

Apple Computer

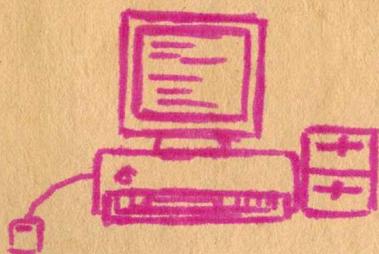
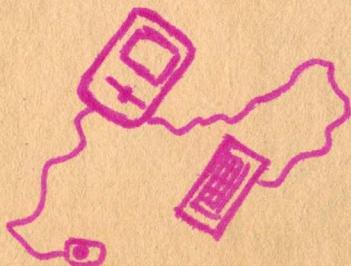


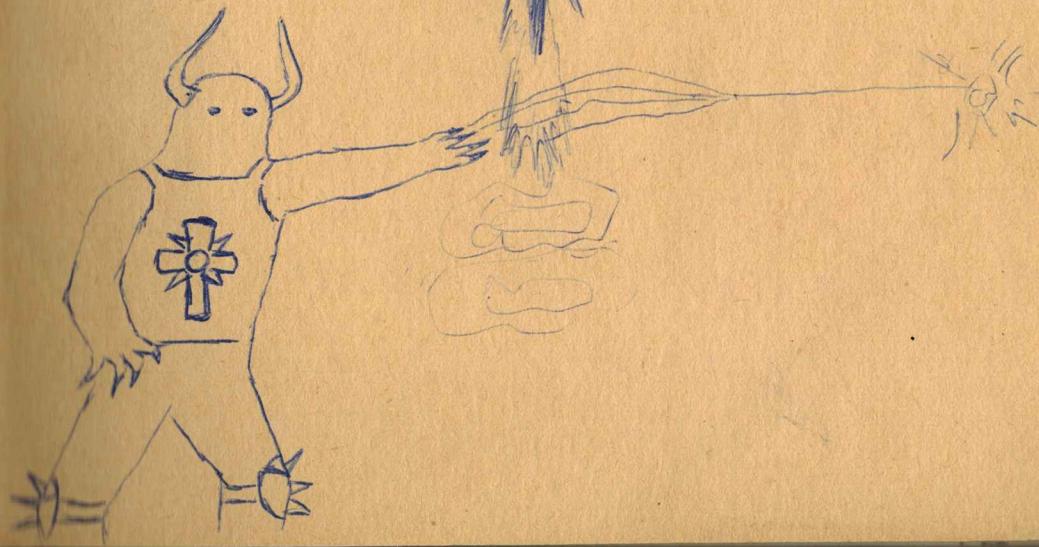
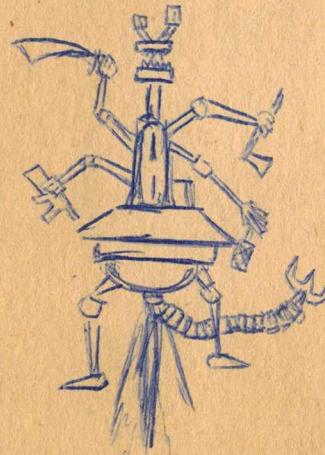
Sanyo



IBM

EKTROHNIKA





578,0 = $5 \cdot 10^2 + 7 \cdot 10 + 8 \cdot 10^0$ Система счисления.
0,1;2;...9 арифм в позицн присты
 10-сист

Системы счисления
 Позиционная
 Непозиционная

"10"-сист. - позиц.

555,₁₀ = $5 \cdot 10^2 + 5 \cdot 10 + 5 \cdot 10^0$ = 555 0101
 const eq

Римская - непозиц.

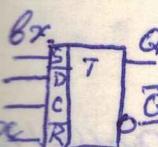
+	0+	1+	0+
0+	0+	1+	1+
+	0	0	1

I; II; III; IV; V; VI.

$$\overline{X} \overline{X} \overline{X} = 30$$

$$\overline{V} \overline{V} = 10$$

$$110 = \overline{0101}$$



вых ($1'' \approx 2 \dots 4,5V$)

вых ($0'' \approx 0,1 \dots 0,5V$)

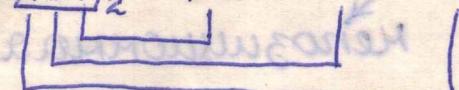
Двоичная-позиционная
система

1 и 0.

15 им - двоичный разряд

см. ил.

$$0111_2 = 1 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2 = 7_{10}$$



сумма

$$1110 = 0 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3 = 14_{10}$$

$$1010 = 0 \cdot 2^0 + 1 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot 2^3 = 10_{10}$$

$$\begin{array}{r} + 0 \\ \hline 1 \\ \hline 1 \end{array} \quad \begin{array}{r} + 1 \\ \hline 10 \\ \hline 0 \end{array} \quad \begin{array}{r} + 0 \\ \hline 0 \\ \hline 1 \end{array}$$

$$\begin{array}{r} + 10 \\ \hline 11 \\ \hline 1 \end{array}$$

$$101 = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 = 5_{10}$$

$$\begin{array}{r} + 011 \\ \hline 111 \\ \hline 1010 = 10_{10} \end{array}$$

$$11.11 = 1 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^{-2}$$

Преимущества двоичной
системы счисления.

Простота выполнения арифмети-
ческих и логических действий

Выполнения логических операций
на базах с помощью простых
схем.

Использование элементов
имеющих два устойчивых
состояния

Информации простота
изодированием.

1 бит - двоичный разряд

1 байт = 8 бит

микро ЭВМ

1 слово = 2 байта

большие ЭВМ

1 слово = 32 байта

L = 50 C = 100 M = 1000

Домашнее задание

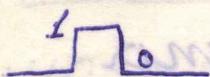
$$01101 = 13$$

$$11101 = 29$$

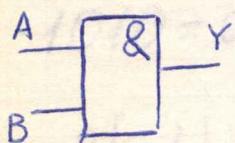
$$1100 = 12$$

$$1101 = 27$$

$$1111 = 15$$

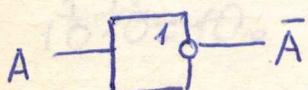


"U"

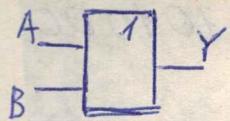


"HE"

A	B/Y
0	0
1	0
0	1
1	1



A	Ā
0	1
1	0



"ИЛИ"

$$\begin{array}{c} A \quad B \quad Y \\ \hline 0 \quad 1 \quad 1 \end{array}$$

$$0 \quad 0 \quad 0$$

$$1 \quad 0 \quad 1$$

$$1 \quad 1 \quad 1$$

0-0-0

3-0-1

4-1-0

2-1-1

6-1-0

7-1-1

8-1-0-0

9-1-0-1

10-1-0-1-0

11-1-0-1-1

12-1-1-1-1

13-1-1-1-1-0

14-1-1-1-0

15-1-1-1-1

16-1-0-0-0

17-1-0-0-1

18-1-0-0-1-0

19-1-0-0-1-1

20-1-0-0-1-1-1

21-1-0-0-1-1-1-1

Л-логическое умножение

В-логическое сложение

& &-амперсанд

$$(a \wedge b) \vee (c \vee d) = ①$$

$$\overline{(a \wedge b)} = 1$$

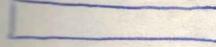
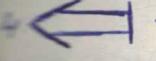
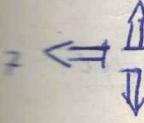
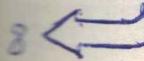
$$(c \vee d) = 1$$

$$(a \wedge b) \vee (c \vee d) = ①$$

$$\begin{array}{r} 15118 \\ \times 7118 \\ \hline 15118 \\ 7118 \\ \hline 10000 \end{array}$$

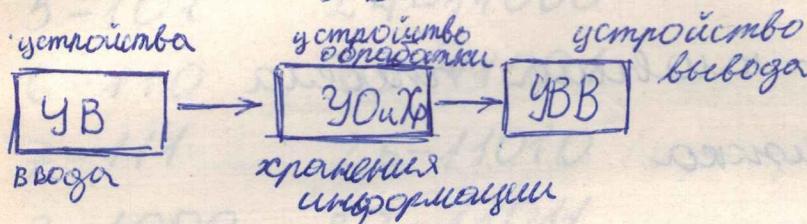
1-001	20-10100	40-101000
2-010	21-10101	
3-011	22-10110	
4-100	23-10111	
5-101	24-11000	
6-110	25-11001	V-10100000
7-111	26-11010	V-10100000 - V
8-1000	27-11011	
9-1001	28-11100	X-01100000 - X
10-1010	29-11101	
11-1011	30-11110	(A) = (AVS)(ACDA)
12-1100	31-11111	$\Delta = \partial V \Delta$
13-1101	32-100000	$\Gamma = (\partial V C)$
14-1110	33-100001	(AVS)(ACDA) = (AVS)(V(CDA))
15-1111	34-100010	
16-10000	35-100011	
17-10001	36-100100	81 131 81 14
18-10010	37-100100	F
19-10011	38-100110	
	39-100111	

Клавиши русского языкового и редактирования.

- 1 Заг - заглавные буквы
- 2 Стр - строчные буквы
- 3  - клавиша пробела
- 4  - ёвильска
- 5  - клавиша сдвига
- 6 ПОВТ - повтор
- 7  - клавиши управления курсором.
- 8  - перевод курсора на строку ниже.
- 9 ЛАТ - латинский шрифт
- 10 РУСК - русский шрифт
- 11  - определение нового значения клавиши.

12 Сброс - сброс.

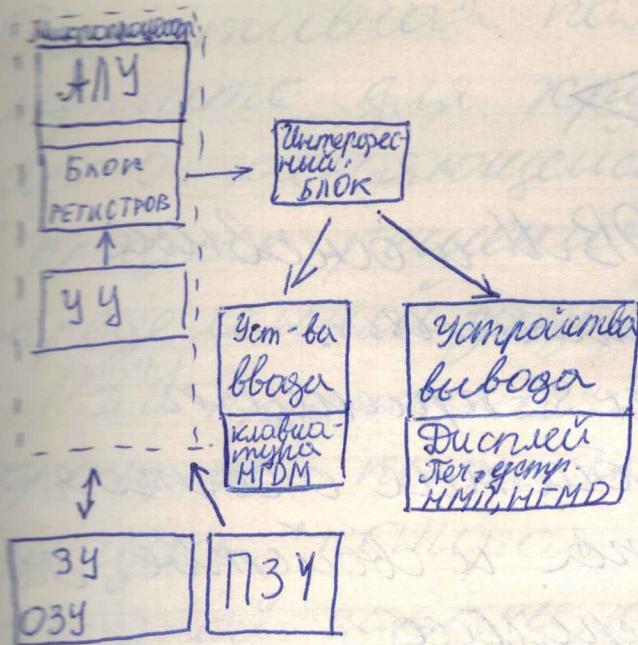
Структурная схема ЭВМ.



Устройство вывода информации.

Устройство вывода предназначено для устройств вывода информации.

Устройство обработки и хранения состоит из процессора и памяти



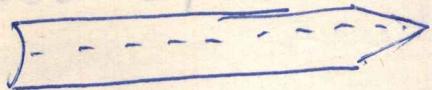
АЛУ - арифметико-логическое устройство

УЧ - устройство управления

ОЗУ - оперативное запоминающее устройство.

ПЗУ - постоянное запоминающее устройство.

Слово - слово



Нашему ЭВМ и основные
типы запоминающих.

Информация - прием
передачи, сообщение о состоя-
нии предмета и свойстве яв-
ления или процесса.

Нашему ЭВМ подразделяется
на внешнюю и внутреннюю.
Внутренние память высокого
быстродействия и ограниченной
емкости.

Она изготавливается из полу-
проводниковых материалов
а также из магнитных
материалов.

Оперативная память
служит для хранения
часто меняющейся информации
в процессе решения задач
определенной задачи.

ПЗУ предназначено для
хранения постоянной информа-
ции, которая независима
от той задачи, которую
решает на ЭВМ. Часто способ
решения той же задачи.

Большая память предназна-
чена для числового хранения
информации независимо от
того: числового или текстово-
го, но машинного. Во большую
память записываю информацию
из меняющейся в процессе
решения задач.

Внешнее пальто go 129 из ~~бак~~ от

Внешние устройства ЗВМ.

НМД и НМЛ.

ОЗУ - ~~где где~~

Внешняя панель предполагена
для заповещенного отражения
инфракрасного.

Магнитное подсвечи: магнитные
ленты, диски, карты, барабаны.

Магнитные ленты: катушечные,
касетные.

Катушечная лента
ширина
~~ширина~~ 12,7 мм
ширина go 75 см

Каскадная лента

шарика 3,7 мкм

длиной от 25го 120 м

пергамент Fe_2O_3

швейцар



+++ - пасмурный
слой индикации

Перегородки

Доступ - это прямое
установление связи с устройством
хранения информации, где
записи и выбора данных.

Устройство прямого доступа
магнитные диски.

mark (анал) - отверстие

маркер размещение - автоматическая
ролька.

число деревенек (как) 9 деревенек
(как) 2 деревни

8 и 32 дама на них

63 и 246 дам на них

Сельскохозяйственная

80 дам = 5 и 5 м 8 м

63 дам = 44 и 7 дам

246 дам / 1 м = 160 и 6 дам

наименование - устройство зданий
общества различий между
ЭВМ и сельскохозяйственным
информационным центром, сельским

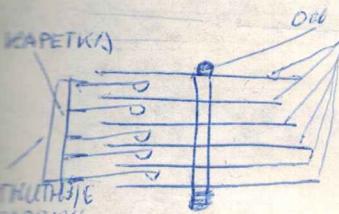
Магнитные диски
Каждая запись информации
имеет свой адрес

Магнитный диск является
носителем информации
прямого доступа.

Диски жёсткие, монолитные.

Таким образом

Жёсткие диски Al от 25го
65го

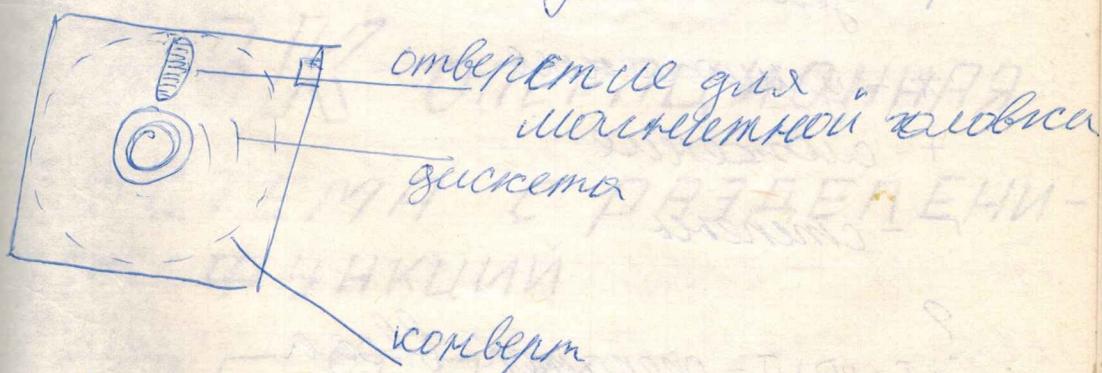


диски жёсткие

203 дорожки

200 используемых
человеком

3-суржедиц



Адрес дорожки, адрес замка,
ширина блока, промежуток
между замками.

This record made наст 1. gesch
Электро БС-5053
22 м волн 512 в бит

<u>Число</u>	<u>дисков</u>	<u>11</u>	<u>1</u>
<u>рабочих</u>	<u>поверхн.</u>	<u>20</u>	<u>1</u>
<u>плотность</u>	<u>леск. 300мкм</u>		
<u>Число</u>	<u>дорожек</u>	<u>200±30мкм</u>	<u>77</u>
<u>Число</u>	<u>секторов</u>	<u>—</u>	<u>26</u>
<u>Электро</u>	<u>сектора</u>	<u>—</u>	<u>128бит</u>

Мод-

- 1 - деление нацело
- \ - деление
- * - умножение
- + - сложение
- сменить

?

- * INPUT - оператор ввода
- ? LET - назначить или присвоить
- ? END - конец
- * RUN - запуск программы

10 INPUT x
20 LET $y = x * \cancel{x^2} - 4$ ($x^2 - 4$)
30 PRINT x, y
RUN ↓
AP2 5

- LIST - оператор чтения
- PRINT - печать

• (номер строки)
10, 20, 30

Программное обеспечение
ДВК ОПЕРАЦИОННАЯ
СИСТЕМА С РАЗДЕЛЕНИ-
ЕМ ФУНКЦИЙ

ОСДВК

формат команды:

COMMAND [OPT] INPUT [OPT1] OUT
PUT [OPT2]

или

CAMMAND [OPT]

PROMT1? INPUT [OPT1]

PROMT2? ~~INPUT~~ [OPT2]

зде

COMMAND - идентификатор
команды

OPT - квалифицированный параметр команды, уточняющий действие команды и относящийся ко всей командной строке.

INPUT - специализированная вход-
ного файла, содержащего
входную информацию.

1OPT1 - свалификатор файла,
нecessиций дополнительную ин-
формацию о введенном файле
OUTPUT - спецификация выход-
ного файла, содержащая
выходную информацию.

1OPT2 - свалификатор файла,
сообщающий дополнительную
информацию о введенном файле

PROMPT - текст напоминания

Символ "%" всегда рассматри-
вается как групповая операция,
но заменяется только символом
во введенной имене файла или
типе файла.

Допустимо одновременное
использование нескольких
знаков "%" в спецификации
файла.

"*" может заменять или имена файлов.

