

УДК 681.3.06 : 51

# ТРАНСЛЯЦИЯ УКРУПНЕННЫХ КОНСТРУКЦИЙ ЯЗЫКА ДЕЛЬТА-ЭПСИЛОН

Л.Т.Петрова

Здесь будут рассмотрены некоторые вопросы трансляции одной из укрупненных конструкций языка ДЕЛЬТА-ЭПСИЛОН [1] на язык ЭПСИЛОН [2]. Этой конструкцией является описание списка-схемы. Пусть, например, список  $S$  описан как список-схема:

$$\begin{aligned} \text{список-схема } S &= D \cup A, \\ A &= D \otimes D, \\ D &= B \parallel C; \end{aligned}$$

Списочные операции  $\cup$ ,  $\otimes$  и  $\parallel$  определены в [1]. На рисунке 1 приводится граф старшинства для списка  $S$ .

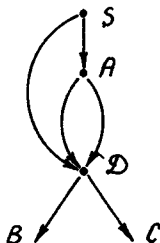


Рис. 1.

Описание списка-схемы определяет собой правило формирования значения этого списка из значений опорных списков. Так, значение списка  $S$  строится в конечном счете из значений

опорных списков  $B$  и  $C$ . Списки  $A$  и  $D$  в нашем примере, так же как и список  $S$ , являются динамическими объектами, их значение не хранится в развернутом виде в памяти машины. При ссылках на значение списка-схемы это значение получается последовательными квантами (последовательно) и, таким образом, "прочитывается".

Для реализации этой возможности при трансляции описания списка-схемы строится замкнутая процедура выборки элемента списка-схемы по номеру этого элемента. Эта главная процедура содержит внутри себя ссылки на подчиненные ей процедуры, которые также строятся при трансляции описания. Кроме того, строятся некоторые операторы, которые раз навсегда вычисляют ряд характеристик для этих процедур. Все эти конструкции вместе и составляют  $\mathcal{E}$ -образ описания списка-схемы. Далее подробно излагается структура этого образа, а также описан процесс трансляции и приведены основные транслирующие процедуры.

## § 1. Обозначения и предположения

Данная реализация языка ДЕЛЬТА-ЭПСИЛОН ( $\Delta\mathcal{E}$ ) естественным образом опирается на конкретную реализацию языка ЭПСИЛОН на машинах типа М-20 и БЭСМ-6 [2]. В частности, в данной реализации языка  $\mathcal{E}$  имеется ограничение на разрядность элемента списка - последняя не может превосходить разрядности машинного слова (ячейки). Это ограничение сохраняется для опорных списков в языке  $\Delta\mathcal{E}$ , но оно снимается для списков-схем. Однако, здесь необходимо принять некоторые соглашения. Будем предполагать, что значение указанного элемента списка-схемы может формироваться в нескольких подряд расположенных ячейках машинной памяти, при этом элемент каждого опорного списка, фигурирующего в схеме, занимает отдельную ячейку. Память, отведенная для записи формируемых элементов списка-схемы, формально организована в ячеечный список  $\Delta$ . Например, при формировании  $k$ -того элемента динамического списка  $Z = \mathcal{X} \parallel \mathcal{Y}$  значение  $Z[k]$ , образованное опорными значениями  $\mathcal{X}[k]$  и  $\mathcal{Y}[k]$ , хранится в двух ячейках  $\Delta[1]$  и  $\Delta[2]$ , причем  $\Delta[1] = \mathcal{X}[k]$ ,  $\Delta[2] = \mathcal{Y}[k]$ .

Для схем, допускаемых в языке  $\Delta\mathcal{E}$  (им отвечают графы без контуров), всегда возможно расположить строки в порядке вы-

числения левых частей этих строк. В любой строке упорядоченной схемы каждый ее аргумент является либо опорным, либо левой частью одной из предыдущих строк. Будем считать, что схема упорядочена так, что вычисления выполняются с конца схемы, от последней строки к первой. В нашем примере это условие выполнено; оно будет нарушено, если, например, вторую и третью строки поменять местами. Для упорядоченной схемы  $\mathcal{E}$  - образ ее описания можно построить за один просмотр схемы. При этом  $\mathcal{E}$  - образ схемы последовательно составляется из образов отдельных строк.

