

УДК 681.3.06:51

ИСЧИСЛЕНИЕ ПРОЦЕДУР В ЯЗЫКЕ ДЕЛЬТА-ЭПСИЛОН
И ВОПРОСЫ ТРАНСЛЯЦИИ

Л.Т.Петрова, Н.Н.Пивкина

Эталонное описание языка ДЕЛЬТА-ЭПСИЛОН имеется в [1]. Особенности языка являются построение в нем исчисления данных, исчисления процедур и введение укрупненного оператора процедуры, в котором объекты задаются формулами в двух названных исчислениях. С одной стороны, эти формулы позволяют алгоритмически конструировать в языке разнообразные структуры данных. С другой стороны, они позволяют описывать вычислительные процессы, конструировать структуры процедур путем согласования и объединения некоторых исходных процедур.

Принцип исчислений является основным и при разработке трансляторов с этого языка. А именно, наряду с исчислением объектов в языке, в трансляторе строится соответствующее исчисление образов этих объектов. Таким образом, весь процесс разработки транслятора разбивается на две автономные части: построение исчислений образов и собственно трансляцию. Исчисления данных и их образов построены в [2]. Здесь будет рассмотрено исчисление процедур и его трансляция. Базой для реализации введенных исчислений выбран машинно-ориентированный язык ЭПСИЛОН [3]. Сам транслятор составлен также на языке ЭПСИЛОН.

§ 1. Исчисление процедур

Исчисление процедур рассматриваемой конкретизации языка ДЕЛЬТА-ЭПСИЛОН вводится конструкцией Δ -процедура (этому отвечает процедура-схема в [1]). Элементарными процедурами исчисления в этой конкретизации являются замкнутые ЭПСИЛОН-процедуры, каждая из которых перерабатывает исходный фрагмент некоторого стандартного списка Δ в другой фрагмент того же списка. Над такими процедурами определяются операции суперпозиции (S), объединения (A), дизъюнкции (\mathcal{Q}), рекурренции (R) и соединения (\mathcal{C}) (в [1] им отвечают операции комп., объед., разв., повт. и соед.).

Δ -процедура задается формулой, которая записывается в виде правила вывода (схемы), так что описание Δ -процедуры имеет, например, такой вид:

$$\Delta\text{-процедура } И = \mathcal{Q}(B, T1, P_m), T1 = R(T2, F, P_k),$$

$$T2 = A(H, T3), T3 = \mathcal{Q}(K, L, P_n).$$

Дерево подчиненности этой схемы изображено на рис. 1

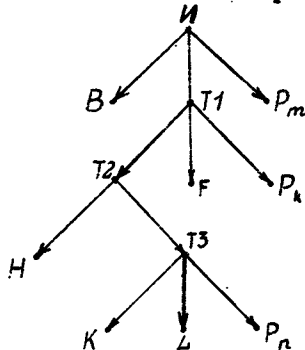


Рис. 1.

Здесь терминальные процедуры B , P_m , F , P_k , H , K , L , P_n суть замкнутые ЭПСИЛОН-процедуры. В схеме одной Δ -процедуры могут быть ссылки на другие Δ -процедуры. Например, вышеприведенное описание Δ -процедуры И можно заменить следующими двумя описаниями:

$$\Delta\text{-процедура } T2 = A(H, T3), \quad T3 = \mathcal{Q}(K, L, P_n);$$

Δ - процедура $I = \mathcal{D}(B, T_1, P_m)$, $T_1 = R(T_2, F, P_k)$,
 T_2 - процедура;

Приведем определения операций над процедурами применительно к данной конкретизации языка.

1. Суперпозиция процедур $I_3 = S(I_1, I_2)$.

Суперпозиция двух процедур I_1 и I_2 определяет собой новую процедуру I_3 , которая состоит в том, что к исходному фрагменту применяется сначала процедура I_1 , а затем к каждому результативному фрагменту I_1 применяется процедура I_2 . Совокупность всех результативных фрагментов процедуры I_2 , полученных в этом процессе, и является результатом новой процедуры I_3 .

2. Объединение процедур $I_3 = A(I_1, I_2)$.