Звёздочка "*" во входной или имена файлов или типе файлов может заменять любое число

Команды ОПЕРАТОРА.

ABSLOAD - загружает в ОЗУ программы в форме абсолютной загрузки:

.ABSLLOAD FILE

такие программы могут быть расположены на любой носителе и после загрузки выполняются без операционной системы. Например:

.ABSLLOAD RKO:PROG1.LDA

Они предназначены для компьютеров, в состав которых не входят дисковые накопители.

ASSIGN

ASSIGN - присваиваем определимое пользователю, логическое или физическое устройству.

.ASSIGN DEV:NAM:

DEV - стандартное имя физического устройства

NAM - логическое имя устройства.

BOOT

BOOT - выполняет загрузку монитора операционной системы. Чтобы прочитать в память новую копию монитора необходимо выполнить команду

BOOTFILE

В этом примере спецификация
рамка представлена устрой-
ством или ини файлом иони-
того, подлежащего началь-
ной загрузке.

COMPILE

COMPILE - вызывает компилятор
или, если это необходимо, пос-
ледовательного несколько каш-
тийоров для трансляции ~~или~~
исходных файлов:

COMPILE [/OPT] FILES[OPT]

Ключи команды COMPILE

/ALLOCATE: SIZE

этот ключ используется
с ключом /LIST или /OBJECT
для резервирования области
на устройстве для

входного радио. Адресицем
SIZE (размер) представляем
с собой число резервируемых
байтов. Это значение изменя-
ется в пределах от 1 до 32767.
значение равное "-1" создаёт
на устройстве радио макси-
мального размера.

(1) * 8040000

IF T% = 4 THEN DL% = 0

PO 20.9372° 85.8078° 87.9070° 05

(1) M12 * 02 + 00 (T) S03 * 02 + 00

IF T% < 5 GOTO 04 NEXT

GOSUB 186

T% = 9

IF KND(1) < 6 THEN T%

~~KV~~ ~~IV~~
~~KOBOA~~

OO OO
VIRUS

10 CLS

20 FOR A = 80 TO 261

30 POKE A,A

40 NEXT

50 BEEP

60 GOTO 50

0

10 CLS

20 FOR T=0 TO 6.28 STEP.05

30 PSET(120+50*COS(T), 90+50*SIN(T))

40 NEXT

50 GOTO 50

~~RR% = 8054500~~

~~R = P~~

~~T% = 0~~

~~T% = T% + 1~~

FOR A% = RR% TO RR% + &01

500 ST &02

IF T% = 1 THEN READ DL%

IF T% = 2 THEN DL% = RND

(1)* &0777777

IF T% = 3 THEN DL% = RND

(1)* &0400000

IF T% = 4 THEN DL% = 0

POKE A% + N%, DL%

NEXT N%, A%

IF T% < 5 GOTO 108

GOSUB 186

T% = 9

IF RND(1) < 0.6 THEN T% =

~~РС~~

& 0177600, &0777, &017

ЗАГРУЗКА В ДВК системы РОДОС



1) Панель

2)

3)

Нажмите (2 и 3) и

поставте в исходное

состояние.

экран:

Нажмите (B)

~~автоматика~~

~~автоматика~~

Надерите МХØ

~~автоматика~~

~~автоматика~~

всегда

~~автоматика~~

~~автоматика~~

DATA (44-AM-F5)

наберите число - месяц - 209

и два раза **(BK)**

После появления ":" наберите ASS

MX1 DK и **(BK)**

R NET^①

Если хотите просмотреть список файлов хранимых на диске
наберите DIR и **(BK)**

Для просмотра ~~этой~~ списка
стёртых файлов используется
команда DIR DEL **(BK)**

Переименовка файлов осуществляется
с помощью команды REN

REN **(BK)**

FILE? имя файла

TO? новое имя файла

Для удаления информации из
данного файла используется
команда DEL

DEL имя файла

Для просмотра содержание
файла используется команда
TYPE

TYPE

~~ERAN~~ FILE? (им файла) ⌂

Переход в монитор.

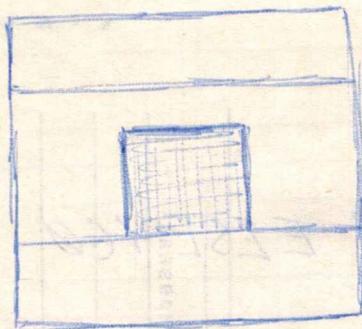
R DIR ⌂

Выход из монитора.

• нажать одновременно

[CY] [C]

SanYO



10 CLS

20 FOR A% = #&040000 TO &077777

30 POKE A%, 80000000

40 NEXT

50 FOR A% = #&0401000 TO &041000 +

(100 * 80100)

60 POKE A% #, 80100001 * RND(1) +

17771

70 NEXT

80 FOR A% = (x) TO (y)

90 POKE A%, 1

100 NEXT

110 ~~10~~ X = PEEK(1047)

120 IF X = 1 THEN 130 ELSE 10

130 FOR A% = (x) TO (y)

140 POKE A%, 0

150 NEXT

160 X = PEEK(1047)

170 IF X = 1 THEN 80 ELSE 160

10 CLS

20 FOR A% = &040000 TO &077777

30 POKE A%, 0

40 NEXT

50 X = PEEK(&01047)

55 IF X = 1 THEN 60 ELSE 50

60 FOR A% = &040000 TO &077777

ST &05

70 POKE A%, &1001 * RND(1)

80 NEXT

90 X = PEEK(&01047)

100 FOR A% = &040000 TO &077777

ST &05

110 POKE A%, 1

120 NEXT

130 PRINT AT (100, 100); "КОСМОС"
140 CLS
150 OR A2 = 1
160 CLS
200 DRAW "BM100, 100;
300 PAINT (102, 98), 1
400 PRINT "ЭТО ВАШ САМОЛЕТ"
600 FOR A% = !x! TO !y! ... ST 80100
700 FOR B% = A% TO A% + &0 ...
800 X(n) = PEEK(801047)
850 N = N + 1
900 NEXT
1000 NEXT TOT
1100 CLS
1200 FOR M% = 80 ... TO ... ST 802
1300 ~~A1 = M1~~
1400 Q = Q + 1
1500 ~~135 FOR A% = M% TO M% + ... ST 80100~~
1500 FOR B% = A% TO A% + &0 ...
1600 ~~170 POKE B%, X(Q)~~
1800 NEXT
1900 NEXT

южнокорейская строка 255 символов.

200 ~~Y=PEEK(8701047)~~
210 ~~IF J=N THEN W~~
220 ~~FOR Y% = M%. + &02 TO ... ST 80100~~
230 ~~PEEK(Y%. + &0100)~~
240 ~~IF Q>R THEN R ELSE 250~~
250 ~~POKE Y%. - &03~~
260 ~~GOTO 220~~
270 ~~POKE Y%. - &02, &0111 * RND(1)~~
280 ~~Y%. = Y%. + &0100~~
290 ~~IF YY. = THEN . ELSE 270~~
300 ~~NEXT~~
310 ~~FOR K%. = 0 TO Y%. + &0100 TO ... ST 80100~~
320 ~~POKE D%. = K%, &0000~~
330 ~~NEXT~~
340 ~~END~~

~~10~~ 20 DIMAXC(32,20)
25 GOSUB 690
30 FOR A1=1 TO 20
40 FOR A2=1 TO 32
50 READ X α
60 A α C(A1,A2)=X α
55 PRINT AT(A1,A2);A α CA
70 NEXT
80 NEXT
90 DATA
100 DATA
110 DATA
120 DATA
130 DATA
140 DATA
150 DATA
160 Z α C=INKEY> α
170 IF Z α C = " " THEN 190
180 K=ASC(Z α)
190 IF Q=1 THEN 360 ELSE 200
200 IF K= THEN B=ABS(B+1)
210 IF K= THEN B=ABS(B-1)
220 IF Q1=1 THEN 230 ELSE 330
230 IF K= THEN C=ABS(C+1)

- 33 \emptyset IF $Q2=1$ THEN 34 \emptyset ELSE
35 \emptyset
- 34 \emptyset IF $R =$ THEN $C = ABS(C-1)$
- 35 \emptyset SOTO 37 \emptyset
- 36 \emptyset $C = C + 1$
- 37 \emptyset IF $AD\langle(B, C+1) < > \dots \rangle$ THEN
 $Q = \emptyset$ ELSE $Q = 1$
- 38 \emptyset IF $AD\langle(B, C+1) = "H"$ THEN $Q1 = 1$
ELSE $Q1 = \emptyset$
- 39 \emptyset IF $AD\langle(B, C-1) = "H"$ THEN $Q2 = 1$
ELSE $Q2 = \emptyset$
- 40 \emptyset IF $AD\langle(M, M1+1) < > \dots \rangle$ THEN
 $Q3 = \emptyset$ ELSE $Q3 = 1$
- 41 \emptyset IF $AD\langle(M, M1+1) = "H"$ THEN
 $Q4 = 1$ ELSE $Q4 = \emptyset$
- 42 \emptyset IF $AD\langle(M, M1-1) = "H"$ THEN $Q5 = 1$
ELSE $Q5 = \emptyset$
- 43 \emptyset IF $Q3 = \emptyset$ THEN $M1 = ABS(M1+1)$
SOTO 50 \emptyset
- 44 \emptyset IF $M > B$ THEN $M = ABS(M - 1)$
- 45 \emptyset IF $M < B$ THEN $M = ABS(M + 1)$
- 46 \emptyset IF $Q4 = 1$ THEN 44 \emptyset ELSE

470 IF M1 < C THEN M1 = ABS(M1+1)
480 IF Q5 = 1 THEN N 460 ELSE
490 IF M1 > C THEN M1 = ABS(M1-1)
500 IF (B=M) AND (C=M1) THEN GOTO
510 PRINT #AT(B,C), "♦"
520 PRINT AT(M,M1); "X"
530 GOTO 155
540 REM КОНЕЦ
550 PRINT "AT(1,20); " ВАС ПОЙМАЛ
ЛЮДОЕД"
560 BEEP
570 FOR L=1 TO 10
580 ~~NEXT~~
590 FOR L=8040000 TO 8077777
600 POKE L, 80101010 * RND(1) + 1771
610 NEXT
620 ~~FOR~~ L=8040000 TO 8077777
630 POKE L, 8000000000
640 NEXT
650 INPUT "ВЫ БУДЕТЕ ЕЩЁ ИГРАТЬ
(D/N)"; L⁹
660 IF L⁹=D THEN GOSUB 680 ELSE
670

670 PRINT "ОЧЕНЬ ХАД"
680 ~~END~~
690 CLS
700 DRAW
710 DRAW C+1♦
720 DRAW A+B(MAT)
730 DRAW SE Q=1
740 DRAW A+B(C-1)
750 DRAW
760 PAINT
770 FOR P= TO ST
780 PAINT(,) -
790 NEXT
800 FOR P=1 TO 12
810 BEEP
820 NEXT
830 RETURN



РАСПРЕДЕЛЕНИЕ АРРЕСНОГО ПРОСТРАНСТВА МИКРО ЭВМ.

	Системная область и стек	16 кбайт
1000	ОЗУ пользователя	
40000	ОЗУ пользователя или экрана	12 кбайт
70000	ОЗУ экрана	4 кбайт
100000	Системное ПЗУ (монитор и драйверы)	8 кбайт
120000	Сменные ПЗУ	8 кбайт
140000		8 кбайт
160000		8 кбайт
177600	Системные регистры (регистры внешних устройств)	8 кбайт

РЕГИСТР УПРАВЛЕНИЯ СИСТЕМНЫМИ ВНЕШНИМИ УСТРОЙСТВАМИ

177716

2 регистра

1 по записи

2 по чтению

РЕГИСТР СМЕЩЕНИЯ

177664

ПОСЛЕ ЗАПУСКА ПРОЦЕССОРА 1330

РЕГИСТРЫ КЛАВИАТУРЫ

РЕГИСТР СОСТОЯНИЯ КЛ.

177660

РЕГИСТР ДАННЫХ КЛ.

177662

РЕГИСТР ПОРТА ВВОДА-ВЫВОДА

177714

(МОНТОР Н ПРАВЕПРИ)

СЧЕТЧИК

У2А

(БЛАНКИ РЕГИСТРОВ ГЕНЕРАТОРЫ)

Дано a, b, x, y, z
определим

$$t = \begin{cases} \sqrt{\sin^2 a + \sin^2 b} & \text{при } x > 0, y > 0, \\ 1 + \sin a + \sin b & \text{при } z > 0 \\ \cos^2 a + \cos^2 b & \text{при } x = 0, y = 0, z = 0 \end{cases}$$

в остальных
случаях

дан x бросаем $y = f(x)$ в
заданной монотонности по
порядку

$$y_{j+1} = y_j + \frac{1}{2} \left(\frac{x}{y_j} - y_j \right) \quad (t = 0, 1, \dots)$$

Начальное значение $y_0 = 1$.
предыдущая монотонность если $|y_{t+1} - y_t| \leq \varepsilon$

X - код операции

Y - имена операндов аргументов

S - сумматор

Op - операция

Kod X

Операция

0

OP - > S

FGET

2

S + OP -> S

FADD

3

S - OP -> S

FSUB

4

S / OP -> S

FDIV

5

S ^ OP -> S

FMUL

6

S -> OP

FPUT

Kod Y

Аргументы
аргент в следующем
слово

МНЕМОНИКА
#OP

0

аргент в R2

1

(R2)+4, аргент в R2

2

Операнд в вершине

стенка

4 Адрес в вершине
стенка

5 Операнд в следующем
двоичном слове
6 Переобратаем

Переводимого с подачи

7074 заданием одномерст-
вие операции над содержимым
одинакового регистра.

команды Операция Индекс регистра

7070 нормализация FNOR
числа в S

7071 Целая часть $S \rightarrow R1$ FINT

7072 $\text{sgn}(S) \rightarrow S$ FSGN

7073 $AB S(S) \rightarrow S$ FABS

7074 $-S \rightarrow S$ FNEG

7075 Перевод числа заданного
последовательностью строк
6S FREAD

7076 вывод S на экран PRINT
R2 - формат вывода

Программное обеспечение (ПО) - это совокупность программ позволяющих организовать решение задач по изображению на COMPUTER

С ПО (специальное) - совокупность программ организующих процесс обработки информации на компьютере

Файл - совокупность одинаковых записей и данных записанных на внешних носителях.

СТЕК - совокупность связанных между собой регистров для хранения

упорядоченных данных.

Шина - группа проводов обеих одинаковых битов и функционирования признаком при передаче информации.

5 REM 'ПРОГРАММА СОРТИРОВКИ ПО ВОЗРАСТАНИЮ' или убыванию

10 CLS

20 INPUT N

30 DIM A(N), B(N)

40 FOR K=1 TO N

50 INPUT A(K)

60 NEXT

70 FOR K=1 TO N

80 FOR K1=1 TO N

90 IF A(K) > A(K1) THEN M=M+1

100 NEXT

110 I=N-M

120 B(I)=A(K)

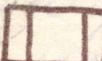
13 Ø NEXT

14 Ø FOR K=1 TO N

15 Ø PRINT B(K)

16 Ø NEXT

17 Ø GOTO 1Ø

- ✓  - процесс
-  - решение
-  - модификация
-  - предопределённый процесс
-  - ручная операция
-  - вспомогательная операция
-  - слияние
-  - выделение
-  - группировка
-  - сортировка
-  - ручной ввод

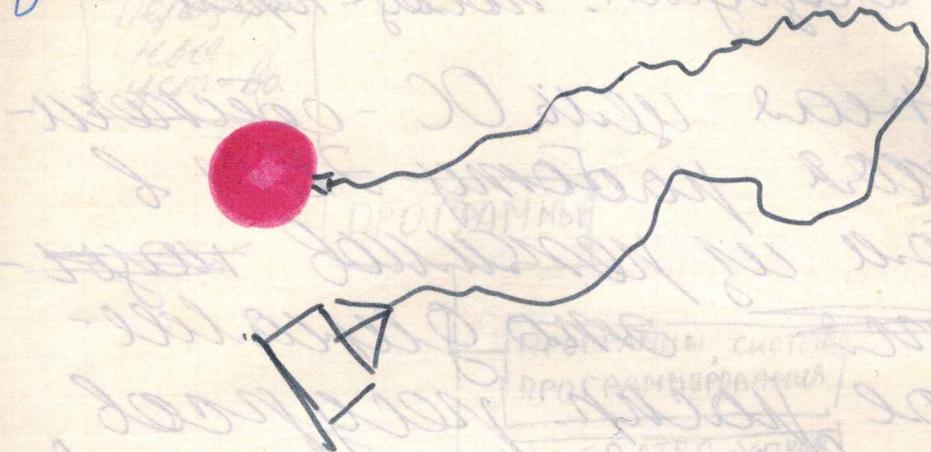
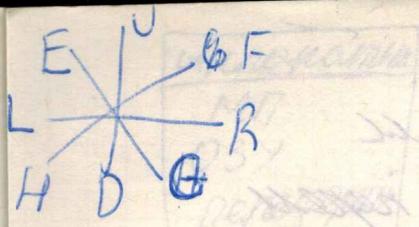
-  - ввод-вывод
-  - неавтоматическая память
-  - автономная память
-  - документ
-  - перфокарта
-  - колода перфокарт
-  - файл
-  - перфолента
-  - магнитная лента
-  - магнитный барабан
-  - магнитный диск
-  - ОЗУ
-  - дисплей

-- линия потока = ПАРАЛЛЕЛЬНОЕ ДЕЙСТВИЕ
 ~ - канал связи ✓ - ПУСК-ОСТАНОВ
 ---[-КОМЕНТАРИЙ] □ - МЕЖСТРАНИЧНЫЙ СОЕДИНИТЕЛЬ
 △ - ИСТОЧНИК-ПРИЕМНИК ДАННЫХ □ - МАГНИТНАЯ КАРТА

```

10 REM 'ВВОД'
20 CLS
30 DIM A$$(250, 100)
40 OPEN "TT: DOUTS"
50 FOR A=1 TO 250
60 FOR B=1 TO 100
70 INPUT #A$(A,B)
80 NEXT
90 NEXT
100 FOR A=1 TO 250
110 FOR B=1 TO 100
120 PRINT #A$(A,B)
130 NEXT
  
```

10 DATA 8012702, &00, 8012700,
&00, &010004, &012737, &0100, &01777
16, &077401, &012737, &00, &177716, &0
10004, &077001, &077215, &0207
20 FOR A = &04000 TO &04000
50 ST &02
25 DATA (romed)
30 READ K
40 POKE A, K
50 NEXT
60 DEF USR1=(&01000)
70 FOR Z=1 TO 17
75 READ N
80 POKE &01006, Z00-N
90 POKE &01002, 401199
95 A=USR1(&01000)
100 NEXT



• Назначение и состав
ОС микро ЭВМ

- ОС - может обеспечивать
работу микро ЭВМ в 1-м
из 3-х режимов.
- 1) Однокропочном
 - 2) Многокропочном / мульти-
прогр.)
 - 3) Многозадачном.

