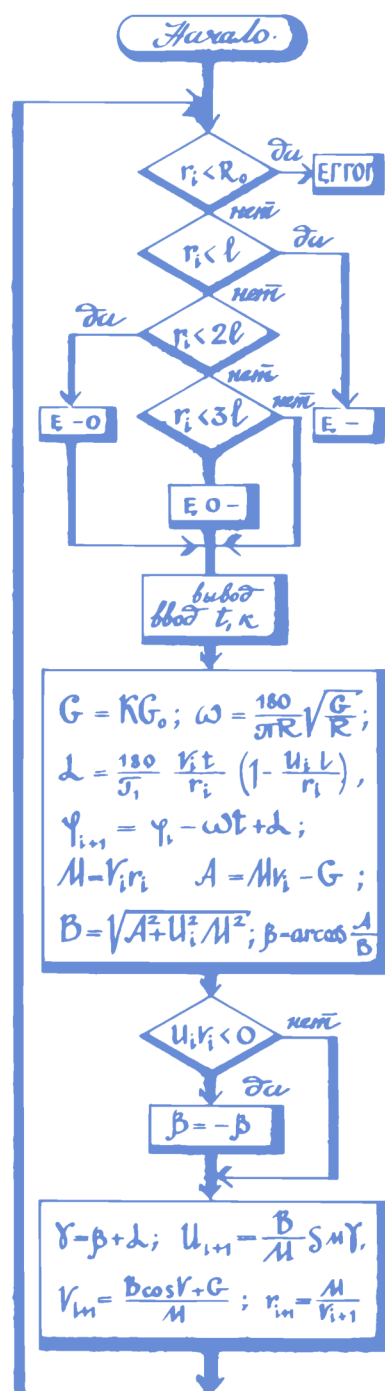
С помощью программы «Гравилет» можно осуществлять самые замысловатые маневры фантастического космического аппарата, оснащенного гравитационным двигателем, в поле тяготения массивного центрального тела (звезды или планеты), причем координаты аппарата отсчитываются относительно другого тела (астероида, спутника либо космической станции), притяжением которого можно пренебречь и которое обращается вокруг первого по круговой орбите. Установленный на аппарате гравитационный двигатель позволяет как угодно менять его гравитационную массу (делать ее положительной, отрицательной или нулевой), в то время как инертная остается постоянной. Легко видеть, что ничего другого от идеального гравитационного двигателя и не требуется. Даже знаменитый уэллсовский «кейворит» таких возможностей не обеспечивал.

Чтобы работать с программой, надо задать числовые величины. В регистр 5 засылается количество градусов в радиане: 180 FП÷ (57,29578) П5. Остальные константы характеризуют планетную систему, куда нас занесло: (радиус звезды R₀

в м) П6 (гравитационная постоянная звезды, равная произведению ускорения силы тяжести на ее поверхность на квадрат радиуса, в $\text{м}^3/\text{с}^2$) П4 (радиус орбиты астероида R в м) ПД (масштаб l, определяющий смену видеосообщений, в м) П8. Удобно задавать в качестве масштаба половину радиуса орбиты: ИПД $2 \div$ П8; тогда на меньших расстояниях от звезды будет выдаваться первое видеосообщение, затем второе, а при выходе за пределы орбиты — третье.

Остается ввести начальные координаты и скорости самого гравилета: (расстояние от центра звезды r в м) ПА (угол отклонения ϕ от прямой, соединяющей звезду и астероид, в градусах) ПС (вертикальная, точнее, радиальная скорость U в м/с) ПВ (горизонтальная, точнее, трансверсальная скорость V в м/с) ПО. Трансверсальная скорость задается в неподвижной системе координат. Положительным для нее и угловой координаты считается направление по часовой стрелке.

Переключатель Р—Г при полетах на гравилете устанавливается в положение Р. Работа с программой начинается командой В/О С/П. При останове на индикаторе появляется видеосообщение, показывающее, где в данный момент находится гравилет. В регистр Y выводится его расстояние от орбиты астероида, оно вызывается командой XY. А в регистре Z находится расстояние до астероида вдоль орбиты, чтобы вывести эту величину на индикатор, надо два раза нажать F0. При выходе корабля за пределы области, перекрываемой видеосообщениями (когда r больше $3l$), видеосообщение не выводится, на индикаторе оказыва-



ется содержимое регистров 4, 5, 6 и т. д. Наконец, при маневрировании вблизи базы разумно пожертвовать видеосообщением, заменив команду 15.КИП9 на 15.КНОП. В этом случае при останове на индикаторе будет расстояние до орбиты астероида, в регистре Y — расстояние до него вдоль орбиты.

Если корабль в результате неудачного маневра вонзается в звезду, на индикаторе появляется ЕГГОГ. Для задания ма-

невра надо набрать время в секундах, нажать ПП, набрать требуемый коэффициент гравитации K и нажать С/П. Если $K=1$, то корабль заданное время летит по обычной гиперболической или эллиптической орбите. Если $K=0$, гравитация на него не действует, и он летит по прямой (конечно, в системе координат, связанной с движущимся астероидом, она выглядит очень даже кривой). При отрицательных значениях K притяжение сменяется отталкиванием. С увеличением абсолютной величины K сила притяжения (отталкивания) возрастает в соответствующее число раз. Коэффициент K при всех маневрах не должен превышать по абсолютной величине 10 — программой предусмотрена соответствующая блокировка, на индикаторе вспыхивает ЕГГОГ (и всемогущество сверхцивилизаций имеет пределы).

Займемся теперь делом: надо помочь нашим героям. Регистры 1, 2, 3 и 5 заполняем согласно инструкции. Радиус нейтронной звезды известен: 1 ВП 4 П6. Радиус орбиты астероида и масштабный коэффициент тоже: 1 ВП 9 ПД ПА 2 ÷ П8 (собственными размерами астероида пренебрегаем). Начальное угловое смещение и радиальная скорость равны нулю: Сх ПВ ПС. Осталась гравитационная постоянная нейтронной звезды. Ее масса, по утверждению рассказчика, «почти равна солнечной», а гравитационная постоянная нашего светила — примерно 1,34 ВП 20 (кстати, для Земли этот параметр равен приблизительно 4 ВП 14, для Марса — 4,3 ВП 13). Примем 1 ВП 20 П4. Начальная трансверсальная (направленная вдоль орбиты) компонента скорости

гравилета равна скорости астероида; чтобы ее определить, нужно разделить гравитационную постоянную на радиус орбиты и из результата извлечь квадратный корень. $IP4 \text{ ИПД} \div F\sqrt{PO}$. Получилось, как видим, больше 300 км/с.

Исходные данные введены. В/О С/П (Е О—) ХУ (О) ИПС (О) ИПВ (О). Гравилет готов к старту. 60 ПП О С/П. Через минуту он оказывается на высоте чуть меньше 180 км. Повторяем ту же команду еще четыре раза. На борту гравилета прошло пять минут, у нас — примерно столько же. В результате и мы и они оказались на расстоянии 4,5 тыс. км за орбитой астероида и отстаем от него почти на 300 км.

Первое задание — вернуться на астероид (то есть в область радиусом порядка 1000 м от начала координат). На меньших расстояниях возможны сюрпризы — нормальному полету начинают мешать охраняющие базу роботы (точность вычислений оказывается недостаточной). Заходить на посадку надо «сверху», с внешней стороны орбиты (база расположена, как мы помним, именно там). Готовых рецептов по управлению гравилетом у нас нет; машина инопланетная, никто пока на таких не летал, так что все находится в равных условиях: и вы, и герои рассказа.

После возвращения на базу можно приступать ко второму заданию: стартовать, приблизиться к нейтронной звезде, обследовать ее окрестности и вернуться на базу. А освоившись с управлением гравилетом, можно попробовать полетать и в нашей Солнечной системе.

Игру придумал и описал М. Пухов

Охота на скалоеда

На других планетах (это достоверно известно из произведений НФ) обитают чудовища, встреча с которыми небезопасна. Внешний вид и образ жизни инопланетных животных не имеют аналогов среди земных форм. Да и для охоты на них традиционные способы не годятся. Нужен как минимум ПМК.

Так называемый скалоед — это один из наиболее ярких представителей внеземной фауны. Он живет глубоко под поверхностью, но находится в постоянном движении, оставляя за собой причудливые переплетения проделанных в граните ходов. Отважный охотник, решившийся бросить чудовищу вызов, может воспользоваться этим лабиринтом, чтобы приблизиться к предполагаемой добыче вплотную.

Охота на скалоеда осложняется одним немаловажным обстоятельством — наткнувшись на свой старый ход, животное (а оно перемещается более-менее беспорядочно) заваливает его крупными обломками скал, делая лабиринт на данном участке непроходимым. Бывали даже случаи, когда охотник оказывался навсегда замурованным глубоко под землей.

