

19/2114

C-289

ОБЪЕДИНЕННЫЙ
ИНСТИТУТ
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ

Дубна

11 - 4117

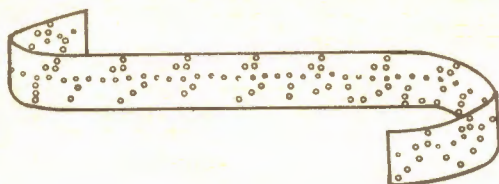


ЛАБОРАТОРИЯ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ
И АВТОМАТИЗАЦИИ

З.Секера

СТАНДАРТНЫЕ ПРОЦЕДУРЫ ОБРАЩЕНИЯ
К АЦПУ ДЛЯ ТРАНСЛЯТОРА С АЛЬФА-ЯЗЫКА

1968

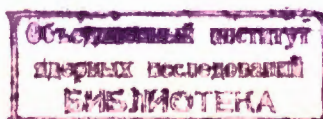


Объединенный институт
ядерных исследований
ЛВТА

11 - 4117

3.Секера

СТАНДАРТНЫЕ ПРОЦЕДУРЫ ОБРАЩЕНИЯ
К АЦПУ ДЛЯ ТРАНСЛЯТОРА С АЛЬФА-ЯЗЫКА



7551/2 up.

А. Введение.

В трансляторе с АЛФА-языка /I/ нет стандартных программ обращения к АЦПУ. Чтобы избежать изменения библиотеки стандартных программ, было принято решение использовать возможности вставки неописанных процедур. За основу наших процедур обращения к АЦПУ была принята система процедур транслятора с АЛГОЛа для машины *GIER*, поэтому идентификаторы некоторых процедур совпадают.

Б. Описание процедур.

procedure outsp (a);

1) Параметры: а - integer, value

внешние - см. часть В.

2) Действие: с места последнего отпечатанного символа ответится на АЦПУ "а" пробелов.

3) а - любое, в случае, если $a < 0$, полагается $a=0$ в теле процедуры.

procedure outex (a);

1) Параметры: а - integer, value

внешние параметры; см. часть В.

2) Действие: отпечатает сформированную строку на АЦПУ (если такая есть) и сделает переход на "а" строк.

3) Замечание:

а) должно быть $1 \leq a \leq 128$, иначе в теле процедуры полагается $a = 1$;

procedure outch (a);

1) Параметры: а - integer, value

внешние параметры: см. часть В.

2) Действие: на АЦПУ отпечатается символ, десятичный код которого равен "а". Таблицу кодов символов см. ниже.

3) Замечание:

Если $a < 0$ или $a > 127$, то процедура ничего не печатает (работает как пустая).

procedure outpt (*a*, *b*, *c*, *d*, *bool*);

1) Параметры: *a* - real, value

b, c, d - integer, value

bool - boolean, value

(внешние параметры-см. часть В).

2) Действие: отпечатает на АЦПУ число "а" с определенным форматом, который задается параметрами *b, c, d*. Параметр *bool* влияет только на печать знака числа (см. ниже); перед печатью "а" округляется.

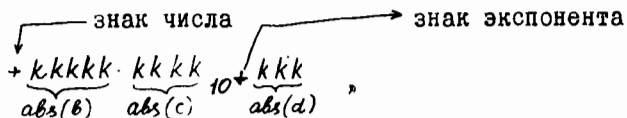
3) Границы параметров: $|b| \leq 7$, $|c| \leq 7$, $|d| \leq 7$.

Если $c = 0$, не печатается десятичная точка.

Если $d = 0$, не печатается 10^+

4) Формат числа:

Число "а" представим в виде:



где "k" - некоторая цифра.

Потом

$abs(b)$ определяет число цифр перед десятичной точкой;

$abs(c)$ _____ " _____ за десятичной точкой;

$abs(d)$ _____ " _____ экспонента.

Отведение места на знак:

Место на печать знака числа отводится в зависимости от параметров *b, c, d, bool* таким образом:

$b < 0$ или ($b = 0$ и $c \leq 0$) или ($b = 0$ и $c = 0$) -

место на знак отводится.

Влияние параметра „bool“

$bool = false$ - знак "-" всегда печатается перед первой цифрой числа, знак "+" не печатается никогда, причем если на знак требуется отвести место (см. выше), то отводится один пробел.

$bool = true$ - всегда печатается знак "+" или "-" в зависимости от знака печатаемого числа "a", знак печатается в качестве первого знака формата (см. примеры ниже).