Новая процедура I_3 состоит в том, что к исходному фрагменту применяются последовательно обе процедуры, сначала I_1 , затем I_2 . Объединение результатов этих процедур составляет результат новой процедуры I_3 .

3. Дизъюнкция процедур: $I_3 = \mathcal{D}(I_1, I_2, I_4)$.

В этой и следующей операциях в число аргументов входит специальная распознающая процедура I_4 , которая по исходному фрагменту вырабатывает признак $\Delta = 0$ или 1.

Дизъюнкция двух процедур I_1 и I_2 с распознающей процедурой I_4 определяет новую процедуру I_3 , которая при $\Delta = 0$ совпадает с процедурой I_1 , а в противном случае - с I_2 .

4. Рекуррентия процедур: $I_3 = R(I_1, I_2, I_4)$.

Здесь основной рабочей процедурой является процедура I_1 , многократно применяемая к обновляемому значению исходного фрагмента. Процедура I_2 перевычисляет исходный фрагмент.

I_4 является распознающей процедурой, при получении $\Delta_r = 0$ процесс останавливается.

Порядок выполнения процедуры I_3 следующий:

а) к исходному фрагменту применяется распознающая процедура I_4 ; при $\Delta_r = 0$ процедура I_3 считается завершенной, иначе продолжаем процесс;

в) к исходному фрагменту применяется процедура I_1 , ее результат включается в результат I_3 ;

с) процедура I_2 перевычисляет исходный фрагмент; возвращение к а).

5. Соединение процедур: $I3 = C(I1, I2)$.

Соединение двух процедур $I1$ и $I2$ определяет новую процедуру $I3$, которая состоит в поочередном применении к исходному фрагменту процедур $I1$ и $I2$. При этом результат процедуры $I3$ формируется следующим образом: каждый результативный фрагмент процедуры $I1$ соединяется с таковым же процедуры $I2$, образуя единый фрагмент.

§ 2. Исчисление образов процедур

Образы терминальных и промежуточных процедур схемы имеют единую форму, они строятся, как операторы, реализующие вычисление очередного результативного фрагмента. Эти операторы имеют одинаковую структуру. А именно, метка $\Delta \alpha T$ помечает вход в оператор, реализующий процедуру T . Каждое обращение к этой метке приводит либо к вычислению очередного результативного фрагмента и выходу через метку $\Delta \omega T$, либо - в случае исчерпания результата - к выходу через метку $\Delta \Omega T$. Предварительная настройка и согласование промежуточных выходов оператора, реализующего Δ -процедуру T , выполняется замкнутой ЭПСИЛОН-процедурой ΔCT , которая составляет вторую часть образа Δ -процедуры.

Такая организация процесса позволит ссылаться на некоторый оператор $\Delta \alpha T$ из разных схем.

В данном исчислении образ Δ -процедуры T состоит из двух конструкций: оператора с входом $\Delta \alpha T$ (и выходами $\Delta \omega T$ и $\Delta \Omega T$) и замкнутой процедуры согласования ΔCT . Эти конструкции наращиваются последовательно для каждой строки схемы, начиная с последней. В нашем примере эти конструкции строятся последовательно для процедур $T3$, $T2$, $T1$ и I .

Итак, пусть терминальная процедура T некоторой схемы описана как замкнутая ЭПСИЛОН-процедура:

замк проц T ; начало < тело процедуры > конец;

Не умаляя общности, можно считать, что терминальная процедура имеет своим результатом один фрагмент, состоящий из $\Delta \vee T$ элементов списка Δ . При этом процедура T перерабатывает исходный фрагмент $\Delta [\Delta \alpha]$, $\Delta [\Delta \alpha + 1]$, ... в результативный фрагмент $\Delta [\Delta \delta]$, $\Delta [\Delta \delta + 1]$, ..., $\Delta [\Delta \delta + \Delta \vee T - 1]$. Параметры