С каждым динамическим списком  $Z$  будут связаны следующие служебные объекты и конструкции:

$\Delta BZ$  - имя замкнутой процедуры, осуществляющей выборку нужного элемента динамического списка  $Z$ ; номер этого элемента является единственным фактическим параметром процедуры, соответствующий формальный параметр носит имя  $\Delta KZ$ , так что заголовок процедуры имеет вид  $\Delta BZ (\Delta KZ)$ ;

$\Delta \mathcal{A}Z$  - имя замкнутой процедуры, осуществляющей вычисление длины списка  $Z$ ;

$\Delta nZ$  - скаляр - длина списка  $Z$ ;

$\Delta KZ$  - скаляр - номер выбираемого элемента списка  $Z$ ;

$\Delta \delta Z$  - скаляр - количество ячеек в списке  $\Delta$ , занятых элементом списка  $Z$  (количество опорных списков в схеме  $Z$ );

$\Delta \alpha Z$  - скаляр - относительный адрес (индекс) первой ячейки в списке  $\Delta$ , занятой выбираемым элементом списка  $Z$  (таким образом, значение элемента  $Z[\Delta KZ]$  формируется в ячейках  $\Delta[\Delta \alpha Z]$ , ...,  $\Delta[\Delta \alpha Z + \Delta \delta Z - 1]$ );

$\Delta \alpha Z$ ,  $\Delta \omega Z$ ,  $\Delta \Omega Z$  - метки;

$\Delta \mathcal{Z}$  - скаляр - выполняет функции рабочей ячейки.

Объекты  $\Delta nZ$  и  $\Delta \alpha Z$  соотносятся к каждому из опорных списков. Отметим, что все служебные имена начинаются с большой греческой буквы "дельта" ( $\Delta$ ). Тем самым в языке становится небезопасным использование имен, начинающихся с этой буквы. Так же, как и в [2], все имена в данной реализации языка  $\Delta \mathcal{E}$  различаются по первым пяти символам. Введенные выше служебные имена позволяют сохранить в именах списков, фигурирующих в схемах, не более двух символов (например, в имени процедуры выборки  $\Delta BZ$  первые три символа  $\Delta B$  являются обязательными служебными, и только два последних символа берутся из имени списка). В приведенных далее процедурах

трансляции (§ 3) для простоты предполагается, что любой список, фигурирующий в схеме, обозначается одним символом.

## § 2. Структура $\xi$ -образа описания списка-схемы

Образ описания списка-схемы состоит из трех основных конструкций. Во-первых, это главная процедура выборки элемента списка-схемы ( $\Delta BZ$ ) с цепочкой подчиненных ей таких же процедур для всех динамических списков из той же схемы. Каждая из этих процедур формируется при анализе очередной строки схемы. Процедуры выборки составляют список ВМБ. Второй конструкцией образа является процедура вычисления длины списка-схемы ( $\Delta \Delta Z$ ). Процедура образуется из операторов, которые включаются в нее при анализе каждой строки схемы. Эти операторы составляют список ДЛ. Наконец, третьей конструкцией образа является оператор вычисления количества ячеек ( $\Delta \delta Z$ ), занятых значением элемента списка-схемы. Этот оператор состоит из последовательности аналогичных операторов для всех динамических списков данной схемы. Эти составляющие операторы образуются при анализе очередной строки схемы и выписываются в список ШИР или сразу в текст (вместо описания списка-схемы).

После завершения анализа всех строк схемы списки ДЛ и ВМБ также переписываются в текст (следом за списком ШИР).