Параметр: „d“

Определяет число цифр, отведенных для экспонента.

Процедура пытается поместить число в формат, в котором сохранено прежнее значение "b" и "c", а "d" полагается = 0. В случае, когда этого нельзя сделать, экспонент выбирается так, чтобы использовался данный формат по мере возможностей (смотри примеры ниже). Экспонент всегда печатается в виде 10^{+kk} или 10^{-kkk} (знак "d" не имеет значения).

Параметр "c":

кроме влияния на отведение знака (см. выше) имеет еще другое значение.

Число $a < 1$ печатается в случаях:

- 1) $b \neq 0$ и $c \geq 0$ с цифрой "0" перед десятичной точкой;
- 2) $c < 0$ без цифры "0" перед десятичной точкой.

Если число "a" не помещается в данный формат, то добавляется экспонент (произойдет сдвиг следующего числа или символа на 3 или 4 места вправо) (экспонент добавляется в виде 10^{+k} или 10^{-kkk}). Если параметры формата подобраны так, что не должно быть отведено место для знака и "a" оказывается отрицательным, то знак "-" отпечатается, но затем произойдет сдвиг данного числа на бумаге на одно место вправо.

procedure word (a, m, n);

1) Параметры: a - array
m, n - integer, value

внешние - см. часть В.

2) Действие: отпечатается любой текст, помещенный в согласии с семиразрядной упаковкой в массиве " a ", начиная с ячейки a[m] по a[n] (включительно).

3) Использование: в начале программы удобно ввести все тексты, заранее пробитые на УПП, в массив " a " (обязательно ввод без перевода - СП 0176 (16, a/I/, a[...], 0, 0);) и потом при обращении к процедуре word определить желаемые параметры " m " и " n ".

В. Внешние параметры.

В начале каждой программы, которая использует хотя бы одну из предлагаемых процедур обращения к АЦПУ, должно быть следующее описание и определение начальных условий:

```
begin array line [-9 : 21];  
line [-2] := line [-3] := 0; line [-1] := - 10;  
go to a 100; outsp (1); outch (1); outpt (0,5, 1,1,0, false);  
outor (1); split(line,1,1,line,1,6); pack(line,-9,-4,line,1,6); a 100;
```

Последние две строки обеспечивают правильную вставку взаимно сильно связанных процедур (требуется в силу особенностей оглавления библиотеки стандартных процедур АЛФА-транслятора). Эти команды в рабочей программе не будут (27 блок их во время трансляции отбросит).

Г. Запись процедур на ленту библиотеки стандартных программ.

Процедуры записываются на ленту обычным образом (см. работу /1/) в оглавлении не учитывается использование других стандартных процедур.

Замечание: процедуры " *pack* " и " *split* " описаны в работе /2/.

Д. ЛИТЕРАТУРА

- [1] - АЛЬФА-система автоматизации программирования, под редакцией А.П.Ершова, издание СО АН СССР, Новосибирск(1966) .
- [2] - М. Джгаркава, З. Секера. Процедуры сжатия, "распаковки" информации и использования КЗУ и барабанов в программах на АЛЬФА-языке, Дубна (1968) г.

Е. Таблица десятичных кодов и соответствующих символов АЦПУ.

КОД	СИМВОЛ	КОД	СИМВОЛ	КОД	СИМВОЛ	КОД	СИМВОЛ
0	0 цифра	20	x	40	u	60	э
1	1	21	=	41	й	61	ю
2	2	22	;	42	к	62	я
3	3	23	ℓ	43	л	63	д
4	4	24	Ј	44	м	64	ф
5	5	25	ж	45	н	65	г
6	6	26	‘	46	буква о	66	и
7	7	27	”	47	п	67	ј
8	8	28	≠	48	р	68	л
9	9	29	<	49	с	69	н
I0	+	30	>	50	т	70	q
II	-	31	:	51	у	71	р
I2	/	32	А	52	ф	72	с
I3	запятая ,	33	Б	53	х	73	u
I4	.	34	В	54	ц	74	v
I5	⌋	35	Г	55	ч	75	w
I6	10	36	Д	56	ш	76	z
I7	↑	37	Е	57	щ	77	чёрта сверху
I8	(38	Ж	58	ы	78-127	пустое место на бумаге
I9)	39	З	59	ь		