$\Delta\alpha$ и $\Delta\delta$ фиксируют соответственно начало аргументов и начало результатов в списке Δ для каждой процедуры. Эти параметры изменяются автоматически. Образом терминальной процедуры T в схеме является следующий оператор:

$$\begin{aligned}\Delta 1T: & \quad \Delta\alpha T = 0; \text{ на } \Delta\Omega T; \\ \Delta 2T: & \quad \text{на } \Delta 1T; \\ \Delta\alpha T: & \quad \emptyset; T; \Delta\alpha = \Delta\delta; \Delta\delta = -\Delta\alpha + \Delta\omega T; \\ & \quad \Delta\alpha T = \Delta 2T; \text{ на } \Delta\omega T; \\ \Delta\omega T: & \quad \emptyset; \\ \Delta\Omega T: & \quad \emptyset;\end{aligned}$$

Этим определены элементарные объекты исчисления образов. Здесь \emptyset означает пустой оператор, который в дальнейшем может замещаться другим (как правило, оператором перехода).

Определим операции S , A , \mathcal{D} , R и C над такими образами (операторами).

1⁰. Суперпозиция операторов:

$$ИЗ = S(И1, И2).$$

Оператор, реализующий процедуру $ИЗ$ будем называть оператором $\Delta\alpha ИЗ$. Суперпозицией операторов $\Delta\alpha И1$ и $\Delta\alpha И2$ называется следующий оператор $\Delta\alpha ИЗ$:

$$\begin{aligned}\Delta\alpha ИЗ: & \quad \text{на } \Delta\alpha И1; \\ \Delta\omega ИЗ: & \quad \emptyset; \\ \Delta\Omega ИЗ: & \quad \emptyset; \\ \Delta 1ИЗ: & \quad \text{на } \Delta\alpha И2; \\ \Delta 2ИЗ: & \quad \text{на } \Delta\alpha И1; \\ \Delta 3ИЗ: & \quad \Delta\alpha ИЗ = \Delta 1ИЗ; \quad \text{на } \Delta\alpha И2; \\ \Delta 4ИЗ: & \quad \Delta\alpha ИЗ = \Delta 2ИЗ; \quad \text{на } \Delta\Omega ИЗ; \\ \Delta 1ИЗ: & \quad \text{на } \Delta 2ИЗ; \\ \Delta 2ИЗ: & \quad \text{на } \Delta 3ИЗ; \\ \Delta 3ИЗ: & \quad \text{на } \Delta\omega ИЗ;\end{aligned}$$

Здесь метками с начальными символами Δz помечены заготовки операторов управления, при помощи которых процедура согласования настраивает промежуточные выходы $\Delta \omega$ и $\Delta \Omega$.

Выходы операторов, реализующих процедуры $И1$ и $И2$, настраиваются по следующему правилу:

$$\Delta \omega И1: -\Delta z1И3;$$

$$\Delta \Omega И1: -\Delta z2И3;$$

$$\Delta \omega И2: -\Delta z3И3;$$

$$\Delta \Omega И2: -\Delta z4И3;$$

Этот фрагмент отвечает процедуре $И3$ и вписывается в общую процедуру согласования.

2°. Объединение операторов:

$$И3 = A(И1, И2).$$

Оператор, реализующий процедуру $И3$, имеет вид:

$$\Delta \omega И3: \emptyset;$$

$$\Delta \Omega И3: -\Delta \alpha; \Delta 1И3: -\Delta \varepsilon; \text{ на } \Delta \omega И1;$$

$$\Delta \omega И3: \emptyset; \Delta \Omega И3: \emptyset;$$

$$\Delta 2И3: \text{ на } \Delta \omega И2;$$

$$\Delta 5И3: \Delta \alpha: -\Delta \Omega И3; \Delta \varepsilon: -\Delta 1И3;$$

$$\Delta \omega И3: -\Delta 2И3; \text{ на } \Delta \omega И2;$$

$$\Delta 3И3: \text{ на } \Delta \omega И1;$$

$$\Delta 4И3: \Delta \omega И3: -\Delta 3И3; \text{ на } \Delta \omega И3;$$

$$\Delta 6И3: \Delta \omega И3: -0; \text{ на } \Delta \Omega И3;$$

$$\Delta z1И3: \text{ на } \Delta \omega И3; \Delta z2И3: -\text{ на } \Delta 5И3;$$

$$\Delta z3И3: \text{ на } \Delta \omega И3; \Delta z4И3: -\text{ на } \Delta 6И3;$$

