

## ВХОДНОЙ ЯЗЫК ДЛЯ АВТОМАТИЧЕСКОГО ПРОГРАММИРОВАНИЯ АРИФМЕТИЧЕСКИХ ЗАДАЧ

М. КОТЛИ

Описываемый входной язык и его транслятор для модернизированной машины М-3 используются в Институте кибернетики Академии наук Эстонской ССР.

Входной язык предназначен для программирования задач арифметического характера и не подходит для записи логических задач. Арифметические выражения записываются на входном языке в форме, близкой к обыкновенной математической записи. Используются переменные и константы двух типов: действительные и целочисленные. Действительные переменные могут иметь до трех индексов, массивы — максимальную размерность три и циклы по параметру — максимальную кратность три. В пределах одной задачи можно использовать до семи разных функций ( $\sin x$ ,  $\sqrt{x}$  и т. д.) одного аргумента, заданных в машинном коде и фигурирующих на входном языке как одноместные операции. Входной язык позволяет использовать подпрограммы, записанные на входном языке. Имеется возможность ввода данных с перфоленты.

Длина транслятора входного языка, составленного для нашей машины, 1934 команды. Из-за ограниченности оперативной памяти машины (1024 ячейки на ферритах и 1024 на барабане), транслятор разделен на три части, хранящиеся во внешней памяти и работающие одна за другой. Внешняя память используется при трансляции и для хранения обрабатываемой программы.

Время трансляции одной программы, которая помещается в оперативной памяти, 5—25 мин.

Составленная программа работает при помощи специальной интерпретирующей программы. Однажды составленную программу можно использовать многократно.

Опыт применения входного языка и транслятора в Институте кибернетики показывает, что он: 1) позволяет программировать большинство небольших инженерных задач, 2) легко и быстро усваивается потребителями машины, 3) уменьшает время, затрачиваемое для составления и отладки программ.

### 1. Алфавит входного языка

Алфавит входного языка состоит из 63 символов, кодируемых двухзначными восьмеричными числами. Слова, напечатанные жирным шрифтом, понимаются как единые символы.

Символ	Код	Символ	Код	Символ	Код	Символ	Код
A	01	0	40	Q	21	—	62
B	02	1	41	R	22	—	63
		2	42	S	23	/	64
C	03	3	43	T	24	×	65
D	04	4	44	U	25		66
		5	45	V	26		67
E	05	6	46	X	27	Функции од- ного аргу- мента	70
F	06	7	47	Y	30		71
G	07	8	50	Z	31		72
H	10	9	51	блок	32		73
I	11	(	52	подпрограм-			74
J	12	)	53	ма	33	если 0	75
K	13	,	54	возврат	34	если +	76
L	14	*	55	начало цик-		переход	77
M	15	:	56	ла	35		
N	16	.	57	конец цикла	36		
O	17	=	60	целые	37		
P	20	+	61				

Символы алфавита делятся на 3 группы: буквы, цифры и разделители.

### 1.1. Буквы

<буква> ::= A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|

Q|R|S|T|U|V|X|Y|Z | блок | подпрограмма |

возврат | начало цикла | конец цикла | целые

### 1.2. Цифры

<цифра> ::= 0|1|2|3|4|5|6|7|8|9

### 1.3. Разделители

<разделитель> ::= (|)|,|\*|;|.|<элементарная операция>

<элементарная операция> ::= <одноместная операция> |

<двухместная операция> | если 0 | если + | переход | =

<одноместная операция> ::= — | <функция одного аргумента>

<двухместная операция> ::= + | — | | — || / | ×.

Функция одного аргумента есть произвольный символ (напр.,  $\sqrt{\quad}$ ,  $\sin$ ), кодируемый двухзначным восьмеричным числом из указанного диапазона.

Символ «|—|» обозначает вычитание модулей, «|/» деление и «|×» — умножение.

## 2. Слова

Для обозначения предметов входного языка из букв и цифр образуются слова.

<слово> ::= <буква> | <слово> <буква> | <слово> <цифра>

Примеры: A

B1

X2Y

блок 2

начало цикла A2C

Длина слова не ограничена, но два слова, в которых пять первых символов совпадают, воспринимаются программирующей программой как идентичные.