Ж. Тексты процедур.

```

procedure outch(a); value a;
begin integer s;
if a<0va>127 then go to al;
line[-1]:=line[-1]+1; s:=line[-1]; line[s]:=a;
if s=-4 then
begin line[-2]:=line[-2]+1; s:=line[-2];
pack(line,-9,-4,line,s,6); line[-1]:=-10;
end;
line[-3]:=line[-3]+1; s:=line[-3];
if s=124 then outcr(1);
al: end;

```

```

procedure outsp(a); value a;
a times do outch(127);

```

```

procedure outcr(a); value a;
begin integer z,p;
if a<1va>127 then a:=1; p:=line[-1];
if p=-5 then
begin line[-4]:=127; line[-2]:=line[-2]+1;
z:=line[-2]; pack(line,-9,-4,line,z,6); p:=-10;
end;
for z:=1,...,-6-p do line[p+z]:=127;
line[-5]:=104; line[-4]:=a-1; line[-2]:=line[-2]+1;
z:=line[-2]; pack(line,-9,-4,line,z,6);
line[-2]:=line[-3]+0; line[-1]:=-10;
сдвиг(193,line[1],line[z],0,0);
end;

```

```

procedure outpt(a,b,c,d,bool); value a,b,c,d,bool ;
begin integer p,k,j,n,g,f,i; boolean sa,sb,sc;
sa:=a<0; a:=abs(a); sb:=b<0; b:=abs(b);
sc:=c<0; c:=abs(c); d:=abs(d); i:=1;
a:=a-frac(a)+okp(frac(a) $\times$ 10fc)/10fc+ $10^{-8}$ ;
al: for j:=-19,j+1 while a>(1- $10^{-8}$ ) $\times$ 10fj do n:=j;
f:=n+1-b; g:=0;
if f>0 and i<3 then
begin a:=a/10f; g:=f; n:=n-f; end else
if d#0 and n<0 then
begin g:=n+1; a:=a/10g; n:=-1 end;
if bool then
begin p:=if sa then 11 else 10; outch(p) end;
p:=b-(if n>0 then n+1 else if sc then 0 else 1);
p times do outch(127);
if  $\neg$ bool then
begin if sa then outch(11) else
if sbvb=0 and (scvc=0) then outch(127) end;
if i=1 and a<0 and b>0 and  $\neg$ sc then outch(0);
a3: for j:=0,...,n do
begin p:=10f(n-j); k:=entier(a/p); outch(k);
a:=a-kxp; if a<0 then a:=0 end;
if i=1 and c>0 then
begin i:=2; outch(14); a:=okp(a $\times$ 10fc);
n:=c-1; go to a3 end;
if i<3 then
begin if g#0 then
begin bool:=true; outch(16); a:=abs(g); b:=abs(d);
sa:=g<0; sb:=sc:=false; c:=1; i:=3; go to a1
end else if d>0 then
begin d+2 times do outch(127) end
end end;

```

```

procedure word(c,m,n); value m,n;
begin array b[1:6]; integer i,j,k;
for i:=0,...,n-m do
begin k:=m+i; split(c,k,k,b,1,6);
for j:=1,...,6 do begin k:=b[j]; outch(k) end
end end;

```

Примеры.

Запятые приведены только для наглядного определения отведенного места на бумаге.

```

a = -3876.6823

```

b	c	d	bool = false	bool = true	Примечание.
-5	0	0	, -3877,	, - 3877,	округление
-5	1	0	, -3876.7,	, - 3876.7,	округление
-5	2	I	, -3876.68	, -33876.68	10^0 никогда не печатается, только отводится место.
-3	2	0	, -387.67 ₁₀ +I,	, -387.67 ₁₀ +I,	добавление экспоненты, число занимает на три позиции больше, неправильно подобраны переменные.
-3	2	I	, -387.67 ₁₀ +I,	, -387.67 ₁₀ +I,	для экспонента отведено место.
0	-3	3	, - 388 ₁₀ + 4,	, - 388 ₁₀ + 4,	
4	4	0	, -3876.6823,	, -3876.6823,	сдвиг вправо из-за знака.

```

a = +3876.6823

```

-5	0	0	, 3877,	, + 3877,	
-5	1	0	, 3876.7,	, + 3876.7,	

```

a = 0.68720010-4

```

5	2	0	, 0.00,	, + 0.00,	сдвиг направо из-за знака (bool = true).
5	2	I	, 0.69 ₁₀ -4,	, + 0.69 ₁₀ -4,	см. выше.
5	-2	I	, .69 ₁₀ -4,	, + .69 ₁₀ -4,	см. выше.

Рукопись поступила в издательский отдел
18 октября 1968 года.