Приведенных теоретических сведений достаточно, чтобы приступить к освоению игровой программы «Охота на скалоеда»:

00.ПД 01. $Fx=0$ 02.35
03.ИПО 04.ИПА 05.—
06. $Fx=0$ 07.32 08.Farctg
09.С/П 10.ИПС 11.ВП 12.5
13.Fsin 14.ПС 15.† 16. Fx^2
17. $F\sqrt{\quad}$ 18.÷ 19.В/О 20.Fo
21.+ 22.КПВ 23.ХУ
24.КИПД 25.+ 26.БП
27.93 28./—/ 29./—/ 30.БП
31.46 32.ИПО 33.ИПА

9	1	1	1	1	1	1	1	1
8	1	0	0	0	0	0	0	1
7	1	1	1	1	1	1	1	1
6	1	0	0	0	0	0	0	1
5	1	1	1	1	1	1	1	1
4	1	0	0	0	0	0	0	1
3	1	1	1	1	1	1	1	1
2	1	0	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1
	1	2	3	4	5	6	7	8

34.С/П 35.КППД 36.КППД
37. $Fx \geq 0$ 38.45 39.ХУ
40./—/ 41./—/ 42.ВП 43.1
44.ХУ 45.ХУ 46.† 47.КИПД
48.+ 49.50.ПВ 51.ВП 52.ПВ
53. $Fx=0$ 54.59 55.ИПД
56.÷ 57.БП 58.94 59.— 60.1
61.0 62.÷ 63.8 64.— 65./—/
66. $Fx \geq 0$ 67.55 68. $F10^x$
69.КИПВ 70.ХУ 71.ИПД
72. $Fx=0$ 73.20 74.+ 75.÷
76.ИП1 77.+ 78.FBx 79.—
80.† 81.† 82.8 83. $F10^x$ 84.+
85.FBx 86.— 87.— 88.1 89.—
90. $Fx \neq 0$ 91.23 92.К—
93.КПД 94.1 95.0 96.ИПД
97.—

Обладатели «МК-61» и «МК-52» должны вписать в конце программы еще две команды: 98.БП 99.00. После ее ввода нужно прежде всего вписать в регистры 1—9 план подземного лабиринта (см. рисунок). Нулями на схеме обозначена скальная порода, единичками — свободные ходы. Хотя дисплей наш ПМК пока что не оборудован, командами ИП1 ИП2... ИП9 нетрудно на любом этапе игры просматривать весь лабиринт (этот оригинальный способ задания игрового поля придумал Владимир Архипов из Москвы).. Нашему рисунку соответствует 11111111 П1 П3 П5 П7 П9 10000001 П2 П4 П6 П8. Лабиринт в принципе можно задавать любым, но в нижнем левом уг-



лу в начале игры обязательно должна размещаться единичка.

Теперь надо задать исходное положение охотника и его потенциальной жертвы. Координатная система такая же, как и в шахматных играх: местоположение человека и животного задается двузначными числами, первая цифра которых означает горизонтальную координату, вторая — вертикальную. Под текущие координаты охотника отведен регистр 0 («охотник»), чудовища — регистр А (по всей видимости, сокращение от английского слова «animal»). Для начала отведем скалоеду участок в центре лабиринта (он на рисунке выделен): 55 ПА, охотнику — левый нижний угол: 11 ПО. Осталось установить переключатель Р—Г в положение Г, ввести в регистр С произвольное дробное число (допустим, 0,1234567 ПС) — и можно начинать игру: Сх В/О С/П.

На индикаторе тут же загораются цифры 55 — координаты чудовища. В регистре Y—11, координаты охотника. Пора ему выступить в рискованное путешествие. Ход человека задается наглядно: БП ХУ (или стрелка вверх) + (или —) 1 С/П. Команда ХУ нужна для перемещения по горизонтальной координате, стрелка вверх — по вертикальной; «плюс» соответствует положительному перемещению, «минус» — отрицательному; единичка показывает, что переход совершается на соседнюю клеточку. А что будет, если попытаться выйти за пределы лабиринта? БП ХУ — 1 С/П. На экранчике ЕГГОГ, сюда путь закрыт. Точно так же среагирует ПМК и на любое другое недозволенное перемещение — охотник может ходить только по

единичкам. Сейчас, например, лишь вверх или вправо. Попробуем вправо: БП ХУ + 1 С/П. ПМК принимается за работу: регистрирует ход охотника и задает случайным образом ответ его предполагаемой жертвы. На индикаторе загорается 65 — зверь тоже пошел вправо. Посмотрим, как выглядит теперь тоннель, в котором он находится: ИП5 (1111211). Проход частично пришел в негодность — цифра 2 (а также 3 и т. д.) означает обвал, здесь теперь не пробраться. А вот если бы животное двинулось вверх (или вниз), оно проделало бы новый тоннель в скальной породе — на маршруте скалоеда нули превращаются в единицы, те в двойки и т. д.

Оставим охотника в размышлениях над следующим ходом — сейчас, очевидно, он может либо пойти дальше, на поле 31, либо вернуться на 11. А чем ответит чудовище, предсказать вообще невозможно. Задача охотника — оказаться со скалоедом в одной клеточке, на индикаторе при этом загорится 00 — сигнал победы. Если же лабиринт в конце концов окажется «перелопаченным» до такой степени, что на любую попытку хода ПМК будет реагировать сообщением ЕГГОГ, то останется лишь уповать на мастерство спасателей.

В программе «Охота на скалоеда» использованы некоторые особенности команд сложения, вычитания, ввода в стек и обмена ХУ. Нетрудно проверить, что, делая ход по горизонтали, мы оказываемся на адресах 40 или 41; ход по вертикали передает управление на адреса ЕО или Е1 «темной зоны»: здесь задублированы команды /—/, записанные по адресам 28 и 29

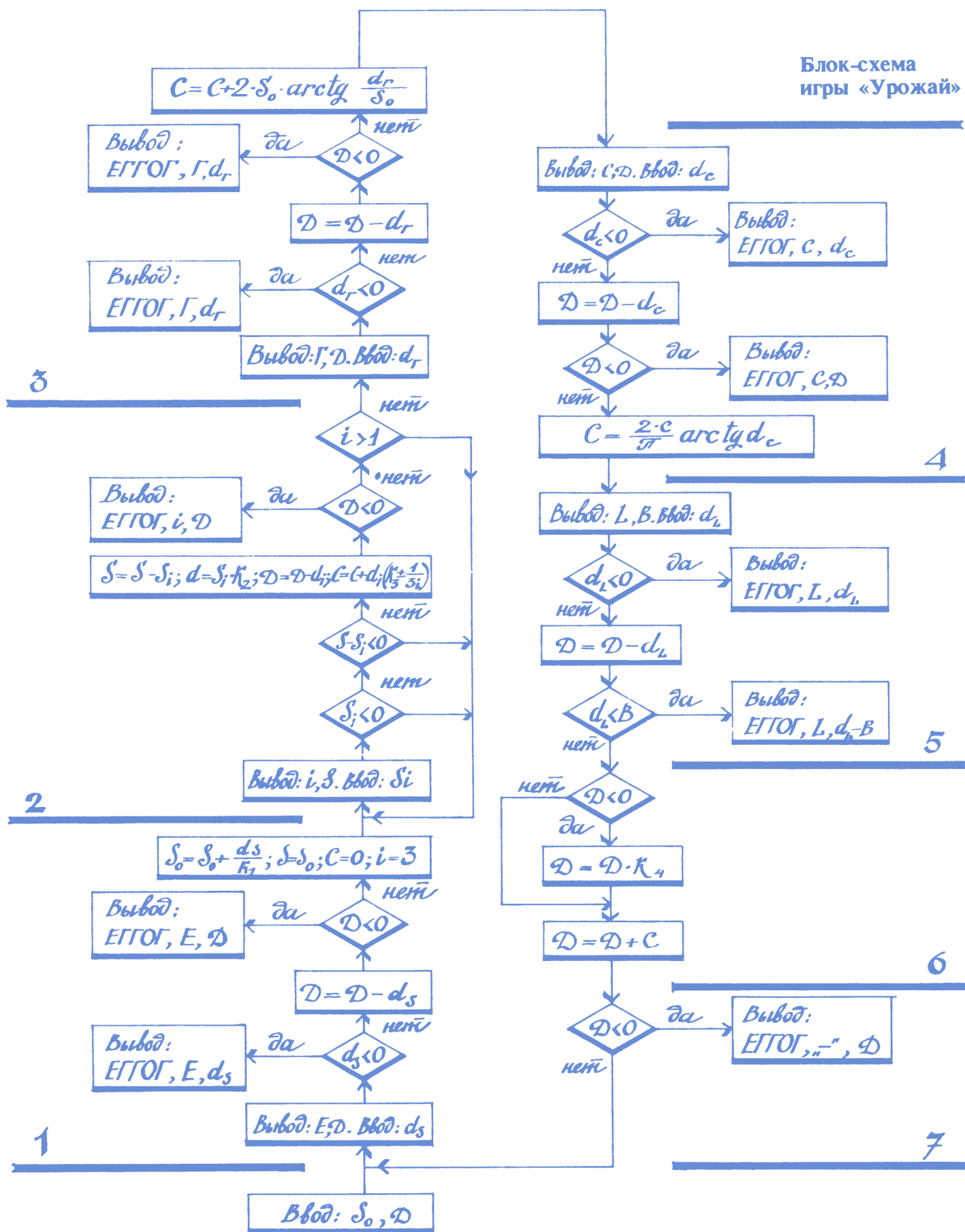
главной ветви, затем управление командой безусловного перехода (30—31) передается на адрес 46, после адреса 97 управление передается на короткую побочную ветвь, но это не приводит ни к каким неприятностям: в крайнем случае (при победе охотника) программа останавливается на адресе А7, на котором продублирована команда С/П с адреса 09. Предварительно команда по адресу 08 (А6) превращает ноль в сообщение 00. Фрагмент 49—51 «срезает» первую цифру числа — нестандартно используется команда ВП. Наконец, на адресах 10—19 записана довольно удобная подпрограмма случайного задания ± 1 ; она-то и управляет движениями скалоеда. Вот и все сложности.

Счастливой охоты!

*Игру придумал и описал
М. Пухов*

Что посеешь, то и пожнешь

Довольно обширный класс компьютерных игр (а их создание поставлено на Западе на широкую ногу) составляют экономические. Действие разворачивается обычно либо в современном капиталистическом обществе, либо в каком-нибудь уединенном королевстве много веков назад (или в иной экзотической обстановке). В первом варианте пользователь имеет шанс, например, за короткое время обогатиться (хотя бы в таком «электронном» варианте); во втором ему предоставляется возможность почувствовать себя всемогущим правителем, в чьих руках находятся судьбы целого государства. Машина

Блок-схема
игры «Урожай»

дает ему разнообразную информацию о численности населения, видах на урожай, стихийных бедствиях и так далее и выполняет его рас-

поряжения: сколько средств и на что нужно употребить. По «истечении года» подводится итог и начинается новый цикл. Подобных прог-

рам в книжках, посвященных играм на ПМК, довольно долго не было — вероятно, никто просто не предполагал, что таковая мо-

жет разместиться в мизерной памяти микрокалькулятора. Но постепенно, когда стало ясно, что возможности «игрушечных компьютеров» не так уж малы, начали появляться и такие игры, построенные, как правило, на тех же принципах, что и их «взрослые» западные аналоги.