В таблице I приводятся образы отдельных аргументов (опорных и динамических) для всех списочных операций. Рассмотрим, например, образ строки  $C = A \parallel B$ , где  $A$  — опорный, а  $B$  — динамический списки. В этом случае в текст вписывается следующий фрагмент:

$$\Delta \delta C := 1 + \Delta \delta B;$$

В список ДЛ вписывается фрагмент:

$$\Delta nA := \Delta \Delta A[1]; \quad \Delta nC := \Delta nA; \quad \Delta nA \geq \Delta nB ? \Delta \Delta \delta C;$$

$$\Delta nC := \Delta nB; \quad \Delta \delta \delta C:$$

В список ВМБ вписывается фрагмент:

$$\text{замки проп } \Delta B \delta C (\Delta \kappa C);$$

$$\text{начало } \Delta [\Delta \alpha C] := A [\Delta \kappa C];$$

$$\Delta \alpha B := \Delta \alpha C + 1; \quad \Delta B \delta B (\Delta \kappa C); \quad \text{конец}$$

Т а б л и ц а I

Строка	Вид аргументов	Название конструкции	$\xi$ - образ первого аргумента	$\xi$ - образ второго аргумента
I	2	3	4	5
$Z = X \odot Y$	$X$ и $Y$ - опорные списки	ШИР	$\Delta \delta Z := I$	$+ I ;$
		$\mathcal{D}l$	$\Delta n X := \Delta \Delta X [I] ;$	$\Delta n Y := \Delta \Delta Y [I] ;$ $\Delta n Z := \Delta n X \times \Delta n Y ;$
		ВЫБ	замкн. проц. $\Delta B \delta Z (\Delta \kappa Z)$ ; <u>начало</u> $ENT (\Delta \kappa Z, \Delta n Z) ;$ $\Delta [\Delta \alpha Z] := X [\Delta \tau] ;$ $\Delta \tau := \Delta \tau - I ;$ $\Delta \tau := \Delta \tau \times \Delta n X ;$ $\Delta \tau := \Delta \kappa Z - \Delta \tau ;$ $\Delta \alpha Y := \Delta \alpha Z + I ;$	$\Delta [\Delta \alpha Y] := Y [\Delta \tau] ;$  <u>конец</u>
	$X$ и $Y$ - динамические списки	ШИР	$\Delta \delta Z := \Delta \delta X$	$+ \Delta \delta Y ;$
		$\mathcal{D}l$		$\Delta n Z := \Delta n X \times \Delta n Y ;$
		ВЫБ	замкн. проц. $\Delta B \delta Z (\Delta \kappa Z)$ ; <u>начало</u> $ENT (\Delta \kappa Z, \Delta n Z) ;$ $\Delta \alpha X := \Delta \alpha Z ;$ $\Delta B \delta X (\Delta \tau) ;$ $\Delta \tau := \Delta \kappa X - I ;$ $\Delta \tau := \Delta \tau \times \Delta n X ;$ $\Delta \tau := \Delta \kappa Z - \Delta \tau ;$ $\Delta \alpha Y := \Delta \alpha Z + \Delta \delta X ;$	$\Delta B \delta Y (\Delta \tau) ;$  <u>конец</u>