Девять слов имеют фиксированное значение, и их нельзя употреблять на входном языке в другом смысле.

Это:

<i>Слово</i>	<i>Русский перевод</i>
1. MASS	массивы
2. STP	стоп
3. TR	печать
4. SV	ввод
5. LOPP	конец программы
6. I	} параметры циклов
7. J	
8. K	
9. Целые	

Слова, начинающиеся буквами **блок**, **подпрограмма**, **возврат**, **начало цикла**, **конец цикла**, выделяются для обозначения разных наименований, как, например, наименования блоков, циклов, подпрограмм и т. д. Все остальные слова можно использовать для обозначения массивов и переменных.

### 3. Числа

$\langle \text{число} \rangle ::= \langle \text{целое число} \rangle | \langle \text{действительное число} \rangle$

$\langle \text{целое число} \rangle ::= \langle \text{цифра} \rangle | \langle \text{целое число} \rangle \langle \text{цифра} \rangle$

Примеры: 0  
21  
1963

$\langle \text{действительное число} \rangle ::= \langle \text{рациональное число} \rangle | \langle \text{число с плавающей запятой} \rangle$

$\langle \text{рациональное число} \rangle ::= \langle \text{целое число} \rangle , | \langle \text{целое число} \rangle , \langle \text{целое число} \rangle$

$\langle \text{число с плавающей запятой} \rangle ::= \langle \text{рациональное число} \rangle * \langle \text{порядок} \rangle$

$\langle \text{порядок} \rangle ::= \langle \text{целое число} \rangle | + \langle \text{целое число} \rangle | - \langle \text{целое число} \rangle$

Примеры: 0,  
3,  
4,51  
 $34,7 * -5$  ( $34,7 \cdot 10^{-5}$ )  
 $0,13 * +3$  ( $0,13 \cdot 10^{+3}$ )

Целые числа могут иметь значения от 0 до 9999999. Рациональные числа могут иметь максимально 7 значащих цифр. Если рациональное число длиннее, то последние цифры отбрасываются. Например, число 0,00342735795 воспринимается как  $0,3427357 \cdot 10^{-2}$ ; число 3452043400,1 — как  $0,3452043 \cdot 10^{10}$ .

#### 4. Наименования

$\langle \text{наименование массива} \rangle ::= \langle \text{слово} \rangle$

$\langle \text{наименование блока} \rangle ::= \text{блок} \mid \langle \text{наименование блока} \rangle \langle \text{буква} \rangle \mid \langle \text{наименование блока} \rangle \langle \text{цифра} \rangle$

$\langle \text{наименование цикла} \rangle ::= \text{начало цикла} \mid \langle \text{наименование цикла} \rangle \langle \text{буква} \rangle \mid \langle \text{наименование цикла} \rangle \langle \text{цифра} \rangle$

$\langle \text{наименование подпрограммы} \rangle ::= \text{подпрограмма} \mid \langle \text{наименование подпрограммы} \rangle \langle \text{буква} \rangle \mid \langle \text{наименование подпрограммы} \rangle \langle \text{цифра} \rangle$

$\langle \text{наименование возврата} \rangle ::= \text{возврат} \mid \langle \text{наименование возврата} \rangle \langle \text{буква} \rangle \mid \langle \text{наименование возврата} \rangle \langle \text{цифра} \rangle$

$\langle \text{конец цикла} \rangle ::= \text{конец цикла} \mid \langle \text{конец цикла} \rangle \langle \text{буква} \rangle \mid \langle \text{конец цикла} \rangle \langle \text{цифра} \rangle$

Примеры: **блок** 1  
**начало цикла** 2A  
**подпрограмма** AB  
**возврат** AB  
**конец цикла** 2A

#### 5. Переменные

$\langle \text{переменная} \rangle ::= \langle \text{простая переменная} \rangle \mid \langle \text{переменная с индексами} \rangle$

$\langle \text{простая переменная} \rangle ::= \langle \text{слово} \rangle$

$\langle \text{переменная с индексами} \rangle ::= \langle \text{наименование массива} \rangle (\langle \text{список индексов} \rangle)$