«Представьте, что вас избрали президентом небольшого островного государства, — так, например, формулирует В. Потапов правила своей игры «Остров». — Задаются количество денег в казне, прожиточный минимум, стоимость засева единицы площади, стоимость очистки земли. Доход постоянен и зависит от образования населения и степени загрязнения земли отходами производства. Промышленность платит за аренду территории определенную сумму. Побочные доходы — туризм. Они растут с ростом чистоты окружающей среды и благодаря рекламе. Требуется ответить на следующие вопросы: 1. Сколько земли вы отдадите в аренду промышленности? 2. Сколько денег выделите населению? 3. Сколько засеете пашни? 4. Сколько отдадите на развитие туризма? 5. На очистку окружающей среды? 6. На образование? После ответа на последний вопрос получаем цифру — количество лет до очередных выборов — или сигнал о том, что вас досрочно сняли с правления, а то и посадили за решетку. Продолжение игры состоит в новом ответе на те же вопросы с учетом нового количества денег в казне, численности населения и загрязнения земли... Очень буду рад, если игру усовершенствуют».

Программа действительно нуждалась в серьезной доработке. Отсутствовал

элемент случайности, математическая модель была слишком упрощенной, при ответах на вопросы легко было сбиться — когда в игре много остановов, желательно как-то отличать их один от другого... Но самое главное — такие программы, как уже отмечалось, построены по образцу западных компьютерных игр. А они преследуют между тем не только развлекательные цели. Напротив, эти игры (разумеется, на упрощенных моделях) прививают пользователю те элементарные экономические знания, которые столь необходимы для жизни при государственно-монополистическом капитализме. Вводят его в мир биржевых и банковских операций, ненавязчиво пропагандируют соответствующий образ жизни, короче говоря, воспитывают будущих бизнесменов. А почему бы в порядке эксперимента не попробовать сделать игру, приближенную к нашей действительности? Игру, моделирующую (конечно, с учетом возможностей ПМК) азы плановой экономики?

Программа В. Потапова была коренным образом переработана (большой вклад в эту работу внес В. Алексеев). Были исключены некоторые второстепенные блоки, введено условие, что на имеющейся пахотной земле выращиваются три различные культуры со случайной урожайностью (она непредсказуемо меняется год от года), доход от сбора даров природы с быстрым насыщением (по мере исчерпания ресурсов он перестает увеличиваться), постоянные выплаты (отчисления в бюджет), средства на охрану собранного урожая... Кроме того, были разработаны варианты, соответствующие экстенсивному и интенсивному зе-

млепользованию, а также оригинальная система буквенной сигнализации, в том числе аварийной. Так появилась экономическая игра «Урожай», которая приводится в модификации для «МК-61» и «МК-52» (обладатели «БЗ-34» и «МК-54» должны помнить, что букве Е в командах по адресам 05, 45 и 56 соответствует на их клавиатуре стрелка вверх, а В/О в конце программы могут не ставить — адреса 98 и 00 у них все равно задублированы).

00.В/О 01.Сх 02.ПС
 03.ИП7 04.П1 05.КППЕ
 06.ИП4 07.÷ 08.ИПА 09.+
 10.ПА 11.П2 12.3 13.П1
 14.ИП2 15.ИП1 16.С/П
 17.Кх \geq 07 18.ИП2 19.ХУ
 20.— 21.Кх \geq 07 22.П2.
 23.ФВх 24.ИП3 25.× 26.†
 27.КПП8 28.Кх \geq 09 29.Fsin
 30.ИП1 31.÷ 32.3 33.÷
 34.ИП6 35.+ 36.× 37.ИПС
 38.+ 39.ПС 40.FL1
 41.14 42.ИП7 43.П1 44.
 КИП1 45.КППЕ 46.ИПА
 47.÷ 48.Farctg 49.2 50.×
 51.ИПА 52.× 53.КИП1
 54.+ 55.ПС 56.КППЕ
 57.Farctg 58.Фп 59.÷ 60.2
 61.× 62.ИПС 63.× 64.ПС
 65.КИП1 66.ИП1 67.С/П
 68.КПП8 69.ФВх 70.ИПВ
 71.— 72.Кх \geq 09 73.КИП1
 74.ИПД 75.Фх<0 76.79
 77.ИР5 78.× 79.ИПС 80.+
 81.ПД 82.Кх<оВ 83.ИП1
 84.ХУ 85.К— 86.ИПД 87.
 ИП1 88.С/П 89.Кх \geq о9
 90.КПП8 91.Кх \geq о9 92.ФВх
 93.В/О 94.ИПД 95.ХУ 96.—
 97.ПД 98.В/О.

Мы уже отмечали, что «космические» имитационные программы (вроде «Лунолета») помогают игроку как бы на собственном опыте «почувствовать» законы физики. Можно надеяться, что такие игры, как «Урожай» (взяв ее за основу и меняя различные блоки, можно моделировать

самые разнообразные ситуации; один такой пример приведен ниже), помогут в постижении основ экономики. Представьте себя директором небольшого совхоза или, например, руководителем коллектива рабочих на бригадном подряде. В вашем распоряжении имеются, скажем (все цифры, конечно, условные — на то она и игра), 1000 га пахотной земли (1000 ПА), прилегающие лесные угодья и довольно круглая сумма в 35 тысяч (35ВП 3 ПД). Задача — рационально ведя хозяйство, обеспечивать ежегодные постоянные отчисления в госбюджет не менее 3500 (3500 ПВ; число в регистре В служит одновременно и адресом перехода на начало программы, поэтому оно обязательно должно оканчиваться двумя нулями). Имеющаяся в вашем распоряжении земля, если ее вспахать, засеять, снять урожай, приносит в среднем 10% прибыли (1,1 Пб); обработка одного гектара обходится, скажем, в 30 рублей (30 ПЗ). Землю можно в любой пропорции отводить под три сельскохозяйственные культуры: допустим, рожь, пшеницу и кукурузу. Средний доход от них одинаков, но из-за погодных и конъюнктурных условий подвержен непредсказуемым колебаниям: рожь дает $110 \pm 11\%$, пшеница $110 \pm 17\%$ и кукуруза $110 \pm 33\%$ от вложенных средств, поэтому каждая культура может приносить не только прибыль, но и убыток. Какую-то часть земли можно вообще не засеивать (как правило, это приходится делать по необходимости — имеющейся суммы уже не хватает на обработку всей земли); этот «остаток», естественно, никак не влияет на состояние ваших финансов. Если дела идут

хорошо, можно выделить какую-то сумму на «расширение производства» — освоение новых земель: каждый гектар обходится, допустим, в 300 рублей (300 ПА); «списывать» землю (уменьшать выделенную площадь) нельзя — ПМК заподозрит финансовые махинации и среагирует незамедлительно.

Второй источник дохода — сбор и реализация даров природы: грибов, ягод, и т. д. При сравнительно малых затратах этот род деятельности дает 100% прибыли, но она быстро падает по мере увеличения вложенных средств: если собраны все грибы в соседних лесах (а их площадь будем считать пропорциональной площади полей), то сколько новых сборщиков ни посылай, будут одни убытки. Так что здесь имеется некий оптимум затрат.

Собрать урожай мало — нужно обеспечить его сохранность, выделив какую-то сумму на склады, сторожей, собак и прочее. Если она будет слишком мала, от урожая останутся рожки да ножки; если же чересчур велика — с урожаем ничего не случится, зато и на отчисления ничего не останется.

Все перечисленные выплаты: на освоение и обработку земли, сбор грибов и охрану — должны производиться из имеющихся наличных средств. Лишь для погашения отчислений в госбюджет можно взять небольшую ссуду в банке; но в этом случае после реализации урожая возвращать придется на 20% больше (1,2 П5). Если же средств на это не хватит, пеняйте на себя...

Собственно, все исходные данные есть, остались вспомогательные числа. В регистр 7 вводится буква Е (1 К — ВП П7); она служит

также адресом перехода и «сырьем» для образования других буквенных сообщений, поэтому иные сигналы использовать нельзя. Владельцам первых моделей «БЗ-34», на которых получить Е обычным способом не удастся, рекомендуем обратиться к помощи «сверхчисел»: $1 \text{ ВП } 50 \text{ Fx}^2 \text{ Fx}^2 \text{ 3} \times \text{ Fx}^2 \text{ ИПС } /- / \text{ ВП } 7 \text{ П7}$. В регистры 8, 9 и Е вводятся адреса переходов: 94 П8 83 П9 86 ПЕ (владельцы «БЗ-34» вместо последней команды могут записать ПО; в «МК-61» регистр 0 остается свободным).

Играть в «Урожай» просто: машинка сама все время подсказывает, что нужно делать. Устанавливаем переключатель Р—Г в положение Р и нажимаем В/О С/П. На индикаторе буква Е — признак блока подведения итогов и освоения новых земель. В регистре У — имеющаяся в вашем распоряжении сумма (35 000); какую-то ее часть вы можете сейчас выделить на освоение. А что, если вы поддадитесь на уговоры и дадите кому-нибудь участок под дачу за наличные? Задаем любое отрицательное число: 3000 /- / С/П. ПМК незамедлительно выдает ЕГГОГ — финансовые органы не дремлют. Нажимаем ХУ — на индикаторе вновь появляется буква Е, признак блока, в котором допущена ошибка. Еще раз ХУ; на индикаторе — 3000, то самое ошибочное число, которое мы только что ввели. А если бы задали на освоение земли сумму, превышающую имеющуюся наличность (скажем, 36 000), машинка выдала бы ЕГГОГ, затем Е и, наконец, допущенный перерасход (—1000).