Выходы операторов, реализующих процедуры $И1$ и $И2$ настраиваются по следующему правилу:

$$\Delta \omega И1: -\Delta z1И3;$$

$$\Delta \Omega И1: -\Delta z2И3;$$

$$\Delta \Omega И2 := \Delta \tau4 И3;$$

$$\Delta \omega И2 := \Delta \tau3 И3;$$

3°. Дивергенция операторов:

$$И3 = \mathcal{D}(И1, И2, И4).$$

Процедуру $И3$ реализует следующий оператор:

$$\Delta \alpha И3: \emptyset; \quad \Delta \Omega И3 := \Delta \alpha; \quad \Delta 1 И3 := \Delta \delta;$$

$$\text{на } \Delta \alpha И4;$$

$$\Delta \omega И3: \emptyset;$$

$$\Delta \Omega И3: \emptyset;$$

$$\Delta 2 И3: \text{на } \Delta \alpha И1;$$

$$\Delta 3 И3: \text{на } \Delta \alpha И2;$$

$$\Delta 4 И3: \Delta \alpha И3 := \Delta 3 И3; \quad \text{на } \Delta \alpha И2;$$

$$\Delta 5 И3: \Delta \alpha := \Delta \Omega И3; \quad \Delta \delta := \Delta 1 И3;$$

$$\Delta \rho \neq 0? \quad \text{на } \Delta 4 И3; \quad \Delta \alpha И3 := \Delta 2 И3;$$

$$\text{на } \Delta \alpha И1;$$

$$\Delta 6 И3: \Delta \alpha И3 := 0; \quad \text{на } \Delta \Omega И3;$$

$$\Delta \tau1 И3: \text{на } \Delta \omega И3;$$

$$\Delta \tau2 И3: \text{на } \Delta 6 И3;$$

$$\Delta \tau3 И3: \text{на } \Delta \alpha И4;$$

$$\Delta \tau4 И3: \text{на } \Delta 5 И3;$$

Выходы операторов, реализующих процедуры $И1$, $И2$ и $И4$, настраиваются по следующему правилу:

$$\Delta \omega И1 := \Delta \tau1 И3;$$

$$\Delta \Omega И1 := \Delta \tau2 И3;$$

$$\Delta \omega И2 := \Delta \tau1 И3;$$

$$\Delta \Omega И2 := \Delta \tau2 И3;$$

$$\Delta \omega И4 := \Delta \tau3 И3;$$

$$\Delta \Omega И4 := \Delta \tau4 И3;$$

4°. Рекуррентия операторов:

$$ИЗ - R(И1, И2, И4).$$

Процедуру ИЗ реализует следующий оператор:

$$\begin{aligned} \Delta \alpha ИЗ: & \emptyset; \Delta \delta 1: -\Delta \delta; \Delta \Pi; \Delta 0ИЗ: -\Delta \alpha; \\ & \Delta 1ИЗ: -\Delta \delta; \text{на } \Delta \alpha И4; \\ \Delta 3ИЗ: & \Delta \alpha ИЗ: -0; \Delta \neq 0? \text{ на } \Delta \Omega ИЗ; \\ & \Delta \alpha: -\Delta 0ИЗ; \Delta \delta: -\Delta 1ИЗ; \text{на } \Delta \alpha И1; \\ \Delta 4ИЗ: & \Delta \alpha: -\Delta 0ИЗ; \Delta \delta: -\Delta 1ИЗ; \text{на } \Delta \alpha И2; \\ \Delta 5ИЗ: & \text{на } \Delta \alpha И1; \\ \Delta 6ИЗ: & \Delta \alpha ИЗ: -\Delta 5ИЗ; \text{на } \Delta \omega ИЗ; \\ \Delta \tau 1ИЗ: & \text{на } \Delta 6ИЗ; \\ \Delta \tau 2ИЗ: & \text{на } \Delta 4ИЗ; \\ \Delta \tau 3ИЗ: & \text{на } \Delta \alpha И4; \\ \Delta \tau 4ИЗ: & \text{на } \Delta 3ИЗ; \end{aligned}$$