$\langle \text{список индексов} \rangle ::= \langle \text{индекс} \rangle \mid \langle \text{индекс} \rangle ; \langle \text{индекс} \rangle \mid \langle \text{индекс} \rangle ; \langle \text{индекс} \rangle ; \langle \text{индекс} \rangle$

$\langle \text{индекс} \rangle ::= \langle \text{параметр цикла} \rangle \mid \langle \text{целое число} \rangle \mid \langle \text{целое число} \rangle \times \langle \text{параметр цикла} \rangle \mid \langle \text{индекс} \rangle + \langle \text{индекс} \rangle \mid \langle \text{индекс} \rangle - \langle \text{индекс} \rangle$

$\langle \text{параметр цикла} \rangle ::= I \mid J \mid K$

Примеры: A  
 X1Y  
 A(0; 1)  
 B1(2 × K - 1; 3 + 2 × I - J; 10)  
 C(3 × J)

#### 6. Декларации

$\langle \text{декларация целых переменных} \rangle ::= \text{целые} * \langle \text{список целых переменных} \rangle$

$\langle \text{список целых переменных} \rangle ::= \langle \text{целая переменная} \rangle \mid \langle \text{список целых переменных} \rangle ; \langle \text{целая переменная} \rangle$

$\langle \text{целая переменная} \rangle ::= \langle \text{простая переменная} \rangle$

Пример:  $\text{целые} * A1; N; M; KI.$

Все переменные, перечисленные в декларации целых переменных, и, кроме того, параметры цикла  $I, J, K$  будут принимать в программе только целочисленные значения. Все остальные простые переменные и переменные с индексами принимают действительные (в форме с плавающей запятой) значения.

$\langle \text{декларация массивов} \rangle ::= \text{MASS} * \langle \text{список массивов} \rangle.$

$\langle \text{список массивов} \rangle ::= \langle \text{массив с размерами} \rangle | \langle \text{список массивов} \rangle$   
 $\langle \text{массив с размерами} \rangle$

$\langle \text{массив с размерами} \rangle ::= \langle \text{наименование массива} \rangle (\langle \text{список размеров} \rangle$

$\langle \text{список размеров} \rangle ::= \langle \text{целое число} \rangle | \langle \text{целое число} \rangle ; \langle \text{целое число} \rangle$   
 $| \langle \text{целое число} \rangle ; \langle \text{целое число} \rangle ; \langle \text{целое число} \rangle$

Пример:  $\text{MASS} * A(10) B1(3; 4) C(2; 5; 3).$

Массивы могут быть одно-, двух- или трехмерные. Если размеры массива  $m_1, m_2$  и  $m_3$ , то индексы элементов массива принимают значения  $0, 1, \dots, m_1 - 1; 0, 1, \dots, m_2 - 1$  и  $0, 1, \dots, m_3 - 1$ . Каждая переменная с индексами должна принадлежать к одному из массивов, перечисленных в декларации массивов.

Элементы массива упорядочены так, что в первую очередь изменяется последний индекс и последним — первый индекс.

Например, элементы массива  $B(2; 3)$  будут  $B(0; 0); B(0; 1); B(0; 2); B(1; 0); B(1; 1); B(1; 2).$

## 7. Арифметические выражения

$\langle \text{арифметическое выражение} \rangle ::= \langle \text{действительное арифметическое выражение} \rangle | \langle \text{целочисленное арифметическое выражение} \rangle$

$\langle \text{первичное выражение} \rangle ::= \langle \text{действительная переменная} \rangle | \langle \text{действительное число} \rangle | (\langle \text{действительное арифметическое выражение} \rangle) | \langle \text{одноместная операция} \rangle \langle \text{первичное выражение} \rangle$

$\langle \text{действительное арифметическое выражение} \rangle ::= \langle \text{первичное выражение} \rangle | \langle \text{действительное арифметическое выражение} \rangle$

$\langle \text{двухместная операция} \rangle \langle \text{действительное арифметическое выражение} \rangle$

$\langle \text{действительная переменная} \rangle ::= \langle \text{переменная с индексами} \rangle | \langle \text{простая переменная} \rangle$