Точно так же срабатывает ПМК и при ошибке во всех остальных блоках ЕГГОГ — признак блока —

3. НАШ НОВЫЙ ДРУГ — КОМПЬЮТЕР

введенное отрицательное число либо перерасход. Если в конце года в кассе не хватает денег даже на погашение взятой в банке ссуды, после ЕГГОГ и ХУ на индикаторе появляется знак «—», смысл которого ясен без дополнительных комментариев (при повторном нажатии ХУ вы узнаете в этом случае сумму недоплаты). В отличие от реальной жизни у вас есть шанс начать игру сначала; достаточно скомандовать В/О ПП В/О, заново ввести начальные значения переменных (землю и деньги) в регистры А и Д, затем С/П.

Так и поступаем (правда, в нашем случае, как легко убедиться, содержимое регистров А и Д еще не изменилось, так что вводить новые исходные данные необязательно). На индикаторе вновь загорается Е, в регистре У по-прежнему 35 000. Попробуем подойти к делу серьезно. Произведем несложный расчет. Земли у нас 1000 га, на ее обработку требуется 30 тысяч. Три с половиной придется отчислить в бюджет, еще, скажем, тысячу (откроем небольшой секрет: эта сумма близка в нынешней ситуации к оптимуму) употребим на сбор грибов и ягод... Итого 34 500. Да, и еще надо отдать сколько-то на охрану урожая! Средств, оказывается, в обрез, о новой земле думать рано. Нажимаем Сх С/П.

На индикаторе цифра 3 — знак того, что предстоит засеять три поля. В регистре У земельная площадь, которую можно обрабатывать, — 1000 (если бы мы в предыдущем блоке отвели, скажем, 300 рублей на освоение новых земель, площадь увеличилась бы ровно на один гектар). Сейчас нужно определить площадь

под посев ржи (с.-х. культура с 11% риска). Команда с отрицательным числом или с превышением имеющейся земли блокируется — на индикаторе вновь загорается цифра 3 (ПМК не прощает лишь серьезных финансовых нарушений, ошибки в планировании сельхозработ он дает возможность исправить). Избираем для начала стратегию «равных полей»: треть земли под каждую культуру. Поскольку суммарная площадь (1000) находится сейчас в регистре У, а в Х — тройка, достаточно нажать ÷. На индикаторе треть всей земли (333,33333) С/П. Задачи определены, ПМК принимается за работу: определяет расходы на обработку поля (очевидно, 10 тысяч), вычитает их из вашей кассы, а заодно сразу рассчитывает ожидаемый доход (с учетом всех колебаний) и заносит его в регистр С (в этой ячейке на протяжении всего цикла накапливаются будущие поступления, а потом, при подведении итогов года, они складываются с оставшимися в кассе деньгами). На индикаторе загорается цифра 2 — осталось засеять два поля. На очереди пшеница (17% риска). В регистре У — еще не засеянная земля (666,6667. Не мудрствуя, снова нажимаем ÷ (333,33335) С/П. На индикаторе мелькают числа, потом загорается цифра 1 — осталась кукуруза (33% риска). Для единообразия повторяем команду: ÷ (333,33335) С/П. Кстати сказать, на каждом оставшемся регистре С проверять, как идут дела; стек восстанавливается командами ИП2 ИП1.

На индикаторе буква Г — признак блока «Грибы и ягоды», в регистре У — оставшиеся в кассе деньги

(5 тысяч). Как намечено, отдаем сборщикам одну из них: 1000 С/П. ПМК послушно анализирует эту операцию, затем выдает на индикатор букву С — признак блока «Сторожа и собаки». В регистре У — оставшаяся сумма (4 тысячи). Пора выделять деньги на охрану собранного урожая и добытых даров природы. 150 С/П (откроем еще один небольшой секрет — это число в сложившейся ситуации также близко к оптимуму). Новые финансовые подсчеты, затем появляется буква L — пора вносить в госбюджет свою скромную лепту. Кроме регистра В, необходимая сумма (3500) находится сейчас и в регистре У. Любую попытку уклониться от этой выплаты калькулятор решительно пресечет.

Выбора нет — нажимаем 3500 С/П. ПМК подводит итоги года: на экранчике загорается буква Е. ХУ (32188,165). Очевидно, у нас неурожай, потеряны почти три тысячи. О новой земле нечего и думать, но и надежду терять рано. Сх С/П. На индикаторе цифра 3. Проводим прошлогоднюю стратегию: ÷ С/П (2) ÷ С/П (1) ÷ С/П (Г) 1000 С/П (С) 150 С/П (L) 3500 (в кассе чуть больше тысячи, волей-неволей приходится обращаться в банк за ссудой) С/П (Е). Второй год кончился, каковы результаты? ХУ (34099,159). Отлично! Дела пошли на поправку. Повторяем все еще раз: Сх С/П (3) ÷ С/П (2) ÷ С/П (1) ÷ С/П (Г) 1000 С/П (С) 150 С/П (L) 3500 С/П (Е). Еще один год позади! ХУ (35187,236). Урожай, стало быть, вновь выдался на славу — в кассе даже больше, чем было вначале!

Для теста, пожалуй, достаточно. Игра очень интересная и не такая уж про-

тая. Вовсе не обязательно, кстати, ограничиваться лишь приведенными константами; однако рекомендуется так их подбирать, чтобы средняя прибыль от урожая и сбора грибов примерно равнялась постоянным отчислениям — при этом достигается максимальный игровой эффект.

Теперь раскроем «секреты» программы. Из блок-схемы видно, что некоторые группы операций повторяются многократно. Соответствующие последовательности команд оформлены в две подпрограммы — ПП8 (она располагается на адресах 94—98) и ППЕ (86—93), а также блок аварийного останова (83—85). Для обращения к этим фрагментам (с адресов 05, 27, 28, 45, 56, 68, 72, 89, 90, 91) использованы команды косвенной адресации. Это сделано просто ради экономии места — команда КППЕ функционально полностью эквивалентна занимающему две ячейки прямому обращению ПП 86 (в регистре Е хранится число 86), КПП8 — ПП 94. С аналогичной целью употреблена и команда косвенного условного перехода по адресу 82: если число в регистре В заканчивается двумя нулями, она эквивалентна $Fx < 000$. Команды блокировки по адресам 17 и 21 используют в качестве адреса условного перехода букву Е, хранящуюся в регистре 7. Каждая из них функционально эквивалентна команде обычного условного перехода на адрес 14 (этому «явному» адресу, как мы знаем, соответствует «тайный» ОЕ).

Команды косвенного вызова КИП1 по адресам 44, 53, 65, 73 преобразуют записанную в регистр 1 букву Е последовательно в Г, С, L и —. Функционально они эквивалентны ИПД, ИПС, ИПВ и ИПА, причем во

втором и третьем случаях используются в программе и в этом качестве.

Осталось сказать о попавшей в начало команде В/О — казалось бы, ничто не мешает сдвинуть ее на «законное» место (адрес 97), начав «Урожай» со следующей команды: 00.Сх. Но если так сделать, нумерация изменится и использовать букву Е в качестве адреса перехода уже не удастся.

Легко видеть, что в разобранном варианте реализована экстенсивная экономическая модель: увеличение средних доходов можно обеспечить лишь за счет расширения посевных площадей. В этом случае вы полностью зависите от «капризов погоды» — даже если дела идут хорошо (попробуйте, например, задать в начале работы побольше наличных средств и освоить на эти излишки значительную новую площадь), достаточно одного неурожая, и в следующем сезоне будет попросту не на что засеять с таким трудом освоенные новые земли (ведь при неурожаяе хозяйство из прибыльного становится убыточным). Чтобы выйти из «опасной зоны», есть только один надежный путь: повысить урожайность (число в регистре б) — если она поднимется над пределами погодных колебаний, никакая засуха не страшна (к аналогичному результату, естественно, приведет и уменьшение самих погодных колебаний; например, вписав по адресу 29 вместо синуса команду Сх, мы сводим их к нулю, после чего нетрудно добиться устойчивого процветания вверенного вам хозяйства). Если модифицировать программу так, чтобы средства, выделяемые в блоке Е, шли не на увеличение площади, а на рост

коэффициента урожайности, получим простейшую модель интенсивного землепользования. Попробуем это сделать.

Казалось бы, существенной переделки не требуется. Просто ввести в регистр 4 другой коэффициент (скажем, 3 ВП 5 П4 вместо 300 П4 — это соответствует одинаковому увеличению средних доходов при одинаковых затратах), а фрагмент 08.ИПА 0.9+ 10.ПА изменить на 08.ИПб 09.+ 10.Пб. Но тут же сталкиваемся с существенной неприятностью: команда по адресу 11 раньше переписывала на каждом годичном цикле всю имеющуюся площадь (содержимое регистра А) в регистр 2 для дальнейшего использования в блоке посева. Надо сделать это и теперь, но при планируемом изменении необходима дополнительная команда ИПА (раньше мы после исполнения команды 10.ПА и без того имели в регистре X землю, теперь же там урожайность).

Куда вставить «лишнюю» команду? Программа перегружена, ни одной свободной ячейки нет, а жертвовать каким-либо блоком не хочется. Обратим внимание на фрагмент 01.Сх 02.ПС. Здесь происходит «обнуление» регистра С, в котором на протяжении цикла накапливаются доходы от урожая. Вовсе не обязательно, чтобы содержимое регистра в точности равнялось нулю — достаточно найти в начале программы место, где результат предыдущей операции заведомо невелик, и вставить туда команду ПС, а фрагмент 01—02 выбросить (ошибка при этом получится буквально копеечная). Практическое зануление регистра X происходит, например, после деления выделенной в

блоке E суммы на содержимое регистра 4 (команда по старому адресу 07) — ведь изменение коэффициента урожайности будет происходить, как нетрудно прикинуть, в самом крайнем случае на сотые доли. (В исходном положении, напомним, он равен 1,1.) Таким образом, приходим к следующему изменению программы: 00.В/О 01.ИП7 02.П1 03.КПЕ 04.ИП4 05.÷06.ПС 07.ИП6 08.+ 09.П6 10.ИПА. Дальше все остается как в прежнем, «экстенсивном» варианте (владельцам «БЗ-34» напомним, что букве E в команде КПЕ соответствует на их клавиатуре стрелка вверх). Проверочный тест, естественно, тоже остается прежним, но при переходе к новому варианту рекомендуем не забывать о восстановлении исходной урожайности (1,1 П6).