Выходы операторов, реализующих процедуры И1, И2 и И4, настраиваются по следующему правилу:

$$\begin{aligned} \Delta \omega И1 &:= \Delta \tau 1ИЗ; \\ \Delta \Omega И1 &:= \Delta \tau 2ИЗ; \\ \Delta \omega И2 &:= \Delta \tau 2ИЗ; \\ \Delta \Omega И2 &:= \Delta \tau 3ИЗ; \\ \Delta \omega И4 &:= \Delta \tau 3ИЗ; \\ \Delta \Omega И4 &:= \Delta \tau 4ИЗ; \end{aligned}$$

5°. Соединение операторов:

$$ИЗ - C(И1, И2).$$

Процедуру ИЗ реализует следующий оператор:

$$\begin{aligned} \Delta \alpha ИЗ: & \emptyset; \Delta 0ИЗ: -\Delta \alpha; \Delta 1ИЗ: -\Delta \delta; \text{на } \Delta \alpha И1; \\ \Delta \omega ИЗ: & \emptyset; \end{aligned}$$

$\Delta\Omega И3: \emptyset;$
 $\Delta10И3: \underline{\text{на}} \Delta8И3;$
 $\Delta4И3: \Delta\alpha И3:-0; \quad \underline{\text{на}} \Delta\Omega И3;$
 $\Delta5И3: \Delta2И3:-\Delta\alpha; \quad \Delta3И3:-\Delta\delta;$
 $\Delta6И3: \Delta\alpha:-\Delta0И3; \quad \underline{\text{на}} \Delta\alpha И2;$
 $\Delta7И3: \Delta\alpha И3:-\Delta10И3; \quad \Delta\delta1:-\Delta3И3;$
 $\Delta П; \Delta\alpha:-\Delta2И3; \quad \underline{\text{на}} \Delta\omega И3;$
 $\Delta8И3: \Delta\delta:-\Delta3И3; \quad \underline{\text{на}} \Delta6И3;$
 $\Delta9И3: \Delta\alpha:-\Delta0И3; \quad \Delta\delta:-\Delta1И3;$
 $\underline{\text{на}} \Delta\alpha И1;$

Выходы операторов, реализующих процедуры И1 и И2 , настраиваются по следующему правилу:

$\Delta\omega И1:-\Delta\alpha И3;$
 $\Delta\Omega И1:-\Delta\alpha И3;$
 $\Delta\omega И2:-\Delta\alpha И3;$
 $\Delta\Omega И2:-\Delta\alpha И3;$

Процедура $\Delta П$, фигурирующая в определении операций C и R , реализует перенос данных, приведем ее описание:

замк проц $\Delta П$; начало $\Delta1:-\Delta\delta1; \quad \Delta2:-\Delta\alpha$;
 $\Delta П0:\Delta[\Delta1]:-\Delta[\Delta2]; \Delta1:-\Delta1+1; \Delta2:-\Delta2+1;$
 $\Delta2<\Delta\delta? \underline{\text{да}} \Delta П0; \Delta\alpha:-\Delta\delta1; \Delta\delta:-\Delta1; \underline{\text{конец}};$

§ 3. Организация трансляции и основные процедуры

3.1. Основные списки

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

Списки

$\Delta\epsilon [\epsilon\Delta\epsilon \times g],$

$\epsilon [\epsilon\epsilon \times g],$

$\Delta\alpha\beta [\epsilon\Delta\alpha\beta \times g],$

$TAX [eTAX \times 15],$
 $OP [e \times 9],$
 $Z [e \times 45],$
 $x [e \times 45],$
 $y [e \times 45],$
 $D [e \times 45],$
 $np x [e \times 1],$
 $np y [e \times 1],$
 $ИМЯ [ИМЯ \times 45],$
 $ШИР [eШИР \times 9],$
 $\mathcal{O} [1 \times 9],$
 $\mathcal{OO} [1 \times 45],$
 $\Phi [e\Phi \times 45],$
 $ПЕЧ [eПЕЧ \times 45].$

В списке $\Delta \epsilon$ записан текст на языке ДЕЛЬТА-ЭПСИЛОН, который должен транслироваться на языке ЭПСИЛОН. Исходный текст переписывается в список $\Delta \epsilon$ частями из внешней памяти.