Примеры: 2,

$-A1$

$(2, + A1 \times B) - 3, / B$

$\sin X + \cos \sqrt{(X + A)} \times B(I; J + 1)$

$\langle \text{целочисленное арифметическое выражение} \rangle ::= \langle \text{целое число} \rangle | \langle \text{целая переменная} \rangle | - \langle \text{целое число} \rangle | - \langle \text{целая переменная} \rangle | \langle \text{целое число или целая переменная} \rangle \langle +, -, \times \text{ или } / \rangle | \langle \text{целое число или целая переменная} \rangle$

Примеры: 2

$$\begin{aligned} I \\ -N \\ I+2 \\ J/N, \end{aligned}$$

где  $N$  переменная, фигурирующая в декларации целых переменных.  $I$  и  $J$  как параметры цикла всегда целочисленные переменные.

Старшинство операций следующее:

- |                       |                     |
|-----------------------|---------------------|
| 1. одностная операция | 5. —                |
| 2. $\times$           | 6. $+$              |
| 3. $/$                | 7. $=$              |
| 4. $ - $              | 8. если 0, если $+$ |

Операции одинаковых рангов выполняются в последовательности их появления в выражении.

В следующем примере отмечен порядок выполнения операции:

$$\begin{array}{cccccccccc} 8 & 5 & 6 & 3 & 4 & 2 & 1 & 7 & 9 \\ -(A - B - C \times A/B \times \sin X + C) + A. \end{array}$$

## 8. Операторы

$\langle \text{оператор} \rangle ::= \langle \text{оператор присваивания} \rangle | \langle \text{оператор перехода} \rangle |$   
 $| \langle \text{оператор ввода} \rangle | \langle \text{оператор вывода} \rangle | \langle \text{оператор останова} \rangle |$   
 $| \langle \text{оператор обращения к подпрограмме} \rangle$

### 8.1.

$\langle \text{оператор присваивания} \rangle ::= \langle \text{действительная переменная} \rangle = \langle \text{действительное арифметическое выражение} \rangle | \langle \text{целая переменная} \rangle = \langle \text{целое арифметическое выражение} \rangle | \langle \text{действительная переменная} \rangle = \langle \text{целое число или целая переменная} \rangle$

Последняя форма оператора присваивания предусмотрена для перехода от целочисленных величин к действительным.

Примеры:  $A1 = -B \times \sin(X - A)$   
 $I = J \times K$   
 $B1(I; J; K - 1) = B1(I; J; K) - 2, \times A$   
 $A = J.$

Целочисленными здесь являются  $I, J, K.$

### 8.2.

$\langle \text{оператор перехода} \rangle ::= \langle \text{оператор условного перехода} \rangle | \langle \text{оператор безусловного перехода} \rangle$

$\langle \text{оператор условного перехода} \rangle ::= \langle \text{арифметическое выражение} \rangle$   
 $\langle \text{операция условного перехода} \rangle \langle \text{наименование блока, цикла или возврата} \rangle | \langle \text{оператор присваивания для действительной переменной} \rangle \langle \text{операция условного перехода} \rangle \langle \text{наименование блока, цикла или возврата} \rangle$

$\langle \text{операция условного перехода} \rangle ::= \text{если } 0 | \text{если } +$

Примеры:  $I$  если 0 блок  $A$

$K - J$  если + возврат 2

$Y = \sin X - B$  если + начало цикла  $C$

$\langle$ оператор безусловного перехода $\rangle ::=$  переход  $\langle$ наименование блока, цикла или возврата $\rangle$

Примеры: переход блок  $A$

переход возврат  $C1$

8.3.

$\langle$ оператор останова $\rangle ::= STP | STP \langle$ номер останова $\rangle$

$\langle$ номер останова $\rangle ::= \langle$ целое число $\rangle$

8.4.