Т а б л и ц а I /продолжение/

1	2	3	4	5
$Z = X \cup Y$	$X$ и $Y$ - опорные списки	ШИР	$\Delta bZ := 1;$	
		ДЛ	$\Delta nX := \Delta \Delta X[1];$	$\Delta nY := \Delta \Delta Y[1];$ $\Delta nZ := \Delta nX + \Delta nY;$
		ВЫБ	<u>замкн. проц.</u> $\Delta BBZ(\Delta kZ);$ <u>начало</u> $\Delta kZ > \Delta nX ?$ да $\Delta \alpha Z;$ $\Delta[\Delta \alpha Z] := X[\Delta kZ];$ на $\Delta \Omega Z;$	$\Delta \alpha Z;$ $\Delta[\Delta \alpha Z] := Y[\Delta kZ - \Delta nX];$ $\Delta \Omega Z;$  <u>конец</u>
	$X$ и $Y$ - динамические списки	ШИР	$\Delta bZ := \Delta bX;$	$\Delta bZ \geq \Delta bY ?$ да $\Delta \omega Z;$ $\Delta bZ := \Delta bY;$ $\Delta \omega Z;$
		ДЛ		$\Delta nZ := \Delta nX + \Delta nY;$
		ВЫБ	<u>замкн. проц.</u> $\Delta BBZ(\Delta kZ);$ <u>начало</u> $\Delta kZ > \Delta nX ?$ да $\Delta \alpha Z;$ $\Delta \alpha X := \Delta \alpha Z;$ $\Delta BBX(\Delta kZ);$ на $\Delta \Omega Z;$	$\Delta \alpha Z;$ $\Delta \alpha Y := \Delta \alpha Z;$ $\Delta BBY(\Delta kZ - \Delta nX);$ $\Delta \Omega Z;$  <u>конец</u>

Т а б л и ц а I /окончание/

I	2	3	4	5
$Z = X \# Y$	X и Y - опорные списки	ШИР	$\Delta \delta Z := 1$	$+ 1;$
		Дл	$\Delta n X := \Delta \alpha X [1];$	$\Delta n Y := \Delta \alpha Y [1];$ $\Delta n Z := \Delta n X;$ $\Delta n X \geq \Delta n Y ? \text{да } \Delta \Omega Z;$ $\Delta n Z := \Delta n Y;$ $\Delta \Omega Z;$
		ВЫБ	<u>замкн. проц.</u> $\Delta B \delta Z (\Delta \kappa Z);$ <u>начало</u> $\Delta [\Delta \alpha Z] := X [\Delta \kappa Z];$ $\Delta \alpha Y := \Delta \alpha Z + 1;$	$\Delta [\Delta \alpha Y] := Y [\Delta \kappa Z];$  <u>конец</u>
	X и Y - динамические списки	ШИР	$\Delta \delta Z := \Delta \delta X$	$+ \Delta \delta Y;$
		Дл		$\Delta n Z := \Delta n X;$ $\Delta n X \geq \Delta n Y ? \text{да } \Delta \Omega Z;$ $\Delta n Z := \Delta n Y;$ $\Delta \Omega Z;$
		ВЫБ	<u>замкн. проц.</u> $\Delta B \delta Z (\Delta \kappa Z);$ <u>начало</u> $\Delta \alpha X := \Delta \alpha Z;$ $\Delta B \delta X (\Delta \kappa Z);$ $\Delta \alpha Y := \Delta \alpha Z + \Delta \delta X;$	$\Delta B \delta Y (\Delta \kappa Z);$  <u>конец</u>

Замечание о справочном слове списка. В той реализации языка  $\mathcal{E}$ , на которую мы опираемся, длина списка используется транслятором единственный раз — при размещении списка в памяти машины (в момент трансляции описания списка). После этого информация о длине теряется. Обращение к длине в тексте  $\mathcal{E}$ -программы невозможно.

В языке  $\Delta\mathcal{E}$  длина списка выполняет и другие функции. Здесь длина опорных списков систематически используется в процедурах выборки элемента сложного списка, т.е. должна сохраняться на все время выполнения  $\Delta\mathcal{E}$ -программы, и к этой длине необходимо организовать доступ. Представляется удобным иметь даже не одну характеристику длины, а две:

1) исходную длину списка (которая используется транслятором при размещении списка),

2) текущую длину списка — счетчик числа элементов, фактически накопленных в списке в данный момент (эта длина может автоматически изменяться при некоторых специальных режимах использования списков, например, в укрупненном операторе процедуры).

Кроме того, оказывается удобным иметь и некоторую другую информацию о списке. Например, о том, находится ли список в данный момент в оперативной или в магнитной памяти и т.п.