В список ϵ записывается оттранслированный текст на языке ЭПСИЛОН. Из этого списка накопленная информация периодически передается во внешнюю память.

В списке \mathcal{WAB} хранятся шаблоны ϵ -образов укрупненных конструкций языка. Эти ϵ -образы приведены в § 2. Специальный символ $*$ используется для отделения друг от друга различных фрагментов шаблона. Текст в шаблоне, относящийся к образу

Δ -процедуры, имеет следующую структуру:

$$\Delta \alpha ИЗ(\beta) * \Delta СИЗ(\beta) * \alpha,$$

где β - соответствующая операция над процедурами.

Список TAX является оглавлением шаблона \mathcal{WAB} . Элемент $TAX [k]$ относится к k -му фрагменту шаблона и определяет порядковый номер первого символа этого фрагмента в списке \mathcal{WAB} .

В списки *ОП*, *Z*, *X*, *Y*, *P* выписывается в инверсном виде схема транслируемой Δ -конструкции: в списке *ОП* хранятся операции всех строк схемы, в списке *Z* - имена результатов, в *X* - имена первых аргументов, в *Y* - вторых и в *P* - третьих. Списки *пр X* и *пр Y* суть шкалы и содержат некоторые признаки, вычисленные соответственно для первых и вторых аргументов выписанной схемы.

В список *ИМЯ* выписываются компоненты очередной исследуемой строки.

Список *ШИР* служит для предварительного накопления параллельно создаваемых частей образа.

Списки *Q*, *QQ* - фиктивные, заголовкам которых в ходе трансляции присваиваются различные значения. Списки *Ф* и *ПЕЧ* - некоторые рабочие списки.

3.2. Структура транслятора

Из списка $\Delta\epsilon$ посимвольно считывается текст и, если он является ϵ -текстом, то переносится без изменений в список $\Delta\epsilon$. Если же встречается укрупненная конструкция собственно языка ДЕЛЬТА-ЭПСИЛОН, включается одна из транслирующих процедур.

Рассмотрим схематично процесс трансляции конструкции Δ - процедура. В блок-схеме на рис. 2 указаны связи процедур при трансляции описания Δ - процедуры.

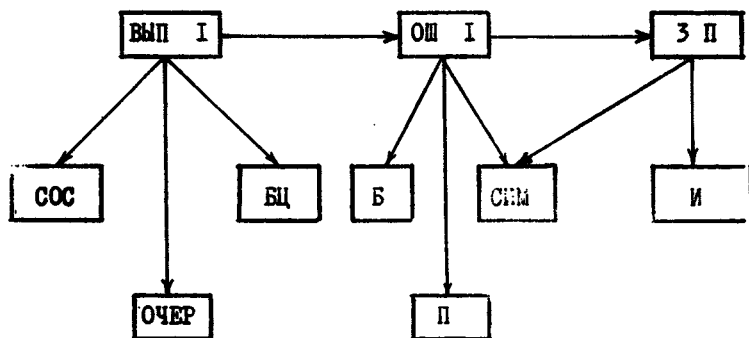


Рис. 2.

Процедура ВМП I выписывает в инверсном виде схему Δ - процедуры в списки OP, Z, X, Y, P , а в шкалы $пр X$ и $пр Y$ заносятся единичные признаки для нетерминальных аргументов.

Процедура ОШ I исчисляет программу ε -образа описания Δ -процедуры. Сначала выписывается в рабочую ячейку первый из терминальных аргументов в данной строке и обрабатываются соответствующие фрагменты шаблона, затем для каждого следующего терминального аргумента строки делается то же самое. После обработки шаблонов всех терминальных аргументов строки процедура ОШ I выписывает в список ИИИ всю строку схемы из списков OP, Z, X, Y, P и обрабатывает фрагменты шаблона, соответствующие этой строке. При этом фрагмент конструкции $\Delta\alpha$ размещается прямо в тексте списка ε , а фрагмент ΔC временно хранится в списке ШИР.

Процедура З П переносит накопленный текст из списка ШИР в список ε .

Процедура СОС печатает фрагмент $\Delta\varepsilon$ -текста в случае обнаружения в нем синтаксической ошибки.