$\langle$ оператор ввода $\rangle ::= SV * \langle$ переменная $\rangle | SV * \langle$ переменная с индексами $\rangle \langle$ переменная с индексами $\rangle \langle$ шаг $\rangle$

$\langle$ шаг $\rangle ::= \langle$ целое число $\rangle | \langle$ целая переменная $\rangle$

Примеры:  $SV * I$

$SV * A1 (I; J; K)$

$SV * A (0; 0) A (9; 9) 1$

$SV * B (0; 0) B (J; 0) 5.$

Оператор ввода первого типа вводит с перфоленты одно число и присваивает его значение указанной переменной. Оператор ввода второго типа вводит с перфоленты массив чисел и присваивает их значения элементам массива, первый и последний элемент которого указаны. Шаг указывает шаг движения по массиву. Например, в случае третьего примера, если размеры массива  $A$   $m_1 \geq 10$  и  $m_2 = 10$ , то вводятся

$A (0; 0) A (0; 1) \dots A (0; 9)$

$A (1; 0) A (1; 1) \dots A (1; 9)$

$A (9; 0) A (9; 1) \dots A (9; 9).$

В случае четвертого примера вводятся

$B (0; 0) B (1; 0) B (2; 0) \dots B (J; 0),$

если размеры массива  $B$  есть  $m_1 \geq J$  и  $m_2 = 5$ .

8.5.

$\langle$ оператор вывода $\rangle ::= TR * \langle$ переменная $\rangle | TR * \langle$ переменная с индексами $\rangle \langle$ переменная с индексами $\rangle \langle$ шаг $\rangle$

Примеры:  $TR * K$

$TR * A (J; K; 0)$

$TR * A (0; 0; 0) A (1; 3; 5) 1.$

Оператор вывода печатает значение указанной переменной или значения элементов массива, первый и последний из которых указаны. Шаг означает шаг движения по массиву.

8.6.

<оператор обращения к подпрограмме> ::= <наименование подпрограммы>

Пример: подпрограмма АВ.

## 9. Блоки

<блок> ::= <наименование блока> \* <совокупность операторов>.

<совокупность операторов> ::= <оператор> | <совокупность операторов>; <оператор>

Пример: блок 1 \*  $Y = A$ ;

$B = Y + C/2$ ;

$N1 = \cos B$  если + блок В;

$TR * N1$ ;

STP.

## 10. Циклы по параметру

<цикл> ::= <заголовок цикла> <совокупность блоков> <конец цикла>.

<заголовок цикла> ::= <наименование цикла> (<параметр цикла>; <начальная величина параметра>; <шаг параметра>; <конечная величина параметра>)

<начальная величина параметра> ::= <целое число> | <целая переменная>

<шаг параметра> ::= <целое число> | <целая переменная>

<конечная величина параметра> ::= <целое число> | <целая переменная>

<совокупность блоков> ::= <блок> | <цикл> | <совокупность блоков> <блок> | <совокупность блоков> <цикл>

Пример: начало цикла  $A(I; 0; H; N)$

блок 1 \*  $A(I) = X + \sin Y - B(I - 1)$ .

начало цикла  $B(J; 1; 2; I)$

блок В \*  $C(I; J) = A(I) \times A(J)$ ;  $TR * C(I; J)$ .

конец цикла В.

конец цикла А.

$H$  и  $N$  в примере целочисленные переменные, декларированные раньше. Так как для обозначения параметров цикла выделено только три переменных  $I$ ,  $J$  и  $K$ , то можно образовать одно-, двух- или трехкратные циклы. Если кратность циклов больше трех, то нужно образовать некоторые циклы другими средствами, например при помощи счетчиков.

## 11. Подпрограммы

$\langle \text{подпрограмма} \rangle ::= \langle \text{список деклараций} \rangle \langle \text{наименование подпрограммы} \rangle * \langle \text{совокупность блоков} \rangle \langle \text{наименование возврата} \rangle$

$\langle \text{список деклараций} \rangle ::= \langle \text{пусто} \rangle | \langle \text{список деклараций} \rangle \langle \text{декларация} \rangle$

Пример: `целые * N . MASS * A (20).`  
`подпрограмма AB * блок A * ... ; ... ; ...`  
`блок B * ... ; ... ; ...`  
`начало цикла 1 (I; 0; 1; N)`  
`блок 1 * ... ; ... ; ...`  
`блок 2 * ... ; ... ; ...`  
`конец цикла 1.`  
`блок C * ... ; ... ; ...`  
`возврат AB.`

## 12. Программа

$\langle \text{программа} \rangle ::= \langle \text{список деклараций} \rangle \langle \text{совокупность подпрограмм} \rangle \langle \text{совокупность блоков} \rangle \text{LOPP.}$