Для этих целей в языке  $\Delta\mathcal{E}$  для каждого опорного списка  $A$  вводится специальное справочное слово  $\Delta\Delta A$ , в слогах которого хранится нужная информация. Это слово строится транслятором  $\Delta\mathcal{E} \rightarrow \mathcal{E}$  для каждого списка, описанного на языке  $\mathcal{E}$ . Условимся, что значением слога  $\Delta\Delta A[i]$  является текущая длина списка.

### § 3. Организация трансляции и основные процедуры

Вся информация, которую транслятор получает, перерабатывает и выдает в качестве результата, организована в следующие списки:

список  $\Delta\mathcal{E} [ \Delta\mathcal{E} \times 9 ]$ ;

список  $\mathcal{E} [ \mathcal{E} \times 9 ]$ ;

список  $\text{ШАБ} [ \text{ШАБ} \times 9 ]$ ;



СПИСОК  $TAX[l \times TAX \times 9]$ ;  
СПИСОК  $ВЫБ[l \times ВЫБ \times 9]$ ;  
СПИСОК  $ДЛ[l \times ДЛ \times 9]$ ;  
СПИСОК  $ШИР[l \times ШИР \times 9]$ ;  
СПИСОК  $ОП[l \times 9]$ ;  
СПИСОК  $Z[l \times 9]$ ;  
СПИСОК  $X[l \times 9]$ ;  
СПИСОК  $У[l \times 9]$ ;  
СПИСОК  $пр X[l \times 1]$ ;  
СПИСОК  $пр У[l \times 1]$ ;  
СПИСОК  $ИМЯ[3 \times 45]$ ;  
СПИСОК  $\Delta[l \times 45]$ .

В списке  $\Delta \mathcal{E}$  записан текст на языке  $\Delta \mathcal{E}$ , который должен транслироваться на язык  $\mathcal{E}$ . В списке  $\Delta \mathcal{E}$  либо хранится весь исходный текст, либо этот текст частями переписывается туда из внешней памяти.

Список  $\mathcal{E}$  является результирующим, в него записывается оттранслированный текст на языке  $\mathcal{E}$ . Из списка  $\mathcal{E}$  накопленная информация периодически передается во внешнюю память.

В списке ШАБ хранятся шаблоны  $\mathcal{E}$ -образов для аргументов списочных операций. Эти  $\mathcal{E}$ -образы приведены в таблице I. Шаблоны отличаются от указанных образов тем, что условные символы  $X$ ,  $У$  и  $Z$  в именах объектов заменены специальным символом  $!$  с последующей цифрой 1, 2 или 3. Сочетание символов  $!1$  при трансляции заменяется именем списка, являющегося первым аргументом исследуемой списочной операции. Символы  $!2$  отвечают второму аргументу, а  $!3$  - результату. Специальный символ  $\star$  используется для отделения друг от друга различных фрагментов шаблона.

Рассмотрим, например, в таблице I фрагменты текста, относящиеся к образу второго опорного аргумента операции  $\otimes$ .

ШИР:  $+1$ ;

ДЛ:  $\Delta nУ := \Delta \Delta У[1]$ ;  $\Delta nZ := \Delta nX \times \Delta nУ$ ;

ВЫБ:  $\Delta[\Delta \alpha У] := У[\Delta z]$ ; конец

В списке ШАБ этому соответствует следующий текст:

конст  $\star +1$ ;  $\star$

$\Delta n!2 := \Delta \Delta !2[1]$ ;  $\Delta n!3 := \Delta n!1 \times \Delta n!2$ ;  $\star$

$\Delta[\Delta \alpha !2] := !2[\Delta z]$ ; конец  $\star$

Подобную структуру имеют тексты шаблона, относящиеся и к другим аргументам. В случае динамического первого аргумента конструктор  $\Delta l$  всегда пуста.

Вообще текст в шаблоне, относящийся к одному аргументу, имеет следующую структуру:

★ ШИР ★  $\Delta l$  ★ ВЫБ ★

Для каждой списочной операции в шаблоне имеется четыре таких текста: отдельно для первого и второго аргументов в случае опорного и динамического списков.