В однокпр. послед.

В двухтпр. паралл.

В многоуровн. посл-перен.

Основная цель ОС - обеспечи-
ванием работы ЭВМ в
многом из нежел. ~~негат-~~
~~важном~~ Это динамич-
еские расп. ресурсов
и упрощение или в соотв.
с требованием быстродействия
ресурсов является важнейший
объект который можно
распределить ОС между
вн. процессорами и микро ЭВМ.

Ресурсы ЭВМ

программы

аппаратные

Аппаратура

МП
ОЗУ
переводчик
кодов
уст.-ва

ПРОГРАММНЫЙ

ПРОГРАММЫ, СИСТЕМЫ
ПРОГРАММИРОВАНИЯ

СРЕДСТВА УПРАВЛЕНИЯ УСТРОЙСТВАМИ И ФАЙЛАМИ

БИБЛИОТЕКИ ПРОГРАММ

СРЕДСТВ УПРАВЛЕНИЯ ЗАДАЧАМИ

ТАКЖЕ ВХОДНОЙ СИГНАЛ ПРЕСТАВЛЕНИЯ ПРЕРЫВАНИЯ

ТАКОЕ ВЫХОДНОЙ СИГНАЛ ПРЕДОСТАВЛЕНИЯ ПРЕРЫВАНИЯ.

VTRQ - ТРЕБОВАНИЕ ПРЕРЫВАНИЯ

AD15 - линия АДРЕСА АДДИСИ

WTBT - ВЫБОР БАЙТА

10CLS

20 DRAW "BM. ; D100; R100; U100; L100;
30 DRAW "BM ; D80; R80; U80; L80"

40 CIR(,),2

50 CIR(,),2

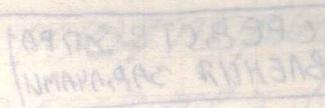
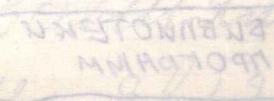
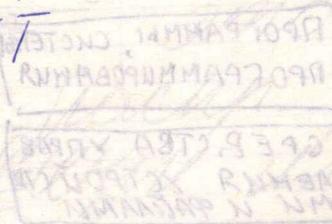
60 CIR(,),2

70 CIR(,),2

80 PAINT

90 PAINT

100



Задание №1

необходимо

анализировать

A1	IRQ1	Останов
A2	-	Общий
A3	-	»
A4	-	Контроль ист. питания +5В
A5	IRQ2	ТРЕБОВАНИЕ ПРЕРЫВА- НИЯ
A23	DIN	ВВОД ДАННЫХ
A25	AD13	Линия АДРЕСА ДАННЫХ
A26	AD11	» » »
A27	« AD9	» » »
A28	« AD5	» » »
A31	« AD0	» » »
B2	« -	Общий
B3	« -	»
B4	« ИАКО	Входной сигнал ПРЕ- СТАВЛЕНИЯ ПРЕРЫВАНИЯ
B4	« ИАКО	Выходной сигнал ПРЕДОСТАВЛЕНИЯ ПРЕРЫ- ВАНИЯ.
B5	« VIRO	ТРЕБОВАНИЕ ПРЕРЫ- ВАНИЯ
B7	AD15	Линия АДРЕСА ДАННЫХ
B11	WTBT	ВЫВОД БАЙТА

B19

INIT ПЕРВОНАЧАЛЬ-
НАЯ УСТАНОВКА МАГИ-
СТРАЛИ.

B20

RPLY СИГНАЛ СИНХРО-
НИЗАЦИИ ПАССИВ-
НОГО УС-ВА (ОТВЕТ)

B21

DOUT ВЫВОД ДАННЫХ

B22

SYNC СИГНАЛ СИНХРО-
НИЗАЦИИ АКТИВНОГО
УСТ-ВА.

B23

« AD11DA » ЛИНИЯ АДРЕСА ДАННЫХ

B24

« AD92DA » >> 55 >>

B25

« AD70DA » >> 85 >>

B26

« AD80DA » >> 7E >>

B27

« AD60AD 6 - » >> 5 >>

B28

« AD 4 - » >> 8 >>

B29

« AD21AI » >> 4 >>

B30

« AD3NA » >> 8 >>

B31

« AD1 » >> >>

B32

« AD7 » >> 78 >>

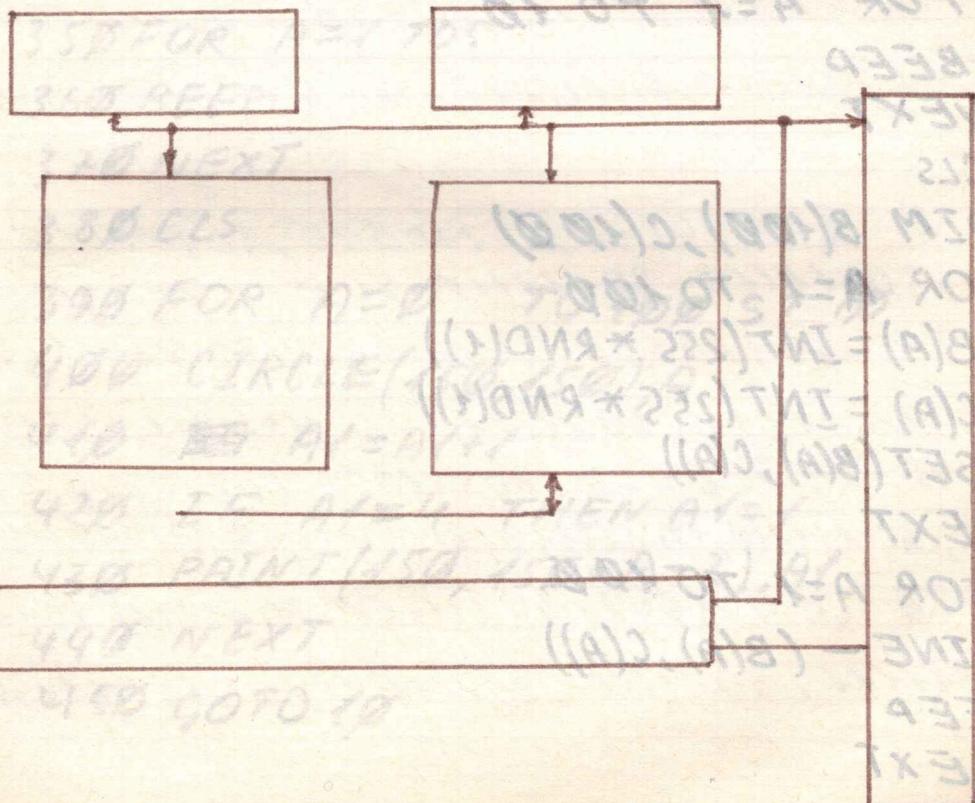
W8T8W 88168 88168 88168

СИСТЕМНАЯ МАГИСТРАЛЬ ТИПА Q-бн (МДИ, ОСТ 11.305.903-80)

А - ВЕРХНИЙ РЯД КОНТАКТОВ.

В - НИЖНИЙ.

СТРУКТУРНАЯ ОРГАНИЗАЦИЯ МИКРОПРОЦЕССОРА К1801ВМ1



10 CLS

20 B=170

30 FOR A=170 TO 255 ST 2

40 B=B-2

50 LINE(170, 0)-(A, 150)

60 LINE(A, 150)-(170, 255)

70 LINE(170, 255)-(B, 150)

80 LINE(B, 150)-(170, 0)

90 NEXT

100 FOR A=1 TO 10

110 BEEP

120 NEXT

130 CLS

140 DIM B(100), C(100)

150 FOR A=1 TO 100

160 B(A)=INT(255*RND(1))

170 C(A)=INT(255*RND(1))

180 PSET(B(A), C(A))

190 NEXT

200 FOR A=1 TO 100

210 LINE-(B(A), C(A))

220 BEEP

230 NEXT

240 CLS

250 FOR A=0 TO 255 ST 4

260 LINE(0+A,0)-(255,A+A)

270 LINE(255,0+A)-(255-A,255)

280 LINE(255-A,255)-(0,255-A)

290 LINE(0,255-A)-(0+A,0)

300 ~~END~~ NEXT

310 FOR A=1 TO 5

320 BEEP

330 NEXT

340 PAINT(155,160)

350 FOR A=1 TO 5

360 BEEP

370 NEXT

380 CLS

390 FOR A=0 TO 100 ST 10

400 CIRCLE(150,150),A

410 ~~END~~ A1=A1+1

420 IF A1=4 THEN A1=1

430 PAINT(150,150+A-2),A1

440 NEXT

450 GOTO 10

220 IF C=6 THEN 230 ELSE +50

ПРОГРАММА „ФЛАГ“

```
10 CLS
20 LINE( , )-( , )
30 DATA &0 000011, &0100111, &011111,
&0111111, &0111111, &0111111, &0111100, &0
011000
40 DATA &0000010, &0100110, &0111110,
&0111110, &0111110, &0111110, &0111100, &001
1000
50 DATA &0100001, &0110011, &0111111, &0
111111, &0111111, &0111111, &00111110, &0011
10
60 DATA &0100000, &0110010, &0111110, &0
111110, &0111110, &0111110, &0011110, &00001100
70 DATA &0110000, &0111001, &0111111, &0
111111, &0111111, &0111111, &0001111, &0000110
80 DATA &0110000, &0111000, &0111110, &0
111110, &0111110, &0111110, &0001110, &00000110
90 DATA &0011000, &0111100, &0111111, &0111111,
&0111111, &0111111, &0111111, &0100111, &0
0000111
```

100 DATA 80011000, 80111100, 80111111,
80111110, 80111110, 80111110, 80100110, 80000
010

110 DATA 80001100, 80011110, 80111111, 80111111,
80111111, 80111111, 80110011, 80100001

120 DATA 80001100, 80011110, 80111110,
80111110, 80111110, 80111110, 80110010, 801
00000

130 DATA 80000110, 80001111, 80111111, 80
111111, 80111111, 80111111, 80111001, 80110000

140 DATA 80000110, 80001110, 80111110,
80111110, 80111110, 80111110, 80111000, 80110000

150 FOR A% = &0 TO &0

160 FOR K% = A% TO A% + &01000 ST &0100

170 READ X%

180 POKE K%, X%

190 NEXT

200 NEXT

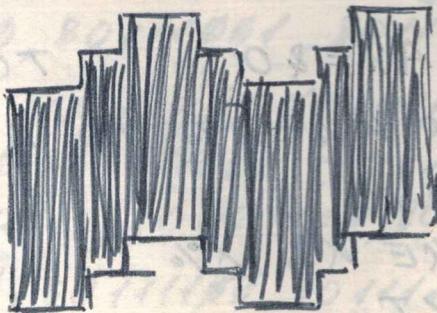
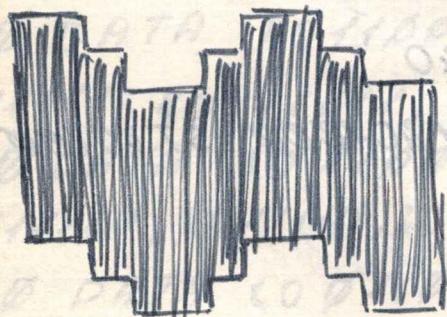
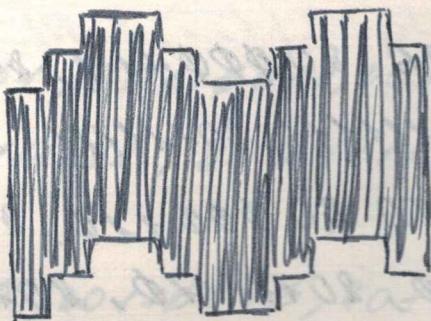
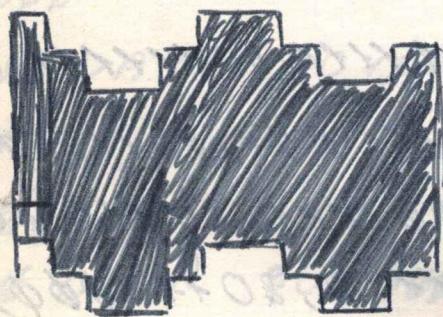
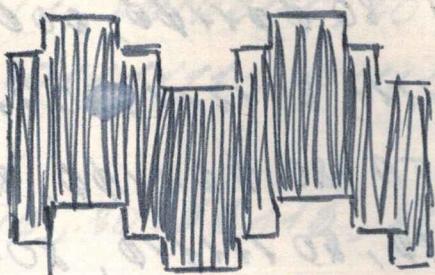
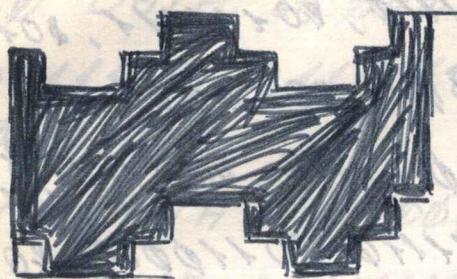
210 C = C + 1

220 IF C = 6 THEN 230 ELSE 150

~~3 Ø C = Ø~~

~~24 Ø RESTORE~~

~~25 Ø GOTO 15Ø~~

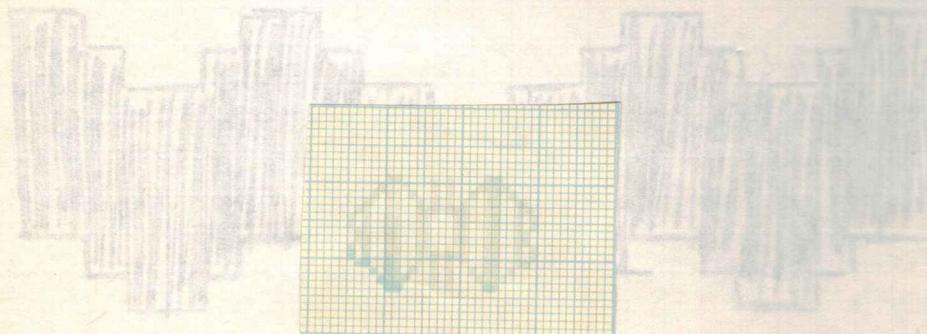
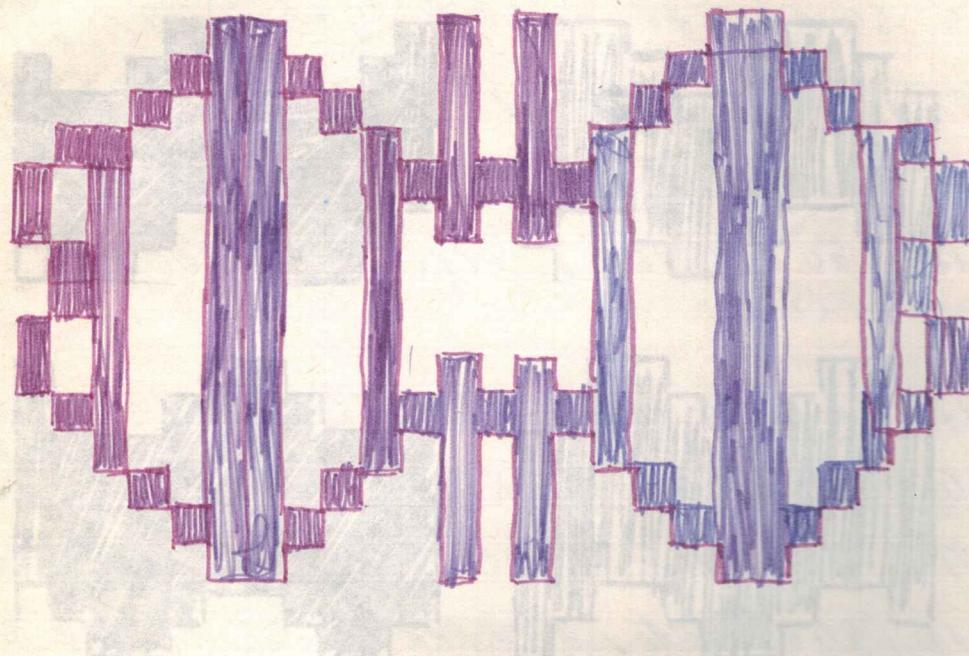