$\langle \text{совокупность подпрограмм} \rangle ::= \langle \text{пусто} \rangle | \langle \text{совокупность подпрограмм} \rangle \langle \text{подпрограмма} \rangle$

Пример: `целые * M; N.`  
`MASS * A(3; 2; 4) B(4).`  
`подпрограмма A * ... возврат A.`  
`подпрограмма B * ... возврат B.`  
`целые * O; P.`  
`блок 1 * ... ; ... ; ...`  
`блок 2 * ... ; ... ; ...`  
`начало цикла 1 ( ; ; )`  
`блок 3 * ... ; ... ; ...`  
`конец цикла 1.`  
`LOPP.`

В программе все подпрограммы должны стоять перед основной программой. При этом в каждой подпрограмме можно обращаться к подпрограммам, стоящим как впереди, так и за этой подпрограммой. Декларации массивов и целых переменных должны стоять перед подпрограммой или программой, в которой первый раз используются элементы, перечисленные в декларациях массивов или целых переменных.

## 13. Примеры программ

а) Скалярное произведение векторов

С перфоленты вводятся уменьшенная на единицу длина векторов  $K < 30$ ,  $K + 1$  элементов вектора  $A$  и  $K + 1$  элементов вектора  $B$ . Скалярное произведение печатается.

MASS \* A(30) B(30).

блок 1 \* SV \* K;

SV \* A(0) A(K) 1;

SV \* B(0) B(K) 1;

S = 0.

начало цикла 1 (J; 0; 1; K)

блок 2 \* S = S + A(I) × B(I).

конец цикла 1.

блок 3 \* TR \* S; STP.

LOPP.

б) Вычисление интеграла

$$J(R) = \int_0^{\pi} \frac{dV}{R^2 - 2R \cos V + 1}$$

для  $R = 0,01; 0,02; \dots; 0,99$  по методу Симпсона (см. Автоматизация программирования, М., 1961, стр. 330). С перфоленты водится увеличенная в три раза точность EPS. Для каждого значения параметра  $R$  печатаются значения параметра и интеграла.

целые \* N.

подпрограмма F \*

блок F \* F = 1, / (R × R - 2, × R × cos V + 1).

возврат F.

блок 1 \* SV \* EPS;

R = 0,01.

блок 2 \* IRMAX = 4, / R;

N = 1;

H = 1,570796;

V = 0,;

подпрограмма F;

F1 = F;

V = 3,141593;

подпрограмма F;

Y = H × (F × F1).

блок 3 \* S = 0.

начало цикла 1 (K; 1; 1; N)

блок 4 \* K1 = K;

V = (2, × K1 - 1,) × H;

подпрограмма F;

S = S + F.

конец цикла 1.

блок 5 \* IR = Y + 4, × H × S;

EPS |—| (IR - IRMAX) если + блок 6;

IRMAX = IR;

Y = (IR + Y) / 4,;

N = 2 × N;

H = 0,5 × H;

переход блок 3.

блок 6 \* IR = IR / 3,;

TR \* R;

TR \* IR;

R = R + 0,01;

0,995 — R если + блок 2;

STP.

LOPP.

Институт кибернетики  
Академии наук Эстонской ССР

Поступила в редакцию  
8 II 1964

SISENDKEEL ARITMEETILISTE ÜLESANNETE AUTOMAATSEKS  
PROGRAMMEERIMISEKS

M. Kotli

*Resümee*

Esitatakse ENSV TA Küberneetika Instituudis kasutatava programmeeriva programmi sisendkeel arvuti M-3 jaoks.

Sisendkeel võimaldab aritmeetilist laadi ülesandeid kirja panna harilikule matemaatilisele kirjutusviisile lähedasel kujul.

*Eesti NSV Teaduste Akadeemia  
Küberneetika Instituut*

Saabus toimetusse  
8. II 1964

PROGRAMMING LANGUAGE FOR AUTOMATIC PROGRAMMING  
OF ARITHMETICAL PROBLEMS

M. Kotli

*Summary*

In this article a programming language for computer M-3 used by the Institute of Cybernetics (Academy of Sciences of the Estonian S.S.R.) is presented.

*Academy of Sciences of the Estonian S.S.R.,  
Institute of Cybernetics*

Received  
Feb. 8th, 1964