В таблице характеристик ТАХ для каждой списочной операции хранится справка о размещении каждого из указанных четырех фрагментов в списке ШАБ. Всего в ТАХ имеется 12 справок ( $4 \times 3$ ).

Как уже говорилось, списки ВЫБ, ДД и ШИР служат для накопления обработанных шаблонов, относящихся к одной схеме. В этих списках строится  $\mathcal{E}$ -образ описания списка-схемы.

В списки ОП, Z, X и Y поэлементно выписывается анализируемая схема. Элементы с номером K относятся к K-той строке схемы. В списке ОП хранятся операции, в списке Z — имена левых частей, в списке X — первые аргументы, в списке Y — вторые.

В списках пр X и пр Y образуются признаки соответствующих аргументов. Если значением элемента  $X[K]$  является имя динамического списка, то соответствующее значение элемента пр  $X[K]$  есть единица. В случае опорного списка имеем нулевой признак. Аналогично связаны элементы списков Y и пр Y.

В список ИМЯ выписываются компоненты очередной исследуемой строки:

ИМЯ [1] := X[K];                      ИМЯ [2] := Y[K];  
ИМЯ [3] := Z[K];

Список  $\Delta$  является стандартным рабочим полем, на котором формируется очередной элемент динамического списка. Имя  $\Delta$  фигурирует в тексте  $\mathcal{E}$ -образа описания списка-схемы.

Имеется в виду следующий порядок трансляции. Из списка произвольно считывается текст и, если он является  $\mathcal{E}$ -текстом, переносится без изменений в список  $\mathcal{E}$ . Если же встречается укрупненная конструкция собственно языка  $\Delta\mathcal{E}$ , включается одна из транслирующих процедур.

Рассмотрим две простые процедуры, транслирующие описание списка-схемы — это процедуры выписки схемы и обработки шаблонов

Условимся, что замкнутая процедура ОЧЕР считывает из списка  $\Delta \varepsilon$  очередной символ и делает его значением скаляра  $c$ . Эта же процедура следит за считыванием в список  $\Delta \varepsilon$  очередной порции текста из внешней памяти. Далее приводится описание процедуры выписки схемы, которая покомпонентно выписывает схему в списки ОП, Z, X, Y и заполняет списки пр X и пр Y. Сигналом к включению данной процедуры является описатель список-схема в  $\Delta \varepsilon$  тексте.

замкнутая процедура      ВЫПИСКА СХЕМЫ;

начало       $k := 0$

$\alpha 1:$        $k := k + 1; \text{ОЧЕР}; Z[k] := c; j := k;$

$\alpha 2:$        $j := j - 1; j = 0 ? \text{да } \Omega; c = X[j] ? \text{да } \alpha 4;$

$\alpha 3:$        $c \neq Y[j] ? \text{да } \alpha 2; \text{пр } Y[j] := 1; \text{на } \alpha 2;$

$\alpha 4:$        $\text{пр } X[j] := 1; \text{на } \alpha 3;$

$\Omega:$        $\text{ОЧЕР}; c \neq ' '? \text{да } \text{СОС};$

$\text{ОЧЕР}; X[k] := c;$

$\text{ОЧЕР}; \text{ОП}[k] := c;$

$\text{ОЧЕР}; Y[k] := c;$

$\text{ОЧЕР}; c = ', ' ? \text{да } \alpha 1;$

$c \neq ', ' ? \text{да } \text{СОС}; \text{конец}$

Процедура СОС печатает сигнал синтаксической ошибки.

После ВЫПИСКА СХЕМЫ работает процедура ОБРАБОТКИ ШАБЛОНОВ. Эта процедура последовательно строит  $\varepsilon$ -образы строк, начиная с последней.