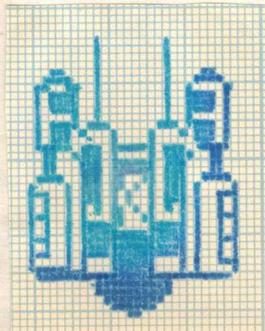
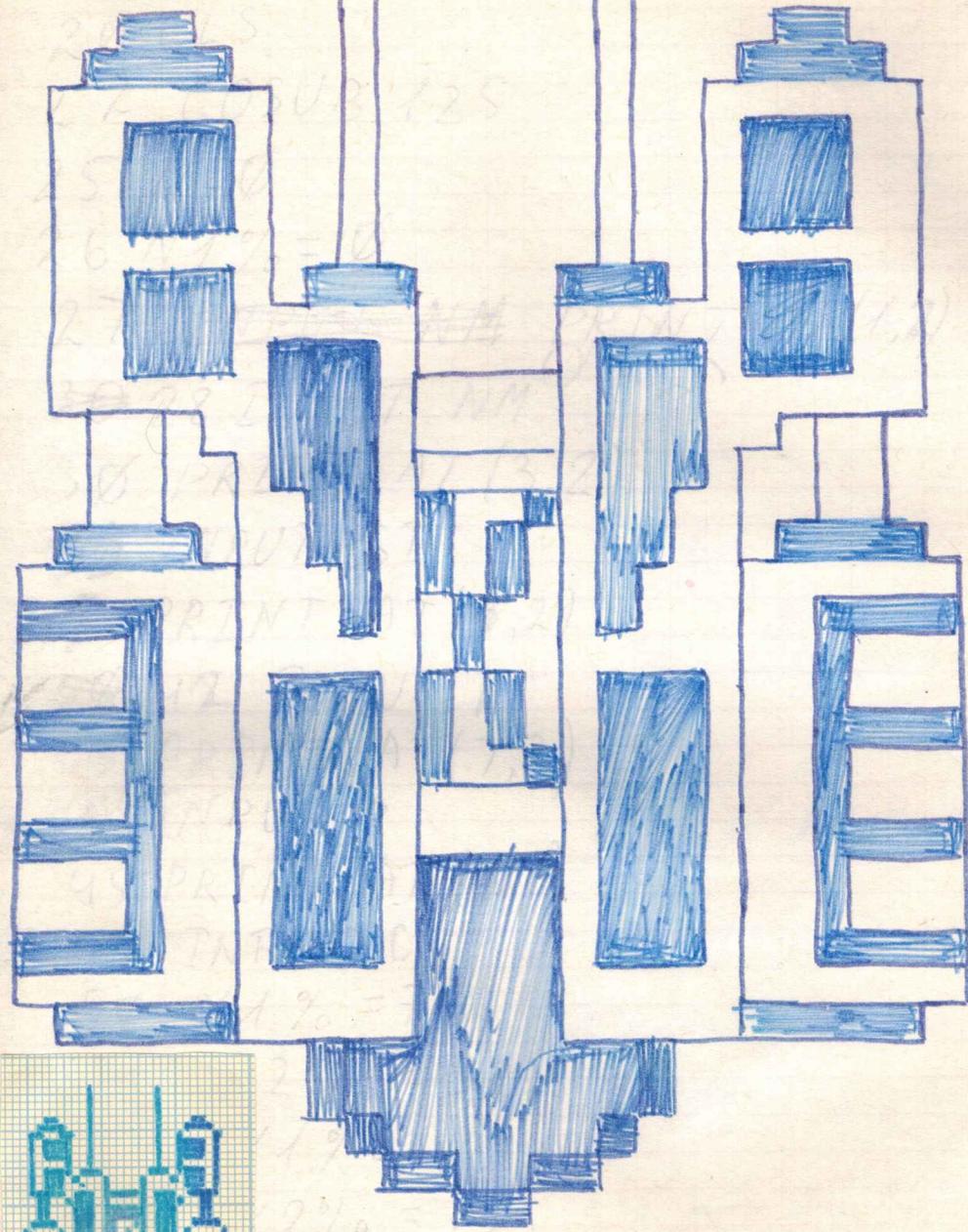
Q27 3253 0853 2=3 75 055
 $3 \times 0 C = C$

30 C-8

240 RESTORE

250 SOTO 150





X 3% - (X 2% - X 1%) / 12

ТЕКСТ ПРОГРАММЫ УЗОРЫ (COVER)

20 CLS

22 GOSUB 125

25 N=0

26 N1% = 0

27 ~~INPUT NM~~ PRINT AT(1,2)

~~28 INPUT NM~~

30 PRINT AT(3,2)

~~31 INPUT SI~~

40 PRINT AT(5,2)

~~42 INPUT A~~

43 PRINT AT(7,2)

44 INPUT B

45 PRINT AT(9,2)

46 INPUT C

50 X1% = 70

51 X2% = 319

52 Y1% = 0

53 Y2% = 199

54 XM% = (X2% - X1%) / 12

- 55 $XD\% = X2\% - X1\%$
- 56 $YD\% = Y2\% - Y1\%$
- 57 $YM\% = (Y2\% - Y1\%) / 2$
- 58 $XS\% = XD\% / 250$
- 59 $YS\% = YD\% / 200$
- 60 $X = \emptyset$ AND $Y = \emptyset$
- 61 ~~GOSUB~~ CLS : GOSUB 125
- 62 LINE $(X1\%, Y1\%) - (X2\%, Y2\%), 2, B$
- 67 $XA\% = XM\% + Y * XS\% * SJ +$
- 75
- 68 $YA\% = YM\% + Y * YS\% * SJ +$
- 75
- 69 IF $XA\% < X1\% \text{ OR } XA\% >$
 $X2\% \text{ OR } YA\% < Y1\% \text{ OR } YA\% >$
 $Y2\% \text{ THEN GOTO 72}$
- 71 PSET $(XA\%, YA\%), N1\%$
- 72 IF $N = NM\# \text{ THEN GOSUB}$
125

73 GOTO 105

75 $XX = Y - SGN(X) * SQR(ABS(B * X - C))$

80 $YY = A - X$

83 $N = N + 1$

85 $X = XX$

86 $Y = YY$

87 $N1\% = N1\% + 1$

90 $N1\% = N1\% - INT(N1\% / 4) * 4$

100 GOTO 67

105 PRINT AT(15, 1)

106 INPUT # "OK?" V

107 PRINT AT(15, 2), "

108 GOSUB 125

109 GOTO 25

125 PRINT AT(1, 1); "N= "

126 PRINT AT(2, 1); NM

127 PRINT AT(4,1); "S = ~~\$~~" EF
128 PRINT AT(5,1); \$I
129 PRINT AT(6,1); "A = " X X 8
130 PRINT AT(7,1); A
131 PRINT AT(8,1); "B = " E 8
132 PRINT AT(9,1); B
133 PRINT AT(10,1); "C = " 8
134 PRINT AT(11,1); C
135 RETURN

ТЕКСТ ПРОГРАММЫ ТОЧКИ
(POINTS)

1 ~~N~~ $N\% = 100$

2 $T = 2$

3 $A = 1$

4 $B = 1$

5 $N1\% = \emptyset$

11 GOSUB 500

15 $S\bar{I} = 6 * T + 1$

20 $X2\% = 30$

21 $X1\% = 319$

22 $Y2\% = 199$

23 $Y1\% = \emptyset$

30 $XD\% = X2\% - X1\%$

43 1 $XS\% = XD\% / S\bar{I}$

40 $YD\% = Y2\% - Y1\%$

41 $YS = YD\% / S\bar{I}$

42 $XM\% = (X2\% + X1\%) / 2$

44 $YM\% = (Y2\% + Y1\%) / 2$

50CLS

60LINE (X1%, Y1%) - (X2%, Y2%)

, 2,B

65 LINE (XM%, YM%) - (XM%, YM%),

67 LINE (X1%, YM%) - (X2%, YM%),

70 X = Ø

71 Y = Ø

72 GOSUB 500

80 FOR I% = 1 TO N%

90 PRINT AT(1,1); I%

100 ~~X3 = X~~

101 Y3 = Y

110 X = (X3 + T) * SIN(Y3)

120 Y = R - B * ~~X3~~

130 X4% = XM% + XS * X

140 $Y4\% = YM\% + YS * Y$
150 PSET ($X4\%$, $Y4\%$), $N1\%$
155 $N1\% = N1\% + 1$
156 IF $N1\% = 4$ THEN 160
160 NEXT
~~200 GOSUB 800~~
~~210 GOSUB 830~~
215 INPUT V_{XK}
220 IF $\{V_{XK} = "T"\}$ THEN 310
230 IF $V_{XK} = "A"$ THEN 330
240 IF $V_{XK} = "B"$ THEN 350
250 IF $V_{XK} = "N"$ THEN 380
260 IF $V_{XK} = "C"$ THEN 15
300 GOTO 215
310 PRINT AT(17, 1)
320 INPUT T
325 RETURN
330 PRINT AT(17, 1)
335 INPUT A
340 RETURN

350 PRINT AT(17,1)
360 INPUT B
370 RETURN
380 PRINT AT(17,1)
390 INPUT N%
400 RETURN
500 PRINT AT(3,1); "T = "; T
510 PRINT AT(6,1); "A = "; A
520 PRINT AT(9,1); "B = "; B
530 PRINT AT(12,1); "N = "; N%
540 RETURN

ПРОГРАММА ГЕНЕТИКА (GENETIK)

1 DIM A(100), B(100), M(3,3), F(9), P(9)

3 M(1,1)=4

4 M(2,2)=4

5 M(3,3)=4

6 M(1,3)=4

7 M(3,1)=4

8 M(2,3)=4

9 M(2,1)=4

10 M(1,2)=4

11 M(3,2)=4

12 DATA $\phi_6, \phi_4, \phi_2, \phi_5, \phi_5, \phi_3,$
 $\phi_{.55}, \phi_{.55}, \phi_{.3}$

13 FOR Z=0 TO 9

14 READ V

15 P(Z)=V

16 NEXT

17 DATA 1,2,3,2,2,3,3,3,3

18 FOR Z=0 TO 9

19 ~~E(Z)=~~ READ V

20 F(Z)=V

21 NEXT
22 PRINT "STANDARD RULES"
23 INPUT A₂₀
24 IF A₂₀ = "Y" THEN 90
25 FOR I = 1 TO 3
30 FOR J = 1 TO 3
40 PRINT I; "*"; J; "SUNS"
41 INPUT M(I,J)
50 PRINT "TYPE OF SUNS AND
PROB."
51 L = (I - 1) * 3 + J
60 INPUT F(L), P(L)
70 NEXT
80 NEXT
90 ~~PRINT~~ INPUT "LENGTH"

L
100 PRINT "INITIAL POPULA
RISU"
110 FOR I = 1 TO L
120 INPUT B(I)
130 NEXT I

135 R = 1

136 CLS

137 140 M = 50 - INT(L/2)

150 FOR I = 1 TO L

160 A(M+I) = B(I) -

161 X1 = (M+I-1)*3

162 Y1 = (R-1)*3

163 LINE(X1, Y1) - (X1+3, Y1+3),
B(I), BF

164 B(I) = Ø

170 NEXT I

180 L1 = 1

190 FOR I = 1 TO 99

200 I1 = A(I)

203 ~~IF~~ Y1 = A(I+1)

205 IF L1 > 100 THEN L1 = 100

206 GOTO 235

207 210 IF I1 = Ø OR Y1 = Ø THEN

B(L1) = Ø

215 GOTO 230

220 GOSUB ~~380~~ 380

230 NEXT I

235 L = L1

240 S = 0

3250 FOR I=1 TO L

260 IF BB(I) = A(I) THEN

S = S + 1

270 NEXT I

280 IF L = S THEN GOTO 350

290 R = R + 1

300 IF R = 60 THEN GOTO

330

310 GOTO 140

330 ~~PRINT~~ AT(24,1)

335 INPUT ~~A~~ "NEXT SCREEN"

A0,

340 IF A0 = "Y" THEN GOTO

135

350 PRINT AT (24,15)

355 INPUT A_{xx}

360 IF A_{xx} = "Y" THEN GOT0 3
ELSE STOP

380 FOR J=1 TO M(I1,Y1)

382 JJ = (I1-1)*3 + Y1

383 IF L1 > 100 THEN

GOT0 420

385 RR = RND(1)

386 IF RR < P(JJ) THEN R1 = 0

387 GOT0 397

390 R1 = INT (F(JJ)*RND(1))

+1

397 B(L1) = R1

400 L1 = L1+1

410 NEXT J

420 RETURN

ПРОГРАММА ЖИЗНЬ (LIFE)

1 DIM A(60,60), B(60,60), C(24,2)

2 S(3)

3 LK=1

4 L1=51

5 INPUT "TYPE"; T

6 IF T=1 THEN L=4

7 IF T=2 THEN L=8

8 INPUT "SIZE"; N

9 SI=INT(160/N)

10 HS=80

11 YS=20

12 INPUT "ROLE OF LIFE"; RL

13 INPUT "TYPE (RANDOM, REGULAR)"; I

14 IF I=0 THEN GOTO 140

15 PRINT "FIRST POINT"

16 GOTO 50

16 CLS
17 $X_1 = X_S + N * 3$
18 $Y_1 = Y_S + N * 3$
19 LINE ($X_S - 1, Y_S - 1$) $\rightarrow X_S + N * SI + 1,$
 $Y_S + N * SI + 1), 3, BF$
20 DATA -1, 0, 1, 0, -1, 1, 1, -1, 0, -1,
0, -1, -1, 1, 1
21 FOR Z=0 TO 8 ~~87 16~~
22 READ V
23 C(Z)=V
24 NEXT
25 FOR Z=0 TO 8
26 READ V
27 C(Z, 2)=V
28 NEXT
29 GOTO 375
~~30~~ 50 INPUT L0
60 FOR I=1 TO N
70 R=I+L0-1 ~~1~~
71 L2=R-INT(R/3)*3
80 FOR J=1 TO N

90 $L_1 = (L_2 + J - 1) - INT((L_2 + J - 1)/3) * 3$
100 $B(I, J) = L_1$
110 NEXT J
120 NEXT I
130 GOTO 16
140 FOR I = 1 TO N
150 FOR J = 1 TO N
160 $B(I, J) = INT(2,99999999 * RND(1))$
170 NEXT J
180 NEXT I
181 GOTO 16
190 $K_0 = \emptyset$
200 FOR J = 1 TO N
210 $I = 1 \text{ TO } N$
215 $S(1) = \emptyset$
216 $S(2) = \emptyset$
217 $S(3) = \emptyset$
220 FOR K = 1 TO L
230 $I_1 = I + C(K, 2)$

240 $J_1 = J + C(K, \ell)$
250 245 $M = 1$
250 IF $I_1 < 1$ OR $J_1 < 1$ OR $I_1 > N$
OR $J_1 > N$ THEN 270
260 $M = A(11, J_1) + 1$
270 $S(M) = S(M) + 1$
280 NEXT K
290 IF INT(S(1)/2) * 2 = S(1)
THEN R = 0
295 GOTO 330
300 R = 2
310 IF $S(2) > S(3)$ THEN R = 1
320 IF RL = 0 THEN GOTO 330
~~330~~ - B325 IF RL = 1 AND
 $A(I, J) > 0$ THEN $R = R(I, J)$
326 GOTO 330
327 IF RL = 2 AND $A(I, J) = 2$
THEN R = 2

330 $B(I, J) = R$

340 IF $B(I, J) = R(I, J)$ THEN 350

K0 = K0 + 1

350 NEXT I

360 NEXT J

370 IF $K0 = N * N$ THEN 470

380 375 PRINT AT (2, 11); "LIF STEP"; LK

377 $LK = LK + 1$

380 FOR I = 1 TO N

390 FOR J = 1 TO N

400 $X1 = SI * (I - 1) + XS$

410 $Y1 = SI * (J - 1) + YS$

420 LINE (X1, Y1) - (X1 + SI, Y1 + SI), B(I, J), BF

430 $R(I, J) = B(I, J)$

440 NEXT J

450 NEXT I

452 IF $LK < L1$ THEN 460

453 PRINT AT(12,1); "CONTINUE"
454 ~~INPUT A\$OC~~
455 IF A\$OC = "Y" THEN 456
ELSE 480
456 PRINT AT(12,1); "
457 PRINT AT(13,1); "
458 L1 = LK + 50
460 FOR I = 1 TO 1000
465 NEXT I
467 GOTO 490
480 ~~INPUT~~ PRINT "NEXT LIFE?"; A\$OC
490 IF A\$OC = "Y" THEN 2
500 STOP

ПРОГРАММА СОЗДАНИЕ (CREATE)