Процедура Б выписывает в список Φ имена процедур, обработанных транслятором.

Процедура ОЧЕР выдает очередной символ исходного текста из списка $\Delta\varepsilon$ и в случае исчерпания этого списка вводит в него очередную порцию информации с перфокарт.

Процедура БЦ выделяет из $\Delta\varepsilon$ текста имя очередного объекта.

Процедура СИМ выдает очередной символ текста из шаблона ШАБ, заменяя служебные символы на конкретные имена.

Процедура И выписывает в ε -текст имя очередного объекта.

Процедура П выдает во внешнюю память (на печать и на перфокарты) накопленную информацию из списка ε .

3.3. Процедуры транслятора

В процедура ОЧЕР ;

начало

- индекс очередного символа.

$\alpha 36: og := og + 1; c := \Delta\varepsilon[og];$ Выборка из списка $\Delta\varepsilon$ очередного символа.

$c = OP$ да $\alpha 36;$ Пропуск пустого символа.

$C \neq '*' ?$ да ω ;

Проверка на конец очередной порции $\Delta \varepsilon$ -текста.

$oz := oz + 1$;

$\Delta \varepsilon [oz] = '*' ?$ да $\Delta P4$;

$\Delta \varepsilon$ -текст исчерпан. Выход из процедуры.

$\Delta 64: (10, 0, \Delta 64, 0); oz := 0$;
на $\Delta 36$; ω ;

Ввод очередной порции $\Delta \varepsilon$ -текста.

конец;

в процедура СИМ ;

начало

$\Delta 38: l := l + 1; c := \text{ШАБ}[l]$; Выписка символа из списка ШАБ.

$c = 0 ?$ да $\Delta 38$; Пропуск пустого символа.

$c \neq 'U' ?$ да $\Omega 1$; Нужна ли замена символа?

$l := l + 1; c := \text{ШАБ}[l]$; Замена служебного символа на конкретное имя
 $c := \text{ИМЯ}[c]$;

$\Omega 1$:

конец;

в процедура ВЫП I.

начало $k = 0$;

k - счетчик очередной строки схемы.

$\Delta 12: k := k + 1$; БЦ (Z) ;

Имя результата заносится в список Z .

$j := k$; $P[k] := 0$;

нр $X[k] := 0$; нр $Y[k] := 0$;

j - счетчик строк.

$\Delta 13: j := j + 1$; $j = 0 ?$ да $\Omega 2$;

Заполнение
шкал:

$Z[k] = X[j] ?$ да $\Delta 15$;

$\Delta 14: Z[k] \neq Y[j] ?$ да $\Delta 13$;

нр $Y[j] := 1$; на $\Delta 13$;

Процедура $Y[j]$ не терминальная.

$\Delta 15: \text{нр } X[j] := 1$; на $\Delta 14$;

Процедура $X[j]$ не терминальная.

$\Omega 2: c \neq '=' ?$ да СОС;

Ошибка в схеме.

ОЧЕР; ОП $[k] := c$; ОЧЕР;

$c \neq '(' ?$ да СОС;

Ошибка в схеме.

БЦ (X) ;

Запись имени I аргумента в список X .

$c \neq ' '? \text{ да } \text{СОС};$

БЦ (У);
 $c = ' '? \text{ да } \angle \text{I6};$

$c \neq ' '? \text{ да } \text{СОС};$
БЦ (Р);

$c \neq ' '? \text{ да } \text{СОС};$

$\angle \text{I6: ОЧЕР}; \quad c = ' '? \text{ да } \angle \text{I2};$

$c \neq ' '? \text{ да } \text{СОС};$

Конец:

замки процедуры ОИ I;

Начало $m2 := 0;$

ИМЯ [3] := Z [1]; $nr := 0;$

$\ell := \text{TAX} [33]; \text{ на } \angle 22;$
 $\angle \text{I7: } nr \cdot X[k] \neq 0? \text{ да } \angle \text{I8};$

ИМЯ [1] := X[k];
 $nr := 1;$
 $F := 21; \text{ Б};$

$\angle \text{I8: } nr \cdot Y[k] \neq 0? \text{ да } \angle \text{I9};$
ИМЯ [1] := Y[k];
 $nr := 2; F := 21;$
 $\text{Б};$