Любителям повозиться со «сверхчислами» можем посоветовать и еще один, очень экзотический вариант зануления регистра С. Для его реализации нужно вначале сформировать и заслать в регистр 9, например, ОС-оборотня, «хвостом» которого является ноль. Делается это с помощью вспомогательной программы 00.Fx² 01.Fx² 02.Fx² 03.× 04.П9 05.С/П. Нажимаем В/О 83, затем стрелку вверх и 1 ВП 60 С/П. После остановка в регистре 9 оказывается нужное «сверхчисло», выполняющее двойную работу: при команде ИП9 регистр С зануляется, а команды косвенной адресации по регистру 9 передают управление на адрес 83 (рекомендуем исследовать самостоятельно, каким образом работают такие команды при использовании «сверхчисел» различных типов, находящихся в разных регистрах). Теперь начало

программы «Урожай» будет выглядеть так: 00.В/О 01.ИП9 02.ИП7 03.П1 04.КПЕ 05.ИП4 06.÷ 07.ИП6 08.+ 09.П6 10.ИПА, а далее как в исходном варианте.

Игру придумали В. Алексеев и М. Пухов. Описал М. Пухов

Кольцо с началом и концом

Начало. Тем, кто бывал на ВЦ (имеется в виду посещение вычислительного центра, а не контакт с внешней цивилизацией), наверняка знакома такая картина — человек, склонившийся над текстом программы, шепчет странные слова: «Так, и куда же нам дуть? Ага, кажется, досюда». Для непосвященного это «птичий язык»: при чем здесь дуть? Специалист же легко узнает профессиональный жаргон. Именно так профессиональные программисты зачастую говорят об операторах цикла. Дело в том, что в алгоритмических языках высокого уровня в конструкции этого оператора используется слово DO (от английского глагола «выполнять»). Но что же такое цикл в программировании?

Это повторяющийся участок программы, который выполняется несколько раз подряд. Представьте себе, что нам надо вычислить значение одного и того же выражения (функции), скажем, в 100 различных точках. (Такие задачи типичны, например, при составлении таблиц.) Вряд ли есть смысл подряд выписывать все 100 этапов расчета. Давайте поступим так: зададим начальное значение аргумента, формулу для вычисления заданного выражения в общем виде и правило нахождения

нового значения аргумента по предыдущему. Теперь, начав работать, программа вычислит первое значение выражения (при начальном значении аргумента) и найдет следующее значение независимой переменной. И все? Пока все, ведь чтобы определить значение выражения во второй точке, надо вернуться в то место программы, где стоят команды вычисления функции. Это можно сделать с помощью передачи управления. Тогда программа найдет второе значение выражения и вычислит следующее значение аргумента. Затем управление опять передается на вычисление функции и т. д. Но что же получилось? Программа «бегает» по кругу и никак не может из него выйти, ведь «у кольца начала нет и нет конца». Как говорят специалисты, программа «зациклилась». Собственно, другого и быть не могло — мы не указали, сколько раз надо выполнить наш цикл вычислений. Как же найти, где кольцо кончается?

Для этого используют так называемый счетчик. В памяти машины выделяется ячейка, в которой записывается число проделанных повторений цикла, а в программу вставляется команда, прибавляющая к содержимому счетчика единицу после каждого выполнения цикла (так называются многократно повторяемые команды). Как только число в счетчике становится равным заданному, программа выходит из цикла и выполняет следующие команды. Возможен и счетчик, работающий на вычитание, когда после каждого повтора его содержимое уменьшается на единицу.

Но это, можно сказать, «ручное» программирование цикла, так как мы сами выписываем команды переда-



чи управления, организуем счетчик. В языках высокого уровня, как уже говорилось, есть специальные операторы цикла, облегчающие труд программиста, которому надо только после оператора цикла указать число повторов и выписать само тело цикла. Вся структура программы с необходимыми передачами управления машина создаст себе сама. Есть и более сложные циклы, когда выход из них определяется выполнением того или иного арифметического или логического условия. Однако не следует думать, что операторы цикла — достояние лишь компьютеров. И у нашего помощника «БЗ-34» есть команды, задающие цикл: FL0, FL1, FL2, FL3. Любая из них ставится после тела цикла, а сразу за ней указывается адрес первой команды тела цикла. Предварительно в счетчик — соответствующий регистр (0, 1, 2 или 3) — записывается нужное число повторов. Заметим, что здесь счетчик работает на вычитание — когда его содержимое равно единице, программа выходит из цикла.

«Полевые работы» повторяются «из года в год», и если вам повезет, то вы сумеете пройти десятки оборотов. При нерациональном ведении хозяйства игра закончится гораздо раньше.

Вообще любая игровая программа «по большому счету» — циклическая. Вы вводите управляющее воздействие — ПМК выдает результат, и опять все сначала. Но и внутри данной программы есть цикл. Работает он следующим образом. После ввода исходных данных и проверок ПМК определяет суммарную посевную площадь, подсчитывает прибыль (естественно, сначала она равна нулю), задает номер поля,

с которого начинается ваша хозяйственная деятельность. В связи с особенностями конкретного калькулятора (счетчик работает на вычитание) первым «обрабатывается» третье поле. На индикаторе загораются номер поля и суммарная посевная площадь. Теперь ваш ход: вы вводите площадь, которую отдаете под культуру № 3. После проверки ПМК находит площадь, свободную для посевов двух других культур (поля № 2 и № 1), затраты на возделывание третьей культуры, оставшуюся сумму денег и прибыль, которую вы получите, собрав урожай с третьего поля. Обратите внимание: прибыль зависит не только от затрат (то есть посевной площади), но и от того, какую культуру вы посеяли. Причем это влияние может быть как положительным, так и отрицательным — третья культура может изменить коэффициент на 1/9, вторая — на 1/6 и первая — на 1/3. Затем следует «финансовая» проверка. Если результат положителен, то можно приступать к «возделыванию» следующей культуры. Калькулятор уменьшает номер поля на единицу, и мы вновь оказываемся на начале цикла, но теперь пора задавать площадь под вторую культуру. Прделав цикл три раза, программа выходит из него и приступает к дальнейшей работе.

Игру описал С. Волков

Стратегическая игра «Волки и козлик»

Хотя программная память нашего «игрушечного компьютера» очень ограничена, его можно обучить неплохо играть и в некоторые

«нормальные» игры, в которых он выступает в качестве полноправного партнера человека. Сюда относятся «Крестики-нолики», игра Баше, «Морской бой» и даже игры на шахматной доске. Представляем вашему вниманию одну из наиболее «интеллектуальных» программ этого рода. Она называется «Победитель»:

00.КБПС 01.Кх≠09 02.5
03.— 04.П7 05.ИП8 06.П6
07.ИПО 08.П5 09.С/П 10.П8
11.ПП 12.44 13.Кх≠09 14.
КИП8 15.ФВх 16.—
17.Кх≥09 18.ФВх 19.КП8
20.ИПС 21.ИП6 22.Фх≥0
23.57 24.КППА 25.Кх=ОД
26.ИП6 27.КППВ 28.
Кх=ОД 29.ИП7 30.Фх² 31.
4 32.— 33.Фх≠0 34.39 35.
ИП6 36. ИПС 37.КППА
38.Кх≠ОД 39.ИП6 40.—
41.П8 42.ИП5 43.+ 44.Ф10^x
45.5 46.ПО 47.КИПЕ 48.ФВх
49.— 50.Кх≠ОД 51.FLO 52.
47 53.ФВх 54.ПО 55.ВП
56.В/О 57.+ 58.П6 59.ИП7
60.Фх=0 61.64 62.КППА
63.Кх=ОД 64.Фх² 65.ПО
66.FLO 67.72 68.ИПА 69.
ИП6 70.КППА 71.Фх=0
72.ИПС 73.КППВ 74.
Кх=ОС 75.ИП6 76.2 77.×
78.КППВ 79.Фх=0 80.26
81.ИПС 82.ИП6 83.— 84.П6
85.2 86.× 87.КППВ 88.Фх=0
89.26 90.ИП7 91.Фх² 92.ПО
93.FLO 94.31 95.FLO^x 96.
ИП7 97.КБПА

Букве Е в команде по адресу 47 на клавиатуре «БЗ-34» соответствует стрелка вверх. Перед игрой в некоторые регистры вводятся адреса переходов: 20 ПС 40 ПА 9 П9 41 ПВ 56 ПД. Содержимое регистров А и С служит вдобавок счетными коэффициентами. Теперь надо задать начальное положение шашек. Игра проходит на доске 9×9 (см. рисунок), правила ее просты: ход делается на одну клеточку по диагонали на свободное поле,

причем «волки» (ими руководит человек) могут ходить только вперед. Задача «козлика», ведомого ПМК, — прорваться на 9-ю горизонталь, цель «волков» — воспрепятствовать этому. Шашка ПМК в начале игры располагается на поле 55, «волки» — на полях 91, 93, 97 и 99. Каждый «волк» имеет собственный номер (от 1 до 4), его координаты хранятся в соответствующем регистре. В начале игры задаем 91 П1 93 П2 97 П3 99 П4. «Козлик» имеет номер 5 и располагается на поле 55: 55 П5. В регистре 6 по ходу игры задается направление предыдущего хода (разность между новыми и старыми координатами), то есть одно из чисел 11, —11, 9, —9. В начале игры можно задать любое из них, допустим, 11 П6. Все исходные данные введены.