1 CLS

3 DIM FL(100)

10 ~~INPUT~~ INPUT "HMAX": HM

20 INPUT "HD": HD

30 INPUT "AI0": AI

35 AL = AL * 3,141596 / 180

40 HD = HD / 15

50 H1 = HD

60 INPUT "MU": MU

70 INPUT "P0": P0

80 ~~INPUT~~ INPUT "K1": K1

85 FOR I = 1 TO 1000

90 FL(I) = 2

95 NEXT I

100 INPUT "P10, P20": K1, K2

110 XT = 160

111 MS = 200 / MM

112 YT = 0

120 P = 0

121 F1 = 1

130 CLS

131 KL = 1

132 AI = AL

133 H = NO

134 HT = Ø

140 A = H / SIN (AI)

150 AH = A * COS (AI)

160 HB = 2 * AH * KL

170 P = 2 * KL * (A + HH) * PO + P

180 K = INT((1 + 2 * MU) / (2 * MU))

190 XØ = XT - HD / 2 * MS

195 II = 1

200 FOR I = 1 TO 2 * KL

205 X1 = XM * MS

210 X = XØ + X1 * (I - 1)

220 I = II * (-1)

230 Y1 = YT + (1 - II) * H / 2 * MS

240 Y2 = YT + (1 - (-1) * II) * H / 2 * MS

250 LINE(~~X~~, Y1)-(X+X1; Y2), 2

260 NEXT I

270 $Y_T = Y_T + H * M_S$

275 $H_T = H_T + H$

280 LINE - (X0, YT), 2

293 $H = H \emptyset$

294 IF ~~HT + HD > HM~~ THEN

460

295 $KK = F \underline{L} (F \emptyset)$

300 FOR I=KK TO K

310 $R_E = I$

311 $L = HB / (2 * (I - 1))$

315 $AL = ATN(H/L)$

316 IF $L > H * 2,5$ THEN N375

320 $R = R_1 * P * SIN(AL) / (H * (KOL *$

100 INPUT P)

110 $XT = 160 - (I - 1) * 33 = IT$

120 $YT = 160 - (I - 1) * 33 + TY = SY$

130 $Y_T = (T1 * (I - 1) - 1) + TY = SY$

140 $P = 0$

151 $E1 = 1$

10 INPUT N

11 DIM A(N)

12 FOR I=1 TO N

13 INPUT A(I)

14 NEXT I

15 K=A(1)

16 FOR I=2 TO N

17 IF A(I) < A(I-1) THEN 100

~~18 ELSE~~

19 K=A(I-1)

20 A(I-1)=A(I)

21 A(I)=K

22 NEXT I

23 PRINT A

UNITAZ

10 CLS
20 DATA &0177700, &017, &0177760,
&077, &0174, &01770, &034, &01000
30 DATA &0 1770, &03000, &0177770,
&0 1777, &0177060, &0777, &040, &077,
&0 10014
40 DATA &0100, &045123, &0140200, &0
45524, &0 100200, &050721, &0140200,
&0601.
50 DATA &0 100200, &050121, &0 140,
~~&0 200~~, &04101, &0 100200, &06121, &0 17776
60 DATA &045700
70 RESTORE
80 FOR D = &0 50000 TO &051477 ST
&0100
90 FOR SM = &030 TO &0 32 ST &02
100 READ X
110 POKE D+SM, X
120 NEXT
130 NEXT
140 FOR A = 793 TO 555 ST -.09

150 PORE &0177664, A
160 NEXT
170 FOR A=A.107 9 2 ST .09
180 PORE &0177664, A
190 NEXT
200 GOTO 140

127XX

XX - номер регистра

CIRCLE (X, Y), R, N, d_o, d_e, K_{xy}

X, Y - координаты центра окружности

R - радиус

N - ивем

d_o и d_e - начальная и конечная толщина дуги определяются углами d_o и d_e

K_{xy} - если K_{xy} < 1 то R соответствует заданному радиусу по оси X
если K_{xy} > 1 то по оси Y

BIN -преобразует целое число из диапазона $[0, 255]$ в восемьразрядный двоичный код.

Функции

$ABS(x)$ модуль аргумента, $|x|$

$ATN(x)$ арктангенс, $\arctg x$

$COS(x)$ косинус, $\cos x$

$EXP(x)$ экспонента, e^x

$INT(x)$ ближайшее целое, не превосходящее x

$LOG(x)$ натуральный логарифм, $\ln x$

$SGN(x)$ знак аргумента

$$SGN(x) = \begin{cases} -1, & \text{если } x < 0 \\ 0, & \text{если } x = 0 \\ 1, & \text{если } x > 0 \end{cases}$$

$SIN(x)$ синус, $\sin x$

$SQR(x)$ квадратный корень, \sqrt{x}

$RND(x)$ генератор случайных чисел из $[0, 1]$

$TAN(x)$ тангенс, $\operatorname{tg} x$

$\text{INT}(B * \text{RND}(1))$

B - предельное число

CLS - очистка экрана

POKE X,Y - запись в ячейку памяти

X числа Y

PEEK(X) - считывание из ячейки

памяти X

Запись Ввод с машиноподходом программы.

~~SAVE~~ запись

SAVE "название" или CSAVE "название"
считывание

LOAD "название" или CLOAD "название"
запись, считывание с дисковы.

запись

SAVE "TT:название" или CSAVE "TT:название"
считывание

LOAD "TT:название" или CLOAD "TT:название"

Считывание программы в машинных кодах с магнитодисковой кассеты

наберите на клавиатуре

MONIT ↵

при появления вопросительного знака набрать букву M и (VK)

?M

?ИМЯ

Наберите название игры и клавишу ввода
пример

?M

?ИМЯ STAR26

?

При введении программы внизу появится
знак вопроса. Если введена программа
с ошибкой то появится сообщение
но все же попробуйте запустить программу

?M ^{ПРИМЕР}

?ИМЯ КЛАД

ОШИБКА ВВОДА

?

Программа запускается при помощи
команды S1000

?S1000

PAINT(X, Y), N, N1

X, Y - точки внутри закрашиваемой
области.

N - цвет закрашивания

N1 - цвет границы закрашивания

X = POINT(X, Y)

X - цвет точки с координатами X, Y

POINT - ставование цвета точки
с заданными координатами.

DEF USRN = (80B)

A = USRN (80B)

запуск подпрограммы в машинных
кодах с адресом B

N - номер [0, 9]

20 INPUT N
30 DIM A(N)
40 FOR I=1 TO N
50 INPUT A(I)
60 NEXT I
70 NI=A(1)
80 FOR I=2 TO N
90 IF A(I) < A(I-1) THEN 100
100 MIN=A(I)
110 K=I
120 NEXT I
130 PRINT "MIN="; MIN, "K="; K
140 END
125 RETURN

80 IF A(I) > A(I+1) THEN 60
130 FOR I =1 TO N-1
140 GOSUB 65

и знову вони дійшли до кінця
циклическої та називалися циклом
но все як написано занесли в таблицю
значення

? K

? ЧИА КДР

Однака вона

?

Программа запускается при помощи
команды S0000

? S0000

THEOREM УПОРЯДОЧЕНИЕ СПУСКА ФАРАУЛИЙ

20 DIM AXOC(10)
30 ~~INPUT~~ PRINT "ВВЕДИТЕ 10 ФАРУЛИЙ"
40 FOR K = 1 TO 10
50 INPUT AXOC(K)
60 NEXT
70 LET M = 0
80 FOR K = 1 TO 9
~~90 IF AXOC(K) <= AXOC(K+1) THEN 150~~
100 BXOC = AXOC(K)
110 AXOC(K) = AXOC(K+1)
120 AXOC(K+1) = BXOC
140 M = 1
150 NEXT K
160 IF M = 1 THEN 70
170 FOR K = 1 TO 10
180 PRINT AXOC(K)
190 NEXT K

Сортировка всплеском

ПЛЕБ

250 NEXT

PAKETA

10 CLS
 20 LINE(100, 100)-(80, 150), 2
 30 LINE-(70, 160), 2
 40 LINE -(70, 170), 2
 50 LINE -(80, 165), 2
 60 LINE -(120, 165), 2
 70 LINE -(130, 170), 2
 80 LINE -(130, 160), 2
 90 LINE -(150, 150), 2
 100 LINE -(100, 100), 2
 110 PAINT(102, 110), 2, 2
 120 FOR A=100 TO 200
 130 LINE(100, A)-(80, A+50), 2
 140 LINE -(70, A+60), 2
 150 LINE -(70, A+70), 2
 160 LINE -(130, A+60), 2
 170 LINE -(150, A+50), 2
 180 LINE -(100, A), 2
 190 LINE (70, A+70)-(80, A+65), 0
 200 LINE -(120, A+65), 0
 210 LINE -(130, A+70), 0
 220 NEXT



БПЕР

Структурная схема ЭВМ

- 1) схема
- 2) характеристика блоков

Блокиющие логические узлы.
Типовые логические узлы и
их характеристика.

Данные проустановлены
блоках для ЭВМ

Алгоритмы и их схема
Основные этапы решения
задачи на ЭВМ

Блоки про. об., ОС.

Структура ОС ВВК и основные
характер. её программ

Проблемы

Принципы изобретательства

4 Задачи

Решающиеся

- 1) Чему является главной целью изобретательства
- 2) Чему являются дополнительными целями
- 3) Ограничения.
- 4) Накопление эмоционального заряда.
- 5) Ограничить от превозможных идей

Для того чтобы вдохновить
себя на изобретение ново-
запомад четырех способов

- 1) Насыще нас собой -
это пустые сроки реше-
ний.
- 2) Высыпивание - когда
решение пока не ясно
изобретатель набрасыва-
ет идеи. Внесем туда
заштаки, поправки пока
не забрызгаем свет. Так
как паническ от тесноты
мужайного - под эту поиск.
- 3) Поиск оригинальных
решений для трудовых
а также для загород (селе-
ческий человек)

4) Систематика - составляемся таблица возможных решений, которые подогревают обдумыванием.

Для того чтобы начать изменять вдохновение надо.

- 1) Научиться подавлять в себе критическое начало.
- 2) Накопить творческую энергию.
- 3) Разделить и выставить.

ПРОГРАММА "НЕБО"- ПЕРЕВОД С ФОРКАЛА
(возможны ошибки)

10 CHR\$\$(12,148,158,155,154,145)
20 CHR\$\$(155)
30 PRINT "ПРОГРАММА НЕБО"
40 CHR\$\$(146)
50 PRINT "НЕКОТОРЫЕ ВОЗМОЖНОСТИ
МИКРО-ЭВМ"
60 CHR\$\$(145)
70 PRINT "ЭЛЕКТРОНИКА БК-0010"
80 CHR\$\$(147)
90 PRINT "УВЕТНАЯ ГРАФИКА"
100 FOR I=1 TO 400
105 BEEP
110 NEXT
120 CHR\$\$(12)
130 X=128
140 Y=127
150 L=2
160 FOR I=1 TO 5
155 DIM R(5)
170 R(I)=RND(1)
180 S=R(1)
190 IF S+.05<0 THEN 240

200 IF $S < 0$ THEN 260
210 IF $S - .5 < 0$ THEN 280
220 CHR20C(145)
230 GOTO 290
240 CHR20C(146)
250 GOTO 290
260 CHR20C(147)
270 GOTO 290
280 CHR20C(148)
290 PSET (X * R(2) / (R(4) + L) + X,
 $Y * R(3) / (R(5) + L) + Y), 1$
300 GOTO 160

ПРОГРАММА "БРЕД"

```
10 FOR A% = 0 TO 250
20 FOR B% = 0 TO 250
30 X=POINT(A%,B%)
40 IF X<>0 THEN X=INT(3*RND(1))+1
50 PSET(A%,B%),X
60 NEXT
70 NEXT
80 GOTO 10
```

МАШИННЫЕ КОДЫ

ЦП - центральный процессор.

РОМ - регистр общего назначения

КОП - код операции

~~RR~~ - R - РОН

СК - стартовый команды (R7)

УС - указатель стека (R6)

РСП - регистр состояния процессора

ССП - слово состояния процессора

SS - поле адресации операнда
использовано

SRC - источник

(SRC) - оператор источника

DD - поле адресации операнда
приемника

DST - приемник

(DST) - операнд приемника

XXX - алиасение (8 разрядов)

NN - алиасение (6 разрядов)

() - содержимое ячейки

^ - логическое умножение ("и")

v - логическое сложение ("или")

✓ - ~~не~~ "исключющее или"

¬ - ~~не~~A (отрицание)

<-- - становится равным

- запись в стек

- выборка из стека

В - байтовая программа

ЯП - ядреиъ по линии

ВСС - восьмеричная система счисления

OPR - под операции

ОЦИ - общая шина

Обращение к стеку

К стеку можно обращаться
трех способами:

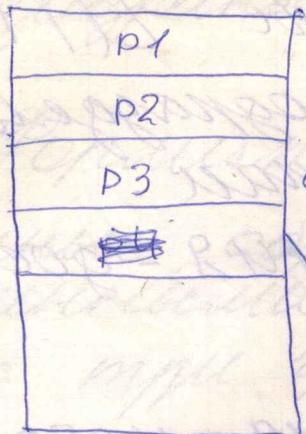
- 1) ПРОГРАММО : с помощью команды
пересылок ;
- 2) Используя команды работы
с подпрограммами ISR и RTS
- 3) Автоматически , при
прерывании программ

При выполнении команды ISR
в стек загружается содержимое
регистра, называемого "указате-
лем связи, а при выполнении
команды RTS содержимое указыва-
емого регистра восстанавливается
из стека.

При прерывании программы
стек используется для автомо-
тического запоминания
текущего содержимого складресса
возбраня (SS17).

Стек индекс убывает
по мере занесения в него
элементов и фиксируется по
мере их извлечения:

верхние
адреса



00 1776

00 1774

00 1772 --yc



P4

Слово состоящие процессора

8 7 6 5 4 3 2 1 0
1/0 T N Z V C

ПРИОРИТЕТ

ПРЕРЫВАНИЕ ПО Т разряду

ОТРИЦАТЕЛЬНЫЙ РЕЗУЛЬТАТ

АРИФМЕТИЧЕСКОЕ ПЕРЕПОЛ.

нулевой результат

ПЕРЕНОС

Регистры L1

Регистры L1 подразделяются на две категории.

ПЕРВАЯ КАТЕГОРИЯ - доступны из б/м.

ВТОРАЯ КАТЕГОРИЯ - используються только процессором.

Системы машинных
каранов.

классифицируются.

Машинные караны делятся на две группы:

- 1) Однодаресное, двухдаресное, передачи управления и безадресное.

2) безавральное (множественное).

Решение задачи

Данные, обрабатываемые машиной для кодирования по заданию на три группы:

исходные коды, число единиц равнености и ненулевых запятых.

1001	01 000 010	01000001	1000
15	B	87	A

Числа ~~за~~ без знака имеют диапазон:

000 - 377 - для байта, 00 000 00 - 1 777 77 - для слова

Битовое величина задает значение отдельных разрядов байта исходного слова.

число с фиксированной запятой
и концом записи точкой или запятой (также 15 разрядов соотв.
являются знаками). Формат
числа с фиксированной запятой:

2 6 0

число

15 14 0

число

^

--- знак
числа
байт

^

--- знак
числа
слово

Значение знака разряда - 0 для
положительных чисел, 1 для
отрицательных. Отрицатель-
ные числа представляются
в дополнительном коде,
который образуется инвер-
сией дополнительных разрядов
10 - 6 для байта, 0-14 - для слова

а добавлены в исходный разряду. Например:

число	Вашеричное-код запоминания	восьмерич- ное под слово
+ 5	005	0000005
- 5	373	177773
0	000	0000000

Диапазон представленных чисел с фиксированной запятой

- 128 90 + 127 10 - для байта,

- 32768 90 + 32767 10 - для слова

Числа ставящиеся запятой заканчиваются дво слово (32 разряда) и представляются в формате.

3 1(15)	23(7) 22(6)	16(0)	15	0
порядок		МАНТИССА		
ЗНАК МАНТИССЫ				

Знач машина определяет значение числа.

0 - для положительных чисел

1 - для отрицательных.

Машина ~~запоминает~~ отрицательных чисел представляемые в дополн. коде.

Описание системы исчислений

В исках однодресных и двухдресных исчислений содержится информация, которая определяет:

- выполняемую функцию (код операции)
- РОН, используемые при вычислении операндов.
- метод адресации (способ использования выделенного РОН)

Роман может быть исполь-
зовано

1) как пакетами

2) Как указатели адреса

3) Как указатели адреса, содержащие некоторое изме-
няемое автоматически с задан-
ной шагом

4) Как адресное поле

формат команд

1) Однодрессные команды:

CLR	DEC	SBC	ROL	JMP	MPS
COM	NEG	TST	ASR	SWAB	SXT
INC	ADC	ROR	ASL	MFPS	

РОД ОПЕРАЦИИ

DDWW SS

2) Двухадресные команды

BIT BIS SUB CMP
 BIC ADD MOV

КОД ОПЕРАЦИИ	DD или SS	DD или SS
15	12 11	6 5

Команда XOR:

КОД ОПЕРАЦИИ	R	DD
15	9 8	6 5

3) Команды управления программы.

а) Команды ветвления:

BR BEQ BLT BLE BMI BIOS
 BNE BGE BST BPL BHI

КОД ОПЕРАЦИИ	X X X
15	8 7 6 5

Б) Команда ISR

код операции	R 3 2 3 4 0	D D
15	9 8	6 5

Б) Команда RTS

код операции	R
15	3 2 3 2 3 2 3 2

Г) Команды прерывания

IOT, EMT, TRAP, BPT

код операции	?
15	1 3 7 4

Д) Команда MARK

код операции	NN
15	6 5

Е) Команда SOB

код операции	R	NN
15	9 8	6 5

4) Спец. команды

HALT RESET RTT
WAIT RIT NOP

код операции

45

5) Команды изменения привилегий

CLC CLD CCCS SEC SEZ
CLV CLN GEN SEV SEC

код операции

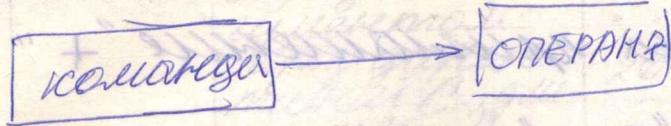
N	Z	V	C
---	---	---	---

4 3 2 1 0

15

МЕТОДЫ ПРЯМОЙ АДРЕССАЦИИ
В структуре прямой адресации в res. общем назначении содержатся мод. операнду и мод. адреса операнда

1) перед \emptyset , регистровый: OPR
R (к содержимому операнда)



ПРИМЕР $\$INC R3$ 005203
увеличим на 1 содержимое
регистра R3

1002	12700
1004	40000
1006	12701
1010	3
1012	12702
1014	1022
1016	12703
1020	77777
1022	20300
1024	103
1026	12004
1030	60104
1032	10400
1034	102
1036	

(167)

Таблица действий отладки.