$\angle \text{I9: } P[k] = 0? \text{ да } \angle 20;$
ИМЯ [1] := P[k];

$nr := 3; F := 21; \text{ Б};$
 $\angle 20:$

ИМЯ [1] := X[k];
ИМЯ [2] := Y[k]; $nr := 4;$

ИМЯ [3] := Z[k]; $F := 16;$
ИМЯ [4] := P[k]; $FO := \text{OIT}[k];$
 $FO = 'A'? \text{ да } \angle 21;$

Ошибка в схеме.

Запись имени П аргумента в список У.

Ошибка в схеме.

Запись имени III аргумента в список Р.

Ошибка в схеме.

Переход к выписке следующей строки схемы.

Ошибка в схеме.

Подготовка к выписке из ИАБ:

фрагмента 33 (начало процедуры $\Delta \in U3$),

образа для I аргумента строки, если аргумент терминальный ($nr \cdot X[k] = 0$),

образа для II аргумента строки, если аргумент терминальный ($nr \cdot Y[k] = 0$),

образа для III аргумента строки.

Заполнение списка ИМЯ именами процедур из очередной строки схемы.

Выбор образа для очередной операции.

$F := F + 1;$ $FO = 'C'?$ да $\Delta 21;$
 $F := F + 1;$ $FO = 'S'?$ да $\Delta 21;$
 $F := F + 1;$ $FO = 'D'?$ да $\Delta 21;$
 $F := F + 1;$ $FO \neq 'R'?$ да $\Delta 21;$
 $\Delta 21:$ $\ell := TAX[F];$
 $\Delta 22:$ СИМ; $c = 'x'?$ да $\Delta 23;$
 $И(\epsilon, \alpha); \alpha := \gamma;$ на $\Delta 22;$
 $\Delta 23:$ СИМ; $c = 'x'?$ да $\Delta 24;$
 $И(ШИР, m2); m2 := \gamma;$
 на $\Delta 23;$
 $\Delta 24:$ $np = 0?$ да $\Delta 17;$
 $np = 1?$ да $\Delta 18;$ $np = 2?$
 да $\Delta 19;$ $np = 3?$ да $\Delta 20;$
 $\alpha < 2000?$ да $\Delta 74;$
 П;
 $\Delta 74:$ $k := k - 1;$ $k \neq 0?$ да $\Delta 17;$
 $\alpha := \alpha + 1;$ $\epsilon[\alpha] := 'конеч';$
 $\alpha := \alpha + 1;$ $\epsilon[\alpha] := ' ';$

конеч;

в процедура $БЦ(\varphi);$

начало $DD := \varphi;$

очер; $DD[k] := c;$

$\Delta 41:$ очер; $cc = '8'?$ да $\Delta 43;$

$z := DD[k] \leftarrow g; DD[k] := z + c;$

на $\Delta 41;$

$\Delta 43:$

конеч;

в процедура $Б;$

начало $\varphi 1 := \varphi;$

Ошибка в схеме.

Запись в список ϵ обработанного текста из ШАБ (оператор $\Delta \Delta ИЗ$).

Запись в список ШИР обработанного текста из ШАБ (процедура $\Delta СИЗ$).

Проверка: обработана ли очередная строка полностью.

Исчерпан ли список ϵ ? Если да, то переход на процедуру П.

Переход к очередной строке.

Формирование конца записи процедуры $\Delta \epsilon ИЗ$.

Пересылка заголовка фактического списка в заголовок фиктивного списка.

Очередной символ не относится к имени.

Запись очередного символа имени.

φ - число процедур, записанных в список φ .

$\psi 2:$ $\psi 1 = 0?$ да $\psi 3;$
 $\varphi[\psi 1] = \text{ИМЯ}[1]?$ да $\psi 4;$

Встречалась ли данная процедура раньше?

$\psi 1 := \psi 1 - 1;$ на $\psi 2;$

$\psi 3:$ $\psi := \psi + 1;$ $\varphi[\psi] := \text{ИМЯ}[1];$

Запись имени процедуры в φ