Игра начинается командой В/О С/П (она же отдается и в том случае, если вы хотите пропустить ход — правилами это не возбраняется). В ответ ПМК выдает новые координаты «козлика» (в стандартной начальной ситуации число 64 или 66; если же «козлик» не знает, куда пойти, ПМК выбрасывает «белый флаг» — на индикаторе появляется ноль). Человек, в свою очередь, может пойти любой из четырех своих шашек. Для этого нужно набрать номер «волка», нажать ПП, набрать номер поля, куда вы решили пойти, и нажать С/П. Программой предусмотрены блокировки неверного хода: если «волк» идет на занятое поле или за границу доски, на индикаторе тут же появляется ноль; если «волк» попытался «сходить» назад, что запрещено правилами, на индикаторе загорается отрицательное число. В обоих случаях ход следует повторить: (номер «волка»)

ПП (номер поля) С/П. Если «козлик» выполнил свою задачу (достиг 9-й горизонтали), а человек пытается продолжить игру, на индикаторе появляется сообщение ЕГГОГ (ошибка). Этот останов происходит в подпрограмме; чтобы не засорять стек возврата, надо скоординировать 1 С/П, на индикаторе через несколько секунд появятся новые координаты «козлика», уже за пределами доски. Теперь можно заново расставлять шашки и начинать новую игру. Давайте посмотрим, как работают основные блоки программы.

На первом ходу команда 00.КБПС (так как в регистре С находится число 20, она эквивалентна БП 20) передает управление на адрес 20. Команды 20—23 анализируют предыдущий ход «козлика». Если он шел вперед, число в регистре 6 положительно (9 или 11), и ПМК проводит в жизнь «стратегию наступления» (блок 24—56). Сразу же следует обращение к подпрограмме проверки возможности хода, 40 (КППА) и 41 (КППВ). Сейчас исполняется команда КППА, следовательно, отправляемся на адрес 40; содержимое регистра 6 вычитается из двадцати, результат (11 или 9) записывается в рабочий регистр 8 и складывается с координатой «козлика» (40—43). ПМК, как видим, намерен идти зигзагом.

Вычисление показательной функции от полученного числа (адрес 44) несет двойную нагрузку. Во-первых, само оно записывается в регистр предыдущего результата X₁ для последующего использования. Во-вторых, если «козлик» уже достиг девятой горизонтали, результатом этой операции будет сообщение ЕГГОГ — «Вы что, не видите, я же уже выиграл!». В нормальной же

ситуации в рабочий регистр О записывается число 5 и цикл 47—52 проверяет, не занято ли намеченное поле одной из шашек (собственная клеточка ПМК проверяется потому, что этот же фрагмент используется и для блокировки неправильного хода человека). По возвращении из подпрограммы, если намеченное поле не занято и не лежит за нижней либо одной из боковых границ доски, в регистре X оказывается номер занятой «козликом» вертикали, команда условного косвенного перехода (25) передает управление на адрес 56, команда В/О перебрасывает его на адрес 01, после чего вычисляется и записывается в регистр 7 отклонение «козлика» от центральной вертикали, в регистры 5 и 6 переписываются соответственно новая координата «козлика» и направление сделанного хода. Программа останавливается в ожидании ответа соперника.

А что произойдет, если намеченный ход невозможен (поле занято либо лежит за пределами доски)? После возвращения из подпрограммы в регистре X оказывается ноль, и ПМК делает попытку хода вперед в прежнем направлении (26—27). Если и туда пойти не удастся, проверяются ходы назад (28—56), причем с 3-й и 7-й вертикалей попытка хода назад к ближайшему борту не производится. Если ходов нет, программа после всех проверок возвращается на адрес 01 с нулем в регистре X, и команда условного перехода передает управление на 09.С/П — ПМК признал себя побежденным.

Если в «атаке» ПМК действует довольно прямолинейно, то в «обороне» (предыдущий ход был сделан назад) ведет себя гораздо более изобретательно и остро-

рожно. Соответствующая ветвь начинается с адреса 57, проанализируйте ее самостоятельно. Любителям нестандартных приемов рекомендуем обратить внимание на кодово-адресную связку 71—72. Если входное число не равно нулю, управление передается на адрес, совпадающий с кодом команды 72.ИПС, то есть 6С. А этому «тайному» адресу соответствует «явный» 72, поэтому исполняется команда ИПС. Когда же входное число равно нулю, сразу исполняется 73. КППВ, по возвращении из ПП, как нетрудно проверить, в регистре X остается 0, поэтому фактически (хотя и с некоторой задержкой) в этом случае происходит переход на адрес 75. Операторы цикла, расположенные по адресам 66—67 и 93—94, также используются нестандартно — в качестве команд условных переходов: следующая по порядку команда исполняется только в том случае, если содержимое регистра 0 равно единице.

Блок ввода хода человека располагается на адресах 10—19. Номер шашки, которой вы решили пойти, записывается в рабочий регистр 8, а после ввода с пульта начисленного номера поля вызывается «укороченная» (начинающаяся с адреса 44) подпрограмма проверки возможности хода. Если поле занято, по возвращении из ПП имеем в регистре X ноль, и вычисления тут же прерываются — срабатывает блокировка (адрес 13). Затем фрагмент 14—17 проверяет, не пошел ли «волк» назад. Если да, срабатывает вторая блокировка. А при ходе вперед на свободное поле новые координаты «волка» записываются в соответствующий регистр, и ПМК погружается в анализ возникшей на доске ситуации.

Для перевода программы «Победитель» на язык новых калькуляторов «МК-61» и «МК-52» надо заменить команду по адресу 47 фрагментом 47.ИПО 48.ПЕ 49.КИПЕ (то есть ввести искусственную связь регистров О—Е), сдвинуть в фрагментах 22—23 и 60—61 адреса переходов на два (теперь они будут выглядеть так: 22. $F \geq O$ 23.59 и 62. $Fx = 0$ 63.66), адрес перехода на участке 66—67 сдвинуть на три (получится 68.FLO 69.75), две старые команды 71—72 заменить тремя новыми: 73. $Fx = O$ 74.78 75.ИПС, и записать другое число в регистр Д: 58 ПД. Таким образом, программа стала немного длиннее, но это не страшно — резервы в новых ПМК есть.

*Игру придумал и описал
М. Пухов*

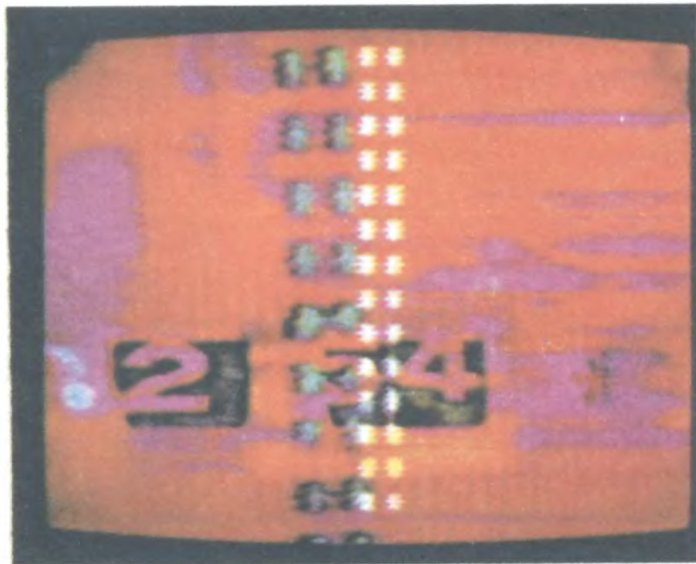
Динамические игры

Индикатор ПМК «Б3-34» и последующих моделей при вычислениях не отключается: на нем все время мелькают промежуточные результаты расчетов. Эта особенность дает возможность не останавливать программу для считывания интересующих игрока параметров (если таковых не слишком много). С другой стороны, ввод управляющего воздействия тоже можно осуществлять «на ходу», через переключатель Р—Г. Если, например, в программе имеется фрагмент, вычисляющий косинус ста, то при установке этого тумблера в позицию Г на выходе блока будем иметь отрицательное число (косинус ста градусов равен $-0,17364817$), при переводе его в положение Р — положительное (косинус ста радианов равен $0,86231977$), в

промежуточной же позиции ГРД (на корпусе Б3-34 она, кстати, не отмечена; углы задаются в сотых долях прямого угла — «градусах») — ноль. Таким образом, манипулируя этим переключателем, получаем возможность направлять ход вычислений по трем независимым ветвям. Этого достаточно для многих игр. Вот, например, программа «Лунолет-Д»:

00.ИПА 01.†02.†03.†04.†05.†
ИПВ 06.†07.†08.†09.†10.ИПД
11. $Fx \neq 0$ 12.35 13.ИПС 14.
 F_{\cos} 15. $Fx \neq 0$ 16.21 17.†18. Fx^2
19. $F\sqrt{20.} \div 21.1$ 22.+ 23. Fx^2
24.ИПО 25.× 26.— 27.FVx
28.XY 29. $Fx < 0$ 30.33 31.+
32.0 33.ПД 34.Fo 35.ИПО
36.— 37.+ 38.ПВ 39.+ 40.2
41.÷ 42.ИПА 43.+ 44.ПА
45. $Kx < OC$ 46.2 47.× 48.XY
49.ИПВ 50.— 51.× 52.ИПВ
53. Fx^2 54.+ 55. $F\sqrt{56.}$ ПИ
57.0 58.ПА 59.+ 60./—/
61.ПВ 62.I 63.3 64.+ 65.
 $Fx < 0$ 66.69 67.FVx 68.ПИ
69.ИП1 70.5 71.× 72.F10^x
73. Fx^2 74. Fx^2 75. Fx^2 76.†77.
ИПД 78.ИПВ 79.С/П

На адресах 01—04, 06—09, 17 и 76 расположилась стрелка вверх, на адресах 19 и 55 — извлечение квадратного корня, на адресе 34 — круговое перемещение по стеку (код 25). Играть в эту игру чрезвычайно просто. В регистр 0 вводится ускорение силы тяжести на планете в m/c^2 (лучше задавать его целым числом), в регистр Д — начальный запас топлива в кг, в регистр А — начальная высота в м, в регистр В — начальная вертикальная скорость в м/с (знак «минус» соответствует снижению). В регистр С закладываем число 100. «Рычагом управления» служит переключатель Р—Г. Позиция Г соответствует выключению двигателя, промежуточное положение ГРД — малой тяге, Р — пол-



ной. Малая тяга в точности уравнивает силу гравитации, а полная в четыре раза больше. Расход топлива численно равен реактивному ускорению.