замкнутая процедура      ОБРАБОТКА ШАБЛОНОВ;

начало

$m := 1; m1 := 1; m2 := 1;$

$\alpha 5:$        $\text{пр} := 0; \text{ИМЯ}[1] := X[k]; \text{ИМЯ}[2] := Y[k];$

$\text{ИМЯ}[3] := Z[k]; F0 := \text{ОП}[k]; F1 := \text{пр } X[k];$

$F := 1; F0 = ' \otimes ' ? \text{да } \alpha 6;$

$F := 6; F0 = ' \cup ' ? \text{да } \alpha 6;$

$F := 11; F0 \neq ' \parallel ' ? \text{да } \text{СОС};$

$\alpha 6:$        $\ell := F + F1; \ell := \text{ТАХ}[\ell];$

$\alpha 7:$        $\text{СИМ}; c = '* ' ? \text{да } \alpha 8;$

$\text{ШИР}[m2] := c; m2 := m2 + 1; \text{на } \alpha 7;$

$\alpha 8$ : СИМ;  $c = ' \star ' ?$  да  $\alpha 9$ ;  
 $\mathcal{D}l[m] := c$ ;  $m := m + 1$ ; на  $\alpha 8$ ;  
 $\alpha 9$ : СИМ;  $c = ' \star ' ?$  да  $\alpha 10$ ;  
 $ВЫБ[m] := c$ ;  $m := m + 1$ ; на  $\alpha 9$ ;  
 $\alpha 10$ :  $пр := пр + 1$ ;  $F := F + 2$ ;  $F := пр \mathcal{Y}[k]$ ;  
 $пр = 1 ?$  да  $\alpha 6$ ;  $l := ТАХ[F]$ ;  
 $пр = 2 ?$  да  $\alpha 7$ ;  $к := к - 1$ ;  
 $к \neq 0 ?$  да  $\alpha 5$ ;

конец

замкнутая процедура СИМ ;

начало  $l := l + 1$ ;  $c := ШАБ[l]$ ;  $c \neq ' ! ' ?$  да  $\Omega 1$ ;  
 $l := l + 1$ ;  $c := ШАБ[l]$   $c := ИМЯ[c]$ ;  
конец

После ОБРАБОТКИ ШАБЛОНОВ список ШИР окаймляется фрагментами

начало ... конец конст  $\Delta \alpha ! 3 = ' ! ' ;$

Список  $\mathcal{D}l$  окаймляется фрагментами

замкнутая процедура  $\Delta \mathcal{D}l ! 3$  ; начало ... конец

Затем списки ШИР, ДЛ, ВЫБ переносятся в список  $\mathcal{E}$  .

#### § 4. Пример

В качестве примера рассмотрим  $\mathcal{E}$  -образ описания списка-схемы, приведенного в начале этой работы:

список-схема  $S = \mathcal{D} \cup A$ ,

$A = \mathcal{D} \otimes \mathcal{D}$ ,

$\mathcal{D} = B \parallel C$ ;

При выполнении процедуры выписки схемы заполняются списки ОП,  $\mathcal{X}$ ,  $\mathcal{X}$ ,  $\mathcal{Y}$ , пр  $\mathcal{X}$ , пр  $\mathcal{Y}$ . Количество элементов в каждом из этих списков равно числу строк в схеме. Значения указанных списков для нашего примера приведены в таблице П.

на таблице П, в частности, видно, что опорным спискам  $B$  и  $C$  отвечают нулевые признаки, а динамическим спискам  $A$  и  $\mathcal{D}$  - единичные.