"ЛАТ" РТ "ВВОД"

и в ответ на приглашение "+" -

"РУС" Т С

обозначение	название	описание	пример
A	установить или проконтролировать значение первого адреса	в микро-ЭВМ за- нимая или изы- чируясь значе- ние текущего адреса	$1000AA = 1000$
D	установить/проконтролировать значение длинны массива (в байтах)	устанавливается границы рабочего массива от адреса A до адреса (A+D)	$100A..R = 100$
P	разложить число в одинаковые адреса	числовой арифметикой, стационарный перед действием Р, за- писывается по кодам из адресов - (A+D). Если арифметик отсут- ствует, дробь- тива не воспри- нимается.	$1000A$ $100R$ $7P$ в консоль желтую строку от 1000 до 1100 записывается 7.
C	сравнить два массива: эм- блемный и	эталонный мас- сив определен в границах: A(A+2)	$1000A$ $20A$ $1P$

контро-
лируемый

Проверяющий
массив имеет
различия: нача-
льный адрес
определен для
измененного
перевод диапазо-
на С, диапа-
зона равен 1.
если массивы
отличны друг
от друга, на
экране: адреса
и содержимое
памяти в
диапазоне
 $A - (A + R_1)$, ар-
гументом - (ару-
мент + R_1)

адр: знак адреса

адр: знак адреса: знак

Если массивов
сравнение не
является специ-
альным "xx"

2040C

2000:1

2040:**

2002:1

2042:**

...

2016:1

2056:**

2040A

1P

2040C

XX

-

** - значение со-

держащееся в
текущем поле

X

подчи-
тать кон-
тролирую-
щий мас-
сив

Подсчитывает
контролируемую
сумму массива
граничью ко-
торого зада-
ны адресами
 $A, A + R_1$

129000 A

20000 R

$x = 177777$

II

пересчет
(сумма
исчислена с
массива)

Производится
пересылка эла-
гированного мас-
сива граничек

2000A

20A

3000 P

Массив, определенный
в диапазоне 2000 -

которого определяются адреса А и В.
Массив инициализируется:

2000, сконструирован по адресу 3000

Начиная конец определяется арифметикой А, т.е. граница совпадает с длиной эталона: равна
В

№ 7 листать (раскручивать массив на экране)

На экран выводится массив данных, начиная с адреса А и длиной, равной значению адреса А и длиной, что арифметично

Д

2000 А 07
Раскручиваем содержимое 4-х ячеек памяти: 2000, 2002, 2004, 2006

И Записать / прочитать содержимое ячеек памяти

Осуществляется запись / чтение ячеек, определенной длиной адресом А. Записывается слово.

2000 А
Изменение
1000 А

Изначально

Б Записать / прочитать содержимое байта

Аналогично предыдущей директиве

4	Цикличес- кое изменение/ запись об- бо	Происходит с цикличес- ким изменением / запись ячейки информации для иска с адресом А. Выход из цикла по условию "стоп"	20000 А 123Ц - запись 20000 Ц 123 123... - изменение
III	Снять за- щиту сис- темной области.	Снимает защиту системной области ОЗУ. Оператор должен следить за пра- вильностью записей в этой области адресов, чтобы не испорчить сис- темной инфор- мации. Установка "стоп"	
,	Изменение/запи- сью с ини- циатором	Происходит измене- ние / запись с ини- циатором, т.е. перемещением содер- жимого ячейки А+2 и запись по происходит по текущему адресу.	10000 А 10, знач- ие записи вы- дается по ад- ресу 10000. Знач - содержи- мое ячейки 1002
:	изменение/запи- сью с ини- циатором	аналогично преды- дущей директиве	

запись за - пись словас декрементом	Происходит за- пись / запись с декрементом т.е. неотменя- ем содержи- мое адресу А-2 и запись происходит не между собой адресу.	1002A20-З ^{нок} 10 записывается по адресу 1000. $A = 1000$
---	--	---

запись / за- пись с декре- ментом(бий- то)	аналогично предыдущей директиве		
МП MC	пуск мотора манипулято- рного стола по шагам мони- торинга	Служат для обеспечения подготовки ма- нипуляторной системы	
М3 М4	записать ин- формацию по МР считать ин- формацию с МР	Директивы про- водят в виде текстов ре- цепции и позво- ляют просмат- ривать записи информации по на манипулятору. Информация — массив памя- ти определя- емых в виде файла с ини- циацией. Число эти- х не дол-	Формализуйте максимо: 1000A20 Р, 1Р МП М3 Напишите т.е. манипулятор "ПУСК", "ЗАП" АДРЕС = 1000 ДЛИНА = 20 ЧИСЛО = ****

число превосходит
16 символов. Адрес
начала, длина, или
вводится в ходе
диалога. Диагности-
ки позволяют фор-
мировать эти динамич-
ные массивы дан-
ных, а также тек-
сты программы.

МП
МЧ
Нажмите
КЛ. МАГН.
"ПУСК"
АДРЕС: ****
ИМЯ: ****
ЗАТРУЖЕН
Файл

МФ
выполните
операцию "при-
остановка про-
цессора"

Осуществляется поиск
файлов, или параметрою
указано в диалоге.
Найденный файл
не записывается в ОЗУ
останов процессора
также происходит в
результате ошибки.

МП
МФ
НАЖМИТЕ
КЛ. МАГН.
"ПУСК"
ИМЯ: ****
останов
после фи-
ла ****

С
Пуск на
программу
пользователя

Адресом, диапазоном
определяет адрес
передачи управления.
Программу ~~пользователя~~
~~пользователя~~ можно ввести
в ОЗУ с помощью
диагностической
возврат в МСД осу-
ществляется клавишой
"СТОП" или передачей
управления по адресу

введите
программу
012700
353
104016
1
Agr. G
K
-

160100

задачи

удобства по-
иска или введен-
ной символ

K

вход из
МСДуправление пере-
ходами экрана

TK

Директивы
переходов в
различные
МСДиспользоваться
для переходов
между моду-
лями МСД

TD

4-1000

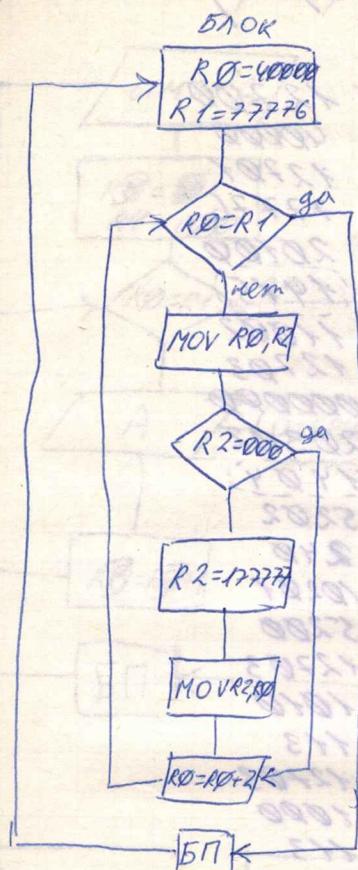
" " "

94

2

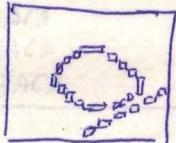
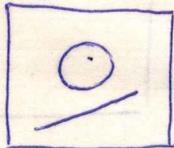
**

прац. I (увеличене розміру)



до виконн.

после



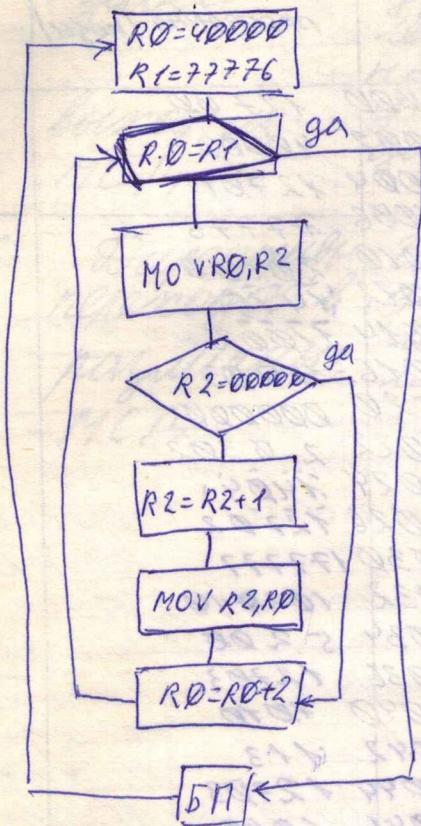
ПРОГРАММА (код)

1000	127 00
1002	400000
1004	12701
1006	77776
1010	20100
1012	74151
1014	71002
1016	12703
1020	0000000
1022	2 0302
1024	74041
1026	72702
1030	177777
1032	102010
1034	5 2 00
1036	12703
1040	1010
1042	113
1044	12703
1046	10,00
1050	113
1052	
1054	

R0
NLDIFP
WAKCNU
B3A8

... - возможны ошибки

программа II (чертежи №61)



1000	12700
1002	40000
1004	12701
1006	77776
1010	20100
1012	(1415)
1014	71002
1016	12703
1020	00000000
1022	20302
1024	(7404)
1026	5202
1030	240
1032	10201
1034	5200
1036	12703
1040	1010
1042	113
1044	12703
1046	1000
1050	113

до выполн.

КОЛЯ
ИГОРЬ
МАКСИМ
ВАСЯ

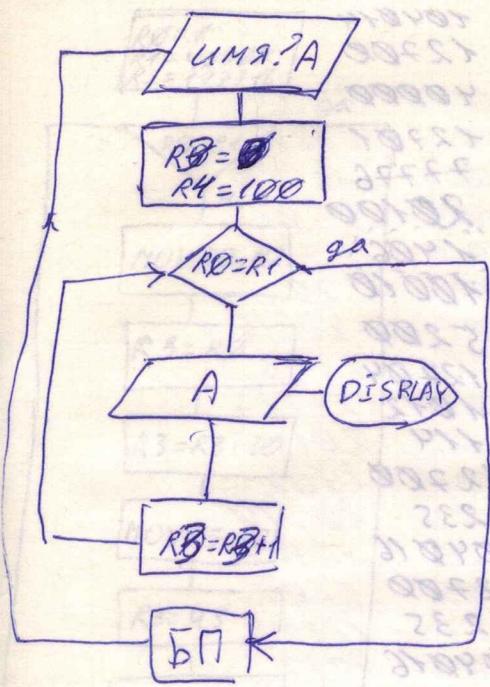
после

ИГОРЬ
МАКСИМ
ВАСЯ

запись



прогр III (услуг)



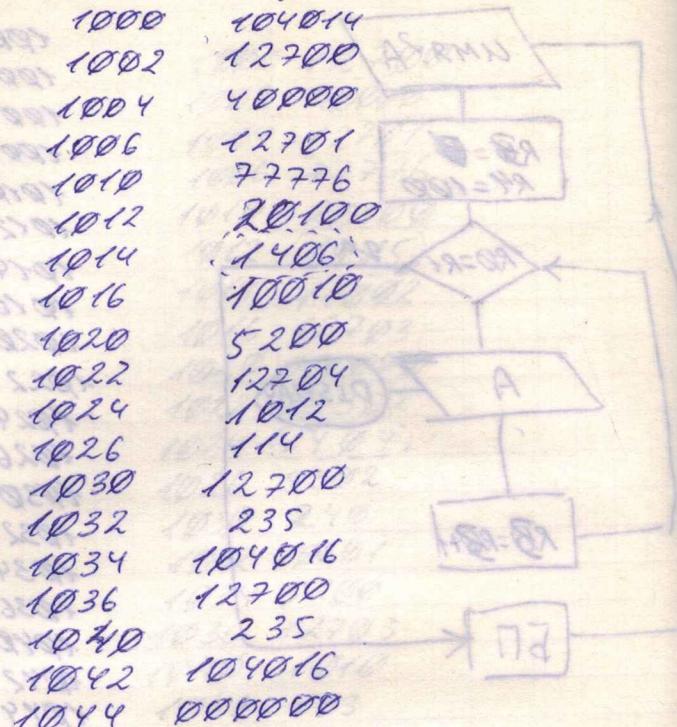
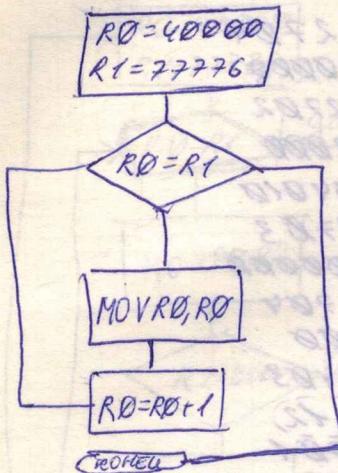
? ВАСЯ

ВАСЯ
ВАСЯ
ВАСЯ
ВАСЯ
ВАСЯ

1000	12701
1002	20000
1004	12702
1006	12000
1010	104010
1012	12703
1014	0000000
1016	12704
1020	100
1022	20403
1024	(1412)
1026	12701
1030	2000
1032	12702
1034	12000
1036	104020
1040	12704
1042	1016
1044	114
1046	5203
1050	12704
1052	1000
1054	114

(2000 строка символов)

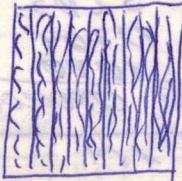
программа IV (исходная)



90



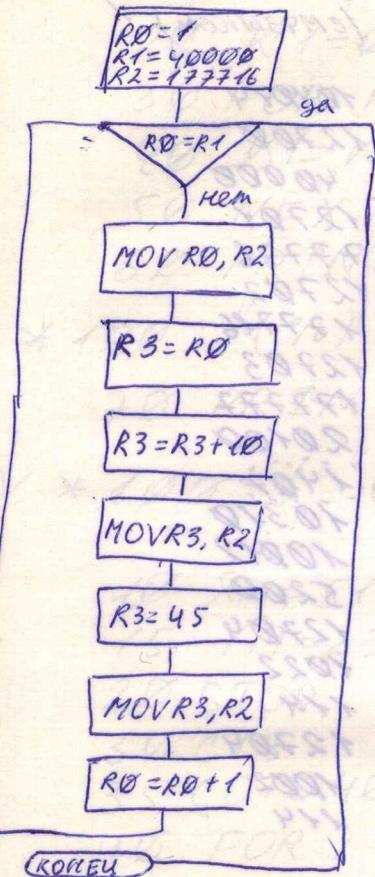
NOCL



MAKUM
BALA



ПРОГРАММА V (МУЗЫКА)



1000	12700	=09
1002	1	=99
1004	12701	=59
1006	400000	=29
1010	12702	=09
1012	177716	
1014	20100	
1016	1416	
1020	10012	
1022	10003	
1024	12704	
1026	10	
1030	60403	
1032	10312	
1034	12703	
1036	45	
1040	10312	
1042	5200	
1044	12703	
1046	1014	
1050	113	

6 R1 - гимнсовместные муз

80 RETURN

90 NEXT

100 R = 19 + INT(40 * (RND(1)+1)/2)

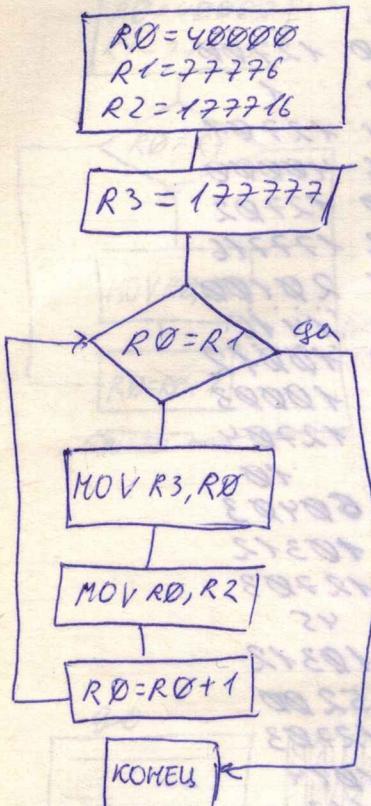
110 X = INT(XM * (RND(1)+1)/2)

120 Y = INT(YM * (RND(1)+1)/2)

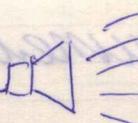
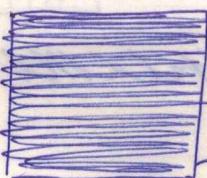
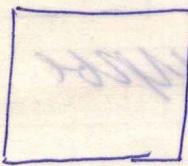
130 DT = 2 * 3.14 / R

140 TH = 0

ПРОГРАММА (VI) (ЗАКРЫТИЕ) (СМЫЗДИКОЙ)



1000	104014
1002	12700
1004	40 000
1006	12701
1010	77776
1012	12702
1014	177716
1016	12703
1020	177777
1022	20100
1024	1407
1026	10310
1030	10012
1032	5200
1034	12704
1036	1022
1040	114
1042	12704
1044	1002
1046	114



100 - 19 8

10 INPUT "ЧИСЛО ИСПЫТАНИЙ"; N
20 K1 = 0
30 FOR K = 1 TO N
40 K1 = K1 + INT(RND(1) * 2)
50 NEXT
60 PRINT "ДОЛЯ НУЛЕЙ РАВНА"; (N - K1) / N
* 100; "(%)"
70 PRINT "ДОЛЯ ЕДЕНЦЕЙ РАВНА"; K1 /
N * 100; "(%)"
10 XM = 511
15 YM = 239
20 STOP
30 CLS
35 GOTO 40
40 FOR J = 1 TO 10
50 GOSUB
60 NEXT
70 FOR K = 0 TO 100
80 RETURN
90 NEXT
100 R = 10 + INT(40 * (RND(1) + 1) / 2)
110 X = INT(XM * (RND(1) + 1) / 2)
120 Y = INT(YM * (RND(1) + 1) / 2)
130 DT = 2 * 3.14 / R
140 TH = 0