Игра начинается командой В/О С/П. На индикаторе мерцает текущая высота полета (это обеспечивает команду 00—04), затем ее сменяет скорость (05—09), на мгновение проблескивает наличный запас топлива (10), и начинаются вычисления. Прежде всего производится проверка, не опустели ли баки (11—12). Если да, то ракетный блок пропускается и управление передается сразу на адрес 35, в блок расчета новых скорости и высоты полета. Если же топливо еще есть, фрагмент 13—20 анализирует положение переключателя Р—Г: если тот установлен в позицию Р, на выходе блока имеем 1, если в позицию Г, то —1, промежуточная же позиция дает 0 (кстати, владельцы «МК-61» и «МК-52» могут заменить шесть команд 15—20 одной-единственной КЗН). Затем на основании полученной информации ПМК определяет реактивное ускорение и расход топлива, новые скорости и высоту полета. Делается это примерно, как и в программе «Лунолет-1», причем шаг во

времени взят равным одной секунде. При анализе программы надо помнить, что в верхних регистрах стека находится скорость и топливо, введенные туда командой 05—10.

Команда по адресу 45 сравнивает новую высоту полета с нулем. Если высота положительна (еще летим), то управление передается на начало программы, и все повторяется. Полный цикл вычислений занимает секунд 15, а при пустых баках — еще меньше. Если же высота отрицательна, задействуется блок прилунения (46—79). Сначала определяется посадочная скорость (46—61), затем она анализируется с помощью «сверхчисел». При скорости, не превышающей 2,5 м/с (посадка на «отлично»), она просто выводится на индикатор и вычисления останавливаются. Если скорость посадки лежит в диапазоне 2,5—5 м/с («хорошо»), на индикаторе появляется ЕГГОГ. Нажимаем С/П, на индикаторе — скорость. В обоих случаях в регистр Y выводится оставшийся запас топлива. При дальнейшем увеличении посадочной скорости ПМК выдает ЗГГОГа; его надо сбросить (Сх), а скорость и топливо смотреть в соответствующих регистрах (ИПВ и

ИПД). Затем наступает очередь серьезных аварийных ситуаций. При прилунении со скоростью 7,5—10 м/с на индикаторе появляется испорченный фрагмент программы; «ремонт» корабля в этом случае можно произвести по следующей методике: ФПРГ ШГ влево Сх С/П ФАВТ ФПРГ ШГ влево ШГ влево, после чего вписать две команды вместо Сх С/П, перейти в режим АВТ и делать следующую попытку. Очень «коварно» ведет себя ПМК, если скорость лежит в диапазоне 10—12,5 м/с. На индикаторе как ни в чем не бывало зажигается ее величина, в регистре Y находится остаток топлива. Однако попытка продолжить полеты не удастся: в регистр С заслан «хвост» ОС-оборотня, поэтому переключатель Р—Г не работает — у лунолета вышел из строя двигатель. Чтобы его исправить, придется снова заслать 100 в регистр С. А при скорости свыше 12,5 м/с вас ждет худшее. Тьма...

В качестве примера можно предложить такой комплект исходных данных 2 ПО 50 ПД 500 ПА 0 ПВ 100 ПС. Задача — сесть с оценкой «хорошо» (по крайней мере, заработать отлично в этом варианте никому еще не удавалось).

Автомобиль

Попробуем теперь отойти от своей обычной космической тематики и попробуем, например, разработать программу «Автомобиль». Раз уж мы им управляем, логично считать позицию Г положением «руль вправо», Р — влево, а промежуточную — прямо. Обучить ПМК распознавать эти сигналы мы уже умеем, остается подумать, как выводить информацию. Цифры изрядно поднадоели, хотелось бы придумать что-нибудь понагляднее.

Очень хорошее видео-сообщение для игр типа «Авторалли» выглядит так: 11181111, причем восьмерка может перемещаться относительно единичек. Картинка — это поперечный разрез дороги; единички изображают полосы движения, восьмерка — легковой автомобиль (мы смотрим на шоссе сверху). Только как удержать это изображение на индикаторе? Если воспользоваться стрелкой вверх, как в предыдущей игре, то справа вплотную к дороге будет гореть код команды (ОЕ), так что картинка окажется подпорченной.

Какими еще командами можно воспользоваться? КНОП, К1, К2 — в любом случае будет мешать код. Идеальная «демонстрационная» команда должна, с одной стороны, быть нейтральной, с другой — не иметь никакого кода! Или, что то же самое, иметь код «пусто-пусто».

К счастью, есть простой способ «опустошения» программной памяти, не описанный в инструкции к ПМК. Командуем В/О ШГ влево ШГ влево ФПРГ КПП9 FАВТ. Подготовительная работа закончена, теперь, чтобы записать код

«пусто-пусто» на адреса А, А+2, А+4, А+6, А+8, достаточно записать в регистр 9 число А+11, повторить пять раз последовательность В/О ШГ влево ШГ влево ПП, а затем перейти в режим ПРГ. Так и поступаем: вначале записываем в регистр 9 число 11 и проделываем всю процедуру («пусто-пусто» вписалось на адреса 00, 02, 04, 06, 08), затем повторяем то же самое с числом 12 («опустошаются» адреса 01, 03, 05, 07, 09), потом с числом 21 (10, 12, 14, 16, 18) и, наконец, 22 (11, 13, 15, 17, 19). В результате коды «пусто-пусто» образовали как бы искусственную «темную зону» на адресах 00—19. «Экран» для демонстрации подготовлен.

В качестве базового видеосообщения используем 11111111 («пустая дорога»). Чтобы получить картинку с автомобилем, надо к этому числу прибавить 7, умноженное на десять в соответствующей степени. Это можно сделать с помощью следующего фрагмента:

20.7 21.ИПВ 22.ИПС
23.Fcos 24. 25.ВП 26.Fx²
27.FV 28.÷ 29.+ 30.ПВ
31.F10^x 32.× 33.ИПА 34.+
35.КБПС

По адресу 24 вписана стрелка вверх, по адресу 27 — извлечение квадратного корня. Владельцы «МК-61» и «МК-52» могут заменить блок 24—28 командой КЗН. Остается вернуться в режим АВТ, ввести в регистр А «пустую дорогу» (11111111 ПА), в регистр С — число 100, в регистр В — начальное положение автомобиля (номер дорожки, считая справа). К примеру, 3 ПВ. Теперь В/О С/П — и катаетесь на здоровье.

*Игру придумал и описал
М. Пухов*

Шахматные часы

К динамическим относится и прикладная программа «Шахматные часы»:

00.ИПО 01.× 02.П1
03.П2 04.2 05.F10^x 06.Fcos
07.Fx≠0 08.19 09.Fx<0
10.15 11.FL1 12.04 13.1
14.С/П 15.FL2 16.04 17.2
18.С/П 19.ИП2 20.ИПО
21.÷ 22.ИП1 23.ИПО 24.÷
25.С/П 26.БП 27.00

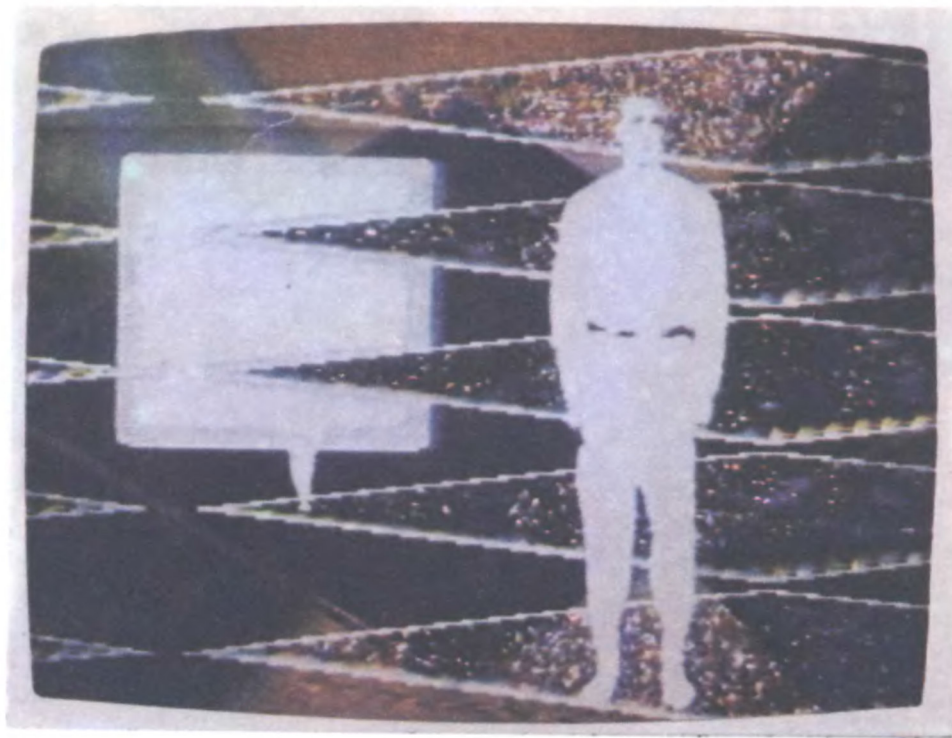
Ее назначение полностью соответствует названию. В регистр 0 надо заслать пересчетный коэффициент — количество циклов, выполняемое программой за одну минуту. Это число лежит в пределах от 17 до 18 и для каждого экземпляра ПМК подбирается опытным путем. Для редакционного оно равно 17,39. Наберите теперь количество минут, отведенное на игру каждому игроку (допустим, 5), установите переключатель Р = Г в нужное положение (Г — первый игрок, Р — второй) и отдайте команду В/О С/П. Сделав ход, игрок переводит переключатель Р = Г в противоположное положение — начинается отсчет времени партнера. При исчерпании лимита времени у одного из игроков ПМК останавливается и высвечивает номер игрока. Если же игра завершилась раньше, переводим переключатель Р = Г в среднее положение (ГРД). На индикаторе — время, оставшееся у первого игрока, в регистре Y — у второго.