Т а б л и ц а II

Имя списка Номер элемента	оп	z	x	y	пр x	пр y
1	'U'	S	D	A	1	1
2	'⊗'	A	D	D	1	1
3	'  '	D	B	C	0	0

После выполнения процедуры обработки шаблона и заключительных операций получаем следующий образ нашей схемы:

Вычисление  $\Delta bS$  :

начало

$$\Delta bD := 1 + 1; \quad \Delta bA := \Delta bD + \Delta bD;$$

$$\Delta bS := \Delta bD; \Delta bD \geq \Delta bA? \text{ да } \Delta \omega S;$$

$$\Delta bS := \Delta bA; \Delta \omega S: \text{конец}$$

конст

$$\Delta \alpha S = '1';$$

замкн проц

$$\Delta \mathcal{D} \Delta S;$$

начало

$$\Delta nB := \Delta \Delta B[1]; \Delta nC := \Delta \Delta C[1];$$

$$\Delta nD := \Delta nB; \Delta nB \geq \Delta nC? \text{ да } \Delta \omega 1D;$$

$$\Delta nD := \Delta nC; \Delta \omega 1D;$$

$$\Delta nA := \Delta nD \times \Delta nD; \Delta nS := \Delta nD + \Delta nA;$$

конец

замкн проц

$$\Delta B5D(\Delta \kappa D);$$

начало

$$\Delta[\Delta \alpha D] := B[\Delta \kappa D]; \Delta \alpha C := \Delta \alpha D + 1;$$

$$\Delta[\Delta \alpha C] := C[\Delta \kappa D];$$

конец

замкн проц

$$\Delta B5A(\Delta \kappa A);$$

начало

$$E \nabla T(\Delta \kappa A, \Delta nD);$$

$$\Delta \alpha D := \Delta \alpha A; \Delta B5D(\Delta \tau); \Delta \tau := \Delta \kappa D - 1;$$

$$\Delta \tau := \Delta \tau \times \Delta nD; \Delta \tau := \Delta \kappa A - \Delta \tau;$$

$$\Delta \alpha D := \Delta \alpha S + \Delta bD; \Delta B5D(\Delta \tau);$$

конец

замкн проц

$$\Delta B5S(\Delta \kappa S);$$

начало

$$\Delta \kappa S > \Delta nD? \text{ да } \Delta \alpha S;$$

$$\Delta \alpha D := \Delta \alpha S; \Delta B5D(\Delta \kappa S); \text{на } \Delta \mathcal{S} \mathcal{S};$$

$\Delta\alpha S: \Delta\alpha A := \Delta\alpha S; \quad \Delta BBA(\Delta\kappa S - \Delta n D):$   
 $\Delta\Omega S: \quad \text{конец}$

Отметим, что процедура выборки элемента каждого динамического списка может использоваться самостоятельно в разных местах текста. Так, наряду с главной процедурой  $\Delta BBS$  в нашем примере могут использоваться также процедуры  $\Delta BBD$  и  $\Delta BBA$ . В каждом случае необходимо позаботиться о том, чтобы предварительно была вычислена длина нужного динамического списка (выполнена процедура  $\Delta DnS$ ). Перед самостоятельным использованием подчиненных процедур  $\Delta BBD$  или  $\Delta BBA$  необходимо указать также место выбранного элемента в списке (например, положить  $\Delta\alpha D$  или  $\Delta\alpha A$  равным единице). Для главной процедуры  $\Delta BBS$  эту функцию выполняет включенный в конструкцию ШИР оператор конст  $\Delta\alpha S = '1'$ ; размещение подчиненных списков выполняется автоматически в соответствующих процедурах выборки. В связи с этим можно заметить, что вычисление характеристик  $\Delta\beta$  и  $\Delta n$  происходит "вверх" по графу старшинства, начиная с опорных списков и кончая главным динамическим списком, который описывается данной схемой. Напротив, характеристика  $\Delta\alpha$  задается для главного списка и "спускается" по графу до опорных.

### Л и т е р а т у р а

1. Петрова Л.Т. Схемно-списочная символика в языке ДЕЛЬТА - ЭПСИЛОН. Наст. сб., стр. 65-79.
2. Катков В.Л., Поттосин И.В., Рар А.Ф., Хоперсков А.Е. ЭПСИЛОН - система автоматизации программирования задач символьной обработки. Новосибирск, "Наука", 1971.

Поступила в ред.-изд.отдел  
 2 февраля 1972 г.