150 $X_1 = X + \text{INT}(R * \cos(\theta))$

160 $Y_1 = Y + \text{INT}(R * \sin(\theta))$

170 IF $(X_1 - XM) \leq 0$ THEN 190

180 $X_1 = XM$

190 IF $(-X_1) \leq 0$ THEN 210

200 $X_1 = 0$

210 IF $(Y_1 - YM) \leq 0$ THEN 230

220 $Y_1 = YM$

230 IF $(-Y_1) \leq 0$ THEN 250

240 $Y_1 = 0$

250 PSET(X_1, Y_1), 1

260 FOR K = 0 TO R

270 GOSUB

280 RETURN

290 $\theta = \theta + DT$

300 $X_2 = X + \text{INT}(R * \cos(\theta))$

310 $Y_2 = Y + \text{INT}(R * \sin(\theta))$

320 IF $(-X_2) < 0$ THEN

330 $X_2 = 0 + ((X - XM) * DX) + TN1 + DT = X$

340 GOTO 310 $((Y - YM) * MX) + TN1 = Y$

350

$DX = 1.8 * S = T$

$S = HT$

символ ("но фразе")

```
10 PRINT "(любое слово)"
20 DATA 20241, 8012700, &040000, &012701, &077776,
&020100, &01410, &011002, &06002, &010210, &05200, &0
12703, &01012, &0113, &012703, &01002, &0113
30 FOR I% = &01000 TO &01040 ST &02
40 READ X
50 POKR I%, X
60 NEXT
70 DEFUSR1 = (&01000)
80 I = USR1(&01000)
```

"СОВЕТСКИЕ СОЛДАТЫ, ХРАБРО СРАЖАЛИСЬ С
ФАШИСТСКИМИ АКУПАНТАМИ"

10 O=0

```
10 CLS
20 ? "СОВЕТСКИЕ СОЛДАТЫ, ХРАБРО СРАЖА
ЛИСЬ С ФАШИСТСКИМИ АКУПАНТАМИ"
30 INPUT "СУЩЕСТВИТЕЛЬНОЕ"; A$0
40 INPUT "ПРИЛАГАТЕЛЬНОЕ"; B$0
50 INPUT "ГЛАГОЛ"; C$0
60 INPUT "НА КАКОЮ БУКВУ В СЛОВЕ", D$0
70 INPUT "ПО ДО ПАДАЕТ УДАРЕНИЕ"; L$0
80 ДО ПАДАЕТ УДАРЕНИЕ"; L$0
90 IF A$0 = "СОЛДАТЫ" THEN O=O+5 ELSE O=O2
100 IF (B$0 = "СОВЕТСКИЕ") OR (B$0 = "ФАШИСТСКИЕ")
    THEN O=O+5 ELSE O=O+2
110 IF C$0 = "СРАЖАЛИСЬ" THEN O=O+5
ELSE O=O+2
```

130 IF $\Delta x = "A"$ THEN $O = O + 5$ ELSE $O = O + 2$
140 ?"ВАЛІДОВЕНКА"; ~~INT(O/4)~~
150 END

180 $X_1 = X_4$
190 $Y_1 = Y_4$
200 $X_2 = X_3$
210 IF ($X_1 < Y_1$) THEN $L = X_1$
220 $Y_1 = Y_4$
230 IF ($-Y_1 < L$) THEN $L = -Y_1$
240 $X_1 = Y_1$
250 $Y_1 = Y_4$
260 FOR $i = 0$ TO L DO $C = C + \text{BOXANGLE}$
270 $X_1 = X_4$
280 $Y_1 = Y_4$
290 $X_2 = X_3$
300 $Y_2 = Y_3$
310 $X_3 = X_4$
320 $Y_3 = Y_4$
330 $X_4 = X_1$
340 $Y_4 = Y_1$
350 $L = L + 1$
360 $C = C + \text{BOXANGLE}$
370 $O = O + C$
380 $O = O + 2$

390 IF $\Delta x = "BPAKAVNCP"$ THEN $C = 0$
400 $C = C + 2$

~~10 ATN 1 G=0~~
10 ~~CLS~~ ~~IF M=0 THEN~~ CLS
20 X=(80177662)
30 IF X=32 THEN M=~~M+1~~
40 ~~K=K+1~~ IF K=30 THEN K=3
50 IF M=1 THEN C=C+1
55 IF C=0 THEN PRINT AT(-K,2); "<~~█~~>"
60 PRINT AT(10,C); "I"
70 PRINT AT(10,C); "L"
80 IF (K=10) AND (C=2) THEN
COSUB 130
90 PRINT ~~IF~~ ? AT(K,2); "~~█~~"
100 ? AT(10,C); "~~█~~"
110 ~~GOTO~~ IF G>10 THEN
120 GOTO 20
130 FOR D=0 TO 10
140 BEEP
150 NEXT
160 G=G+1

~~x3~~
 10 FOR A = 8042000 TO 8042076 ST ~~802~~ 802
 20 FOR A1 = 800 TO 803300 ST ~~800~~ 800
 30 FOR A2 = 803500 TO A1 ST ~~80100~~ 80100
 40 A3 = A
 50 A3 = A3 + A2
 60 A4 = A3
 70 A4 = A4 - 80100
 80 X = PEEK(A4)
 90 POKE A3, X
 100 A3 = A3 + 80100
 110 X = PEEK(A4)
 120 POKE A3, X
 130 NEXT A2
 140 NEXT A1
 150 NEXT A
 160 END

APU ©

King

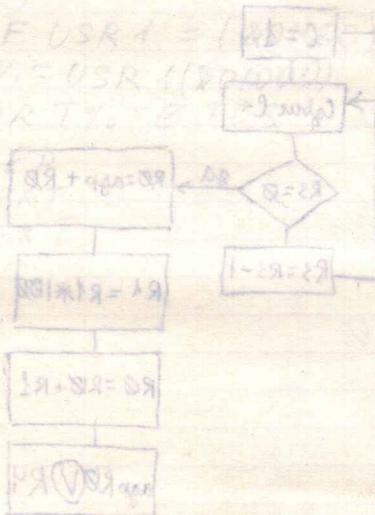
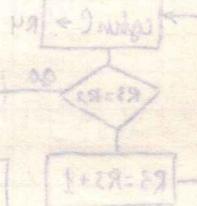
STAR

1000	12700	1050	1014
1002	42000	1052	22701
1004	12701	1054	3300
1006	0	1056	1404
1010	12702	1060	62701
1012	3500	1062	300
1014	10003	1064	137
1016	60203	1066	1010
1020	10304	1070	22700
1022	162704	1072	42076
1024	100	1074	1404
1026	11413	1076	62700
1030	62703	1100	2
1032	100	1102	137
1034	11413	1104	1004
1036	20102	1106	207
1040	1404		
1042	162702		
1044	100		
1046	137		

Расширение буфера на 1 строку в право

1000	12700
1002	40000
1004	10001
1006	12002
1010	6202
1012	6202
1014	50221
1016	20027
1020	77776
1022	1402
1024	137
1026	1004
1030	207

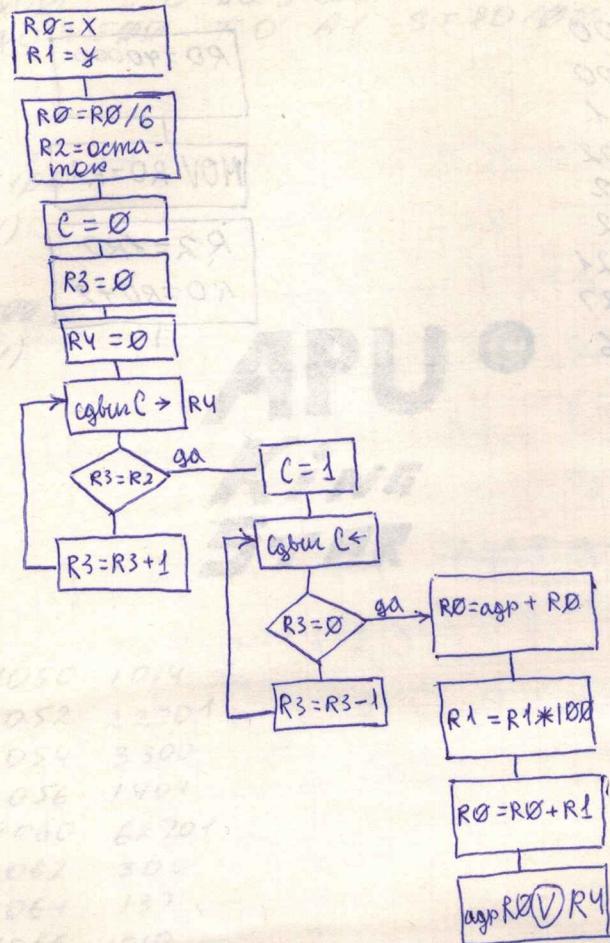
```
RD = 40000  
MOV RD = R1  
R2 = 1RD  
RD = RD + 2
```



CiR.

PSET (APU)

20 FOR A1
30 FOR A2
40 A2 = R1
50 A3 = A3 + A2
60 A1 = A1 - A3
70 A4 = A4 - A1
80 X = PEER (A1)
90 R0 = A3 / A1
100 A3 = A3 + 30
110 X = PEER (A1)
120 PDRE A3 X
130 NEXT A1
140 NEXT A1
150 NEXT A1
160 END

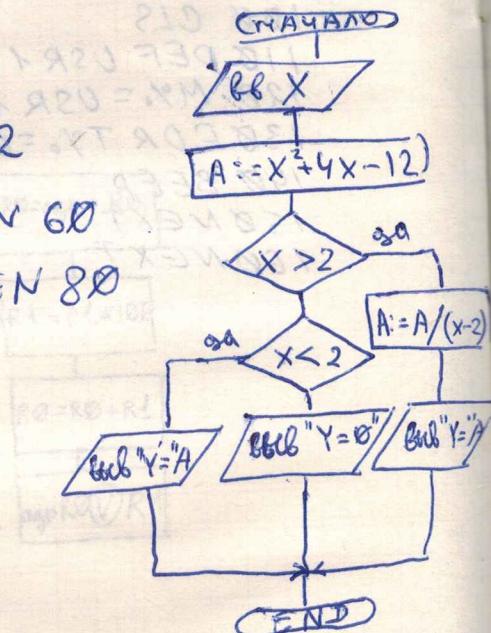


10 ? CHR205
 20 ? CHR20C
 30 FOR I% = &010001 TO &01023 ST&02
 40 READ X
 50 POKE I%, X
 60 NEXT
 70 DATA &012700, &040000, &020027, &077776,
 &034005, &010010, &062700, &00, &0127, &01004, &0207
 80 FOR I% = 1 TO 5
 90 POKE &01016, I%
 100 CLS
 110 DEF USR1 = (&01000)
 120 M% = USR1(&01000)
 130 FOR T% = 0 TO 5 * H + X*X = A
 140 BEEP
 150 NEXT
 160 NEXT
 170 INPUT "НІЧНІЙ СЕКТОР"
 180 INPUT "ВСЕХАЛІ ПРЕДІЛ"
 190 INPUT "ШАГ" N
 200 INPUT "Y=" Y
 210 FOR X = X1 TO X2 STEP N
 220 Z = ? Z = (SIN(X*X))*SIN(Y*Y)) / 2
 230 NEXT X

$$1) Y = \begin{cases} ax^2 + bx + c & x < 2 \\ x^2 + d & x > 2 \\ x & x = 2 \end{cases}$$

$$2) Y = \begin{cases} \frac{x^2 + 4x - 12}{x-2} & x > 2 \\ x^2 + 4x - 12 & x < 2 \\ 0 & x = 2 \end{cases}$$

2) ~~READ~~ INPUT X
 20 A = X * X + 4 * X - 12
 30 IF X > 2 THEN 60
 40 IF X < 2 THEN 80
 50 GO TO 90
 60 A = A / (X - 2)
 70 ? "Y = "; A : END
 80 ? "Y = "; A : END
 90 ? "Y = 0": END



1) 10 INPUT A, B, C, X

20 $D = B * B + -4 * A * C$

30 IF $X < 2$ THEN 60

40 IF $X > 2$ THEN 80

50 GOTO 100

60 $Y = A * X * X + B * X + C$

70 ? "Y = ", Y : END

80 $\underline{Y} = D + X * X$

90 ? "Y = ", Y : END

100 $Y = X * X * X * X$

110 ? "Y = ", Y

$$3) \quad z = (\sin x^2 \cdot \sin y^2)$$

$$4) \quad y = \sqrt{x^2 + z^2} \cdot l$$

3) 10 INPUT "НИЖНИЙ ПРЕДЕЛ": X_1

20 INPUT "ВЕРХНИЙ ПРЕДЕЛ": X^2

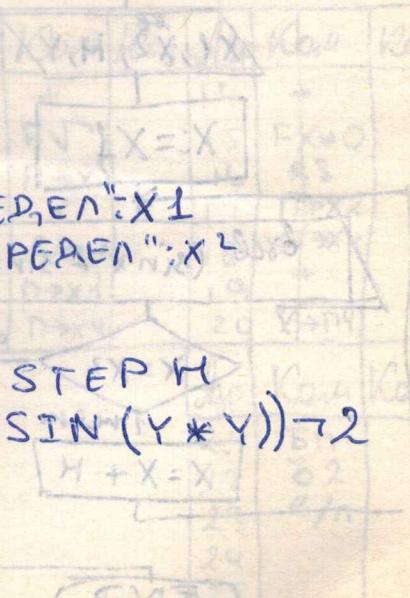
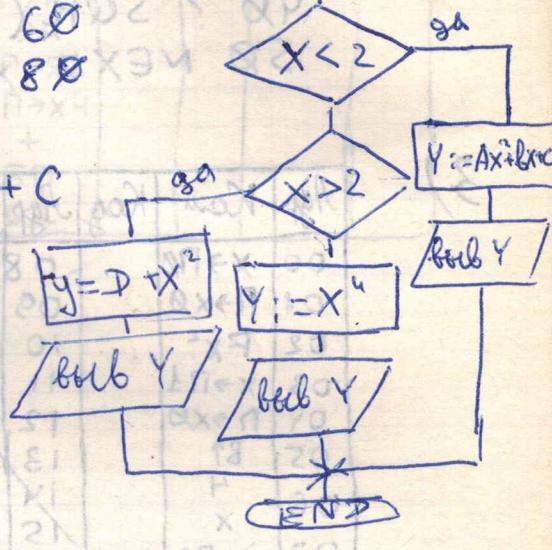
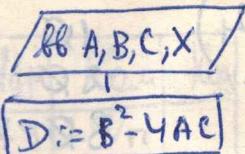
30 INPUT "ШАГ": H

40 INPUT "Y = ", Y

50 FOR $X = X_1$ TO X^2 STEP H

60 Z ? $\underline{Z} = (\sin(X * X) * \sin(Y * Y))^{-2}$

70 NEXT X



4)

 1) INPUT X_1, X_2, M

 2) INPUT Z, E

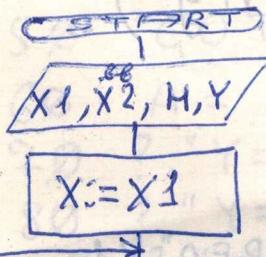
 3) FOR $X = X_1 + 0.1 \times X_2$ STEP M

 4) ? SQR($(X*X + Z*Z)*E$)

 5) NEXT X

3)

Alg	Kom	Kog	Alg	Kom	Kog	Stgp	Kom	Kog	
00	$X \rightarrow \text{P}0$		08	$R \rightarrow X_1$		17	2		26
01	$\text{P}0 \rightarrow X_0$		09	$B \uparrow$		18	-		27
02	$F \times^2$		10	$\text{P} \rightarrow X_2$		19	$F \times 0$		28
03	$X \rightarrow \text{P}1$		11	+		20	$F \times 0$		29
04	$\text{P}1 \rightarrow X_0$		12	1		21	$F \times 0$		30
05	$B \uparrow$		13	2		22	$\text{P} \rightarrow X_0$		31
06	4		14	+		23			32
07	X		15	$X \rightarrow \text{P}1$		24	2		33
	$X \rightarrow \text{P}2$		16	$\text{P} \rightarrow X_0$		25	-		34



$$\text{Calc } (\sin X^2 - \sin Y^2)^2$$

$$X = X^2$$

1 mem

$$X = X + H$$

END

Agr	Kom	Kog	Agr	Kom	Kog	Agr	Kom	Kog	Agr	Kom	Kog
00	$\pi \rightarrow X_0$		08	$F \sin$		16	-		24	e/n	
01	$X \rightarrow \pi_4$		09	B^+		17	$F \neq 0$				
02	$\pi \rightarrow X_4$		10	$\pi_7 \rightarrow X_5$		18	24				
03	$F x^2$		11	X		19	$\pi \rightarrow X_2$				
04	$F \sin$		12	$F x^2$		20	$\pi \rightarrow X_4$				
05	$X \rightarrow \pi_5$		13	e/n		21	+				
06	$\pi \rightarrow X_3$		14	$\pi \rightarrow X_1$		22	π				
07	$F x^2$		15	$\pi \rightarrow X_4$		23	02				

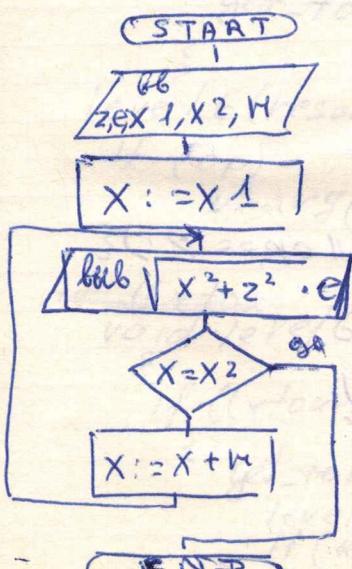
$X_1 \rightarrow R_0$

$X_2 \rightarrow R_1$

$\pi \rightarrow R_2$

$Y \rightarrow R_3$

$X \rightarrow R_4$



Agr	Kom	Kog	Agr	Kom	Kog	Agr	Kom	Kog
00	$\pi \rightarrow X_0$		07	+		14	-	
01	$X \rightarrow \pi_4$		08	$F \sqrt{ }$		15	$F \neq 0$	
02	$\pi \rightarrow X_4$		09	$\pi \rightarrow X_5$		16	π_3	
03	$F x^2$		10	X		17	$\pi \rightarrow X_2$	
04	B^+		11	e/n		18	$\pi \rightarrow X_4$	
05	$\pi \rightarrow X_3$		12	$\pi \rightarrow X_1$		19	+	
06	$F x^2$		13	$\pi \rightarrow X_4$		20	$X \rightarrow \pi_4$	

$X_1 \rightarrow R_0$

$X_2 \rightarrow R_1$

$\pi \rightarrow R_2$

$Z \rightarrow R_3$

$X \rightarrow R_4$

$e \rightarrow R_5$

Agr	Kom	Kog
21	π	
22	02	
23	e/n	
24		
25		
26		

get_token - наим. сущ. можн.