Предупреждаем — во всех динамических играх переключатель Р = Г нужно переводить плавно, осторожно, без лишних рывков. Все-таки ПМК не рассчитан на такой режим эксплуатации.

Календарь для прошлого, календарь для грядущего

Когда читаешь мемуары или исторические рассказы, нередко возникает желание узнать, в какой день недели произошло то или иное событие — эта информация может прояснить описываемую ситуацию. Но как ее получить? Вопрос непростой, к тому же неоднозначный — во многих странах отсчет времени ведется по-разному, причем в разные периоды нередко применялись различные календарные системы. Например, в отечественной истории с февраля 1918 года действует григорианский календарь, ему предшествовал юлианский, но и тот применялся лишь начиная с реформы Петра Первого, а до этого был еще один вариант летоисчисления — от «сотворения мира» и Новый год наступал, увы, не 1 января. Да что там далекие века — даже для 30-х годов определять день недели абсурдно, в этот период семидневная неделя была временно заменена на шестидневную: вспомните фильм «Веселые ребята» или «Укрощение огня».

На сегодня существует множество различных «вечных календарей»: всевозможные таблицы, формулы и даже хитроумные «бумажно-механические» устройства. В книге Я. Трохименко и Ф. Любича «Микрокалькулятор, ваш ход!» была предложена методика расчета дней недели на программируемом микрокалькуляторе «Электроника БЗ-21», однако ограниченные возможности этого ПМК позволяют «уложить» в программную память только



саму формулу, что значительно усложняет работу с программой.

При решении чисто практических задач наряду с получением числовых данных иногда появляется необходимость сформировать служебную информацию, объясняющую полученные результаты, например, выделяющую одно число из ряда других, или имеющую самостоятельное значение, например, дни недели. Вариант применения видеосообщений для первого случая продемонстрирован в игровой программе «Урожай», а использование ПМК для получения нецифровой информации попробуем показать на примере программы «Вечный календарь». Для ПМК «Электроника БЗ-34» она выглядит так:

00. † 01.КППА 02.П5 03.—
04.2 05.F10^x 06.× 07.†
08.КППА 09.П4 10.—11.4
12.F 10^x 13.× 14.ПЗ 15.ИПВ
16.—17.F X ≥ 0 18.20
19.КИП5 20.ИП4 21.3 22.—
23.F X < 0 24.29 25.КИПЗ
26.ИП4 27.9 28.+ 29.ИПС
30.× 31.ИПД 32.+
33.КППА 34.5 35.ИПЗ 36.×

37.4 38.÷ 39.КППА 40.+
41.ИП5 42.+ 43.† 44.† 45.7
46.÷ 47.КППА 48.7 49.×
50.— 51.F X = 0 52.54 53.7
54.П8 55.7 56.ПО 57.9 58.†
59.Сх 60.ХУ 61.КНОП 62.ВП
63.КП † 64.F L0 65.63
66.ИП8 67.КП8 68.8 69.ПО
70.КИПО 71.ВП 72.7 73.П8
74.КИП8 75.КИП † 76.ИП8
77.ХУ 78.ХУ 79.ВП 80.1
81./—/ 82.F L0 83.73
84.С/П 85.П8 86.КИП8
87.F ⊗ 88.F ⊗ 89.ИП8
90.В/0

Для «Электроники МК-61»:

00. † 01.К[X] 02.П5 03.—04.2
05.F 10^x 06.× 07.К[X]
08.П4 09.F Вх 10.К{X} 11.4
12.F 10^x 13.× 14.ПЗ 15.ИПВ
16.—17.FX ≥ 0 18.20
19.КИП5 20.ИП4 21.3
22.—23.F X < 0 24.29
25.КИПЗ 26.ИП4 27.9 28.+
29.ИПС 30.× 31.ИПД
32.+ 33.К[X] 34.5 35.ИПЗ
36.× 37.4 38.÷ 39.К[X]
40.+ 41.ИП5 42.+ 43.†
44. † 45.7 46.÷ 47.К[X]
48.7 49.× 50.— 51.F X = 0
52.54 53.7 54.ПО 55.П6
56.КИП6 57.FL0 58.62
59.3 60.КПП8 61.КБПЕ
62.F L0 63.66 64.БП 65.85
66.FL0 67.70 68.КППА

69.КБПЕ 70.FLO 71.76
 72.КППА 73.ИП9 74.К+
 75.КБПЕ 76.F L0 77.81 78.3
 79.КПП8 80.КБПЕ 81.F L0
 82.85 83.КППА 84.КБПЕ
 85.2 86.ПП 87.96 88.К Инв
 89.К{X} 90. † 91.ВП 92.7
 93.С/П 94.2 95./—/ 96.ИП6
 97.F 10 ÷ 98.^х 99.9 —0.F
 1/X —1.5 —2.x —3.+
 —4.V/0

Структурно каждая из них состоит из трех частей: первая — расфасовка даты по регистрам (адреса 00—14), вторая — собственно счетная (15—54), третья — формирование календарного видеосообщения. Первые две части в обеих программах практически идентичны, только на «МК-61» выделение целой части производится одной операцией (адреса 01, 07, 33 и 39), а на «БЗ-34» для этой цели используется КППА (85—90). Кстати сказать, возможности «БЗ-34» не позволяют отбросить дробную часть у числа, если целая равна нулю, поэтому иногда для первого воскресенья марта месяца, точнее, для дробного високосного коэффициента без целой части, эта подпрограмма не срабатывает. Блок «растаскивания» числа по регистрам структурно очень прост и в комментариях не нуждается. Непосредственно счетная часть, как уже сообщалось, целиком взята из книги Я. Трохименко и Ф. Любича и поэтому подробно здесь тоже не рассматривается, следует только отметить, что расширенные возможности новых ПКМ позволили более чем в два раза «ужать» этот алгоритм. Завершающая же часть программ выполнена по-разному. На «БЗ-34» для формирования итогового видеосообщения используется метод, опубликованный в журнале «Техника — моло-

дежи» № 6 за 1986 год: в регистры 1—7 засылается знак «—» (адреса 57—65), порядковый номер дня недели вписывается в соответствующий регистр (66—67), и из полученных знаков образуется видеосообщение, в котором вычисленный день недели, обозначенный цифрой от 1 до 7, стоит на своем порядковом месте, а прочие дни обозначены прочерком. На «МК-61» используются логические операции с двоично - шестнадцатеричными числами: сначала операторы F L 0 передают управление на тот участок программы, который ответствен за формирование буквенно-цифровой заготовки нужного видеосообщения: к цифре 5, стоящей в нужном знакомстве в числе 0,55555555, прибавляется ± 2 или 3 (подпрограмма КППА по адресам 94—A4), формирование буквы L (заготовка для четверга) производится при помощи операции логического сложения (адреса 73—74), затем видеосообщение приводится к окончательному виду командой K Инв, при этом цифра 3 трансформируется в букву С (среда, суббота), 2 — в Г (понедельник, пятница), 7 — в В (вторник, воскресенье), L — в Ч (четверг), пятерки же — в прочерк. Таким образом, цифры на «МК-61» заменены буквами — так нагляднее.

Пора приступить к работе: вводим программу, затем константы. В регистр В — год перемены старого стиля на новый (1918 ПВ), в регистры С и Д — счетные и поправочные коэффициенты (2.6 ПС и 0.4 ПД). Эти цифры одинаковы для обоих типов ПКМ, а адреса переходов разные. Для «БЗ-34» ввод констант заканчивается командой

придется набрать последовательность 95.П8 10008 П9 94 ПА 88 ПЕ. Порядок работы прост: набираем число и через запятую шесть цифр — месяц и год, например, 9 мая 1945 года соответствует 9.051945, затем В/О С/П, через минуту на индикаторе «БЗ-34» появляется видеосообщение «—3—», на МК-61 — «—С—», то есть «среда». Вводим следующую дату: 9.011905 В/О С/П, результат: «—7—» или «—В—» — «Кровавое воскресенье» 1905 года. А XXI век начнется с понедельника: 1.012001 В/О С/П «1 —» или «Г —».

Для отечественной истории действие этого календаря с учетом старого и нового стилей распространяется от принятия Петром Первым юлианского летосчисления — 1700 год — до 2100 года. Этот год, хотя и делится на 4 без остатка, тем не менее согласно григорианскому летосчислению не является високосным, а счетный алгоритм программ этого не учитывает, впрочем, в некоторые периоды, когда накапливающаяся ошибка будет кратна семи, предложенный вечный календарь будет давать верные результаты, читателям предлагается самостоятельно определить дальнейшие интервалы применимости программ. А если кто-нибудь захочет досконально разобраться в их счетной части, а также составить программы других календарных расчетов или, допустим, биоритмов, то ничто не мешает обратиться к первоисточнику: Трохименко Я., Любич Ф., Микрокалькулятор, ваш ход! («Радио и связь», 1985).

*Игры придумал и описал
 В. Алексеев*