+ - * / % = ; () < > - DELIMITER

число - NUMBER

перем - VARIABLE

функция - FUNCTION

void level2(double *result)

{ register char op;
double hold;

level3(result);

while ((op = *token) = '+' || op == '-')

{ get_token();

level3(&hold);

arith(op, result, &hold);

}

void level3(double *result)

{ register char op;
double hold;

level4(result);

while ((op = *token) = '*' || op == '/' || op == '%')

{ get_token();

level4(&hold);

arith(op, result, &hold);

}

TRATZ

H X = H

S X = X

S * S X / dsh

S X = X

H + X = X

END

```
void level4 (double *result)
{
    double hold;
    levels(result);
    if (*token == '^')
    {
        get_token();
        level4(&hold);
        arith('^', result, &hold);
    }
}
```

```
void levels (double *result)
```

```
{ register char op;
op = 0;
if ((token-type == DELIMITER) && *token == '+' ||
```

```
*token == '-')
{
```

```
    op = *token;
    get_token();
}
```

```
level6(result)
```

```
if (op)
    unary(op, result)
}
```

~~if (*token~~

```
void level6 (double *result)
```

```
{ if ((*token == '(') && (token-type == DELIMITER))
    {
        get_token();
        level2(result);
        if (*token == ')')
            error(i);
    }
}
```

```
get_token();  
else primitive(result)  
{  
    if (token == NUMBER)  
        result = atof(token);  
    else if (token == FUNCTION)  
        result = fun();  
    else if (token == VARIABLE)  
        result = find_var();  
    else if (token == PRIMITIVE)  
        result = primitive();  
    else error(0);  
}
```

```
void unarg (char o, double *r)  
{
```

```
    if ((o == '-') * r = - (*r));  
}
```

```
void primitive (double *result)  
{
```

```
    char *t  
    switch (token_type)  
    {
```

```
        case VARIABLE : *result = find_var(); get_token();  
        case NUMBER : *result = atof(token); get_token(); return;  
        case FUNCTION : *result = fun(); get_token();  
        default : error(0);  
    }
```

```
    double *res; /* result */  
    if (o == '+') res = result + *result;  
    else if (o == '-') res = result - *result;  
    else if (o == '*') res = result * *result;  
    else if (o == '/') res = result / *result;  
    else if (o == '%') res = result % *result;
```

```
    while (o == '=')  
    {  
        if (get_token() != '=') error(0);  
        if (get_token() == '=') error(0);  
        if (get_token() == '=') error(0);  
    }
```

```
    if (o == '=')  
    {  
        if (get_token() == '=') error(0);  
        if (get_token() == '=') error(0);  
        if (get_token() == '=') error(0);  
    }
```

```
    if (o == '=')  
    {  
        if (get_token() == '=') error(0);  
        if (get_token() == '=') error(0);  
        if (get_token() == '=') error(0);  
    }
```

732

1000
1000
1000
1000
1000
<u>1000</u>

1000

полубогатый избыт

52525

125252

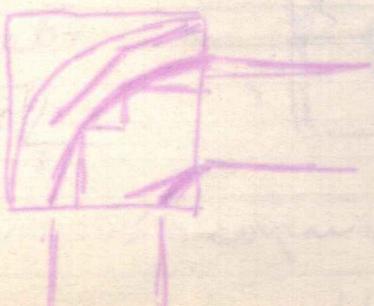
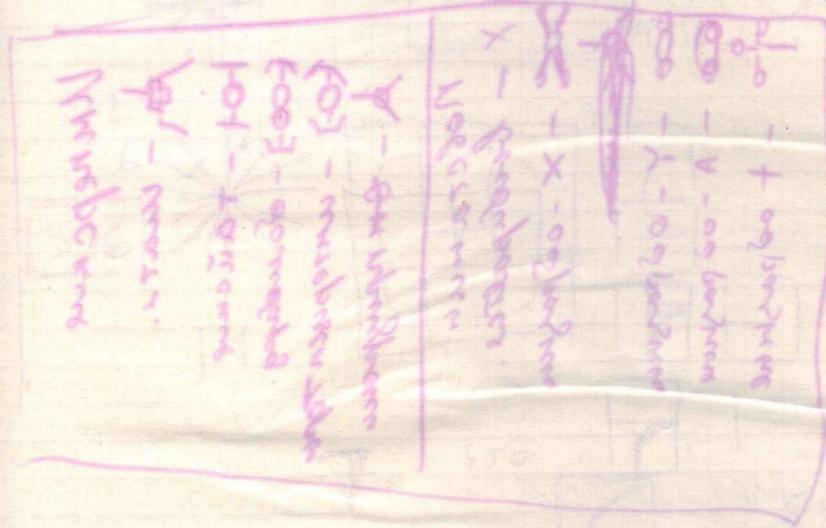
полупустой

52525

177777



45
Pausen
Raum



8 m²
8 m²
8 m²
8 m²

40
200
25
200

ca. 80000
ca. 100000

Ludop coprolo.

~~F. lopezi.~~

10 *fuscob*
coprolo.

luminesc

luminesc

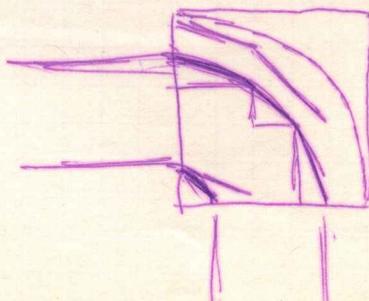
F. f. Mr. Gnozr

~~luminesc~~

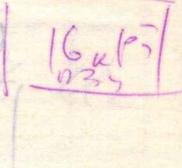
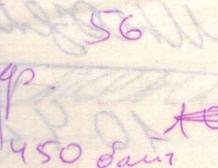
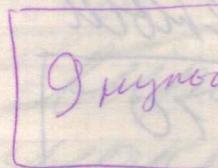
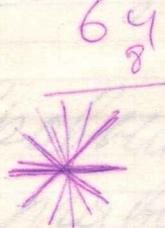
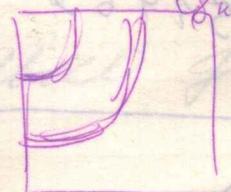
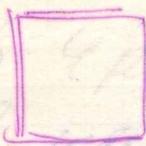
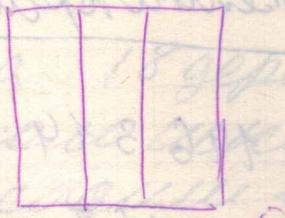
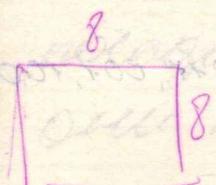
luminesc
fragilis

luminesc

37 *кбг* 9 *м/кг*
на 6 *квагр.*
8 *м/кг*.



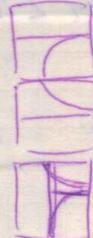
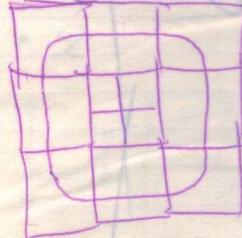
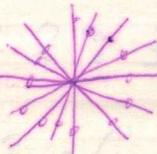
321



512

576

576



$$\begin{array}{r} 64 \\ \times 8 \\ \hline 512 \end{array}$$

32x57

14

см².

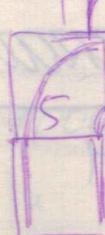
7

450

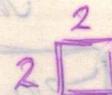
16x57

108187450
13

5



576



40

200 дм²

225

21 дм²

21 дм²

2



32 дм²

36 дм²

21 дм²

21 дм²

72 см².

36 см². 450 дм²

110 см². 227 см²

227 см²

Мемнон красный

146 3 14	
3 146 3	

001, 100, 110, 011, 001, 100
011 001 100

Мемнон зеленый

80	10 42 10
8-8	
80	10 42 10
80	
512	
512	

Мемнон синий

80	4 2 10 4
80	10 42 1

Решим задачу: Когданко
посадили 54 дерева 18 деревьев
они посадили сколько деревьев
а оставшееся посадили в парке

в Чурило . Сколько деревьев
в саженцах рябь

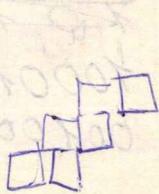
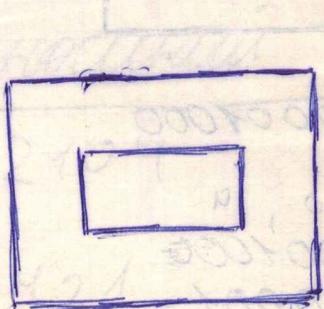
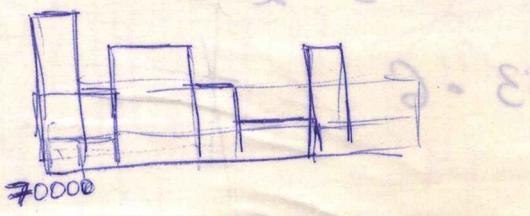
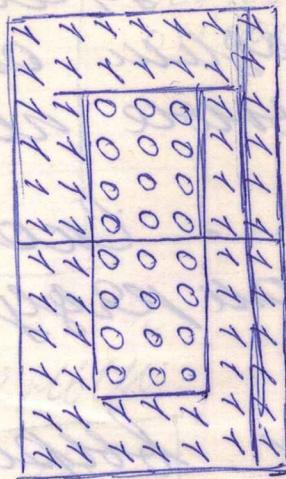
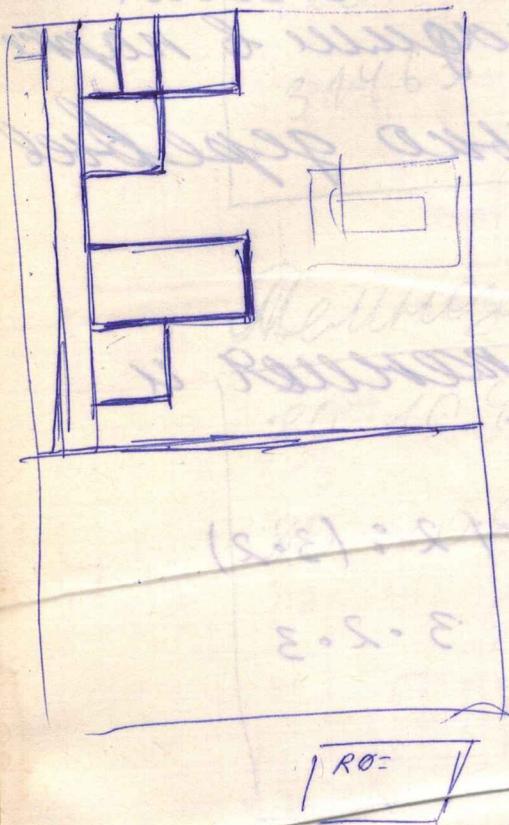
Сравни выражения и
поставь звёздки:

$$12 : 3 \cdot 2 \quad \boxed{12 : (3 \cdot 2)}$$

$$3 \cdot 6 \quad \boxed{\dots} \quad 3 \cdot 2 \cdot 3$$

1000100010001000

1000100010001000
001000100010001



63146

Рано проснем бот

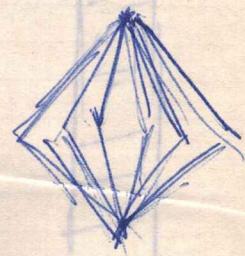
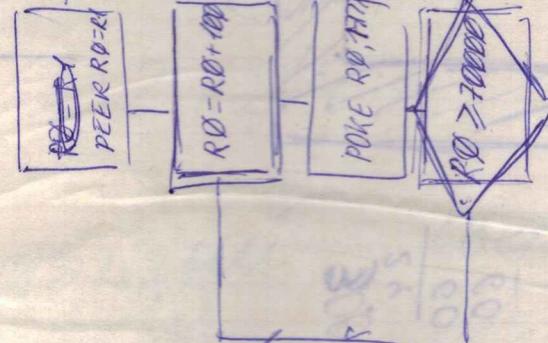
Балбамо -
неумоу

80	163346
80	134679
80	67156

151515

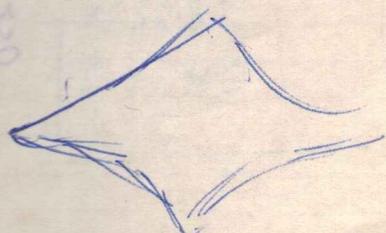
32323

96964



~~concametam~~

63146
119631



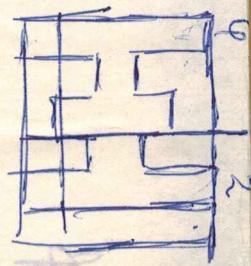
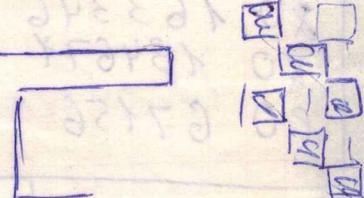
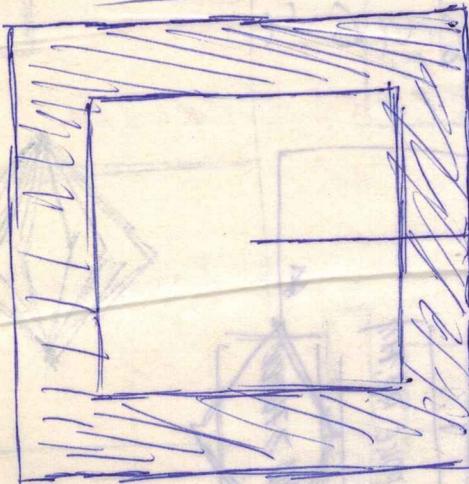
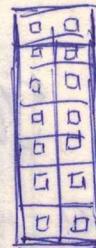
R
EMT 22

Белый с розовым
аметиком

80 155555

80 133333

80 66666



60.
60.
60.
24.
30000
250



~~50~~ 15 DIM A(N), B(N)
10 INPUT N
20 FOR I=1 TO N
30 INPUT A(I)
40 NEXT
50 INPUT K
60 FOR I=K TO N
70 B(I)=A(I)
80 NEXT
90 FOR I=1 TO N
100 PRINT A(I), B(I)
110 NEXT

26214
- 21847

26214
- 21847

80167356

80135673

Семнадцати

115151

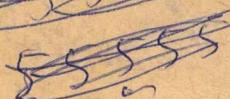
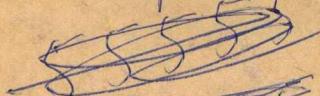
123232

69646

Сбемд

~~Дикорыч~~
~~Родина~~

135673



Розобоки

73567



SUPER

красный - 1

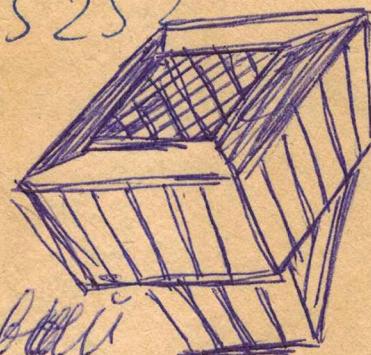
-21896

серый с зел. оттенком

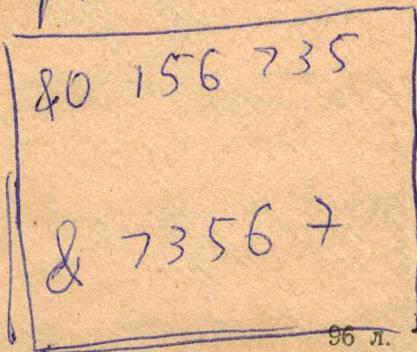
80114631

8063146

Зел 125252



ричалометровий



Ленинград
96 л. ТОСТ 13309-74 сорт 1
Арт. 6344р цена 44 коп.

FOR M% = 1 TO 4590
POKE 80177716, M%
POKE 80177716, M% * 3
POKE 80177716, 4C
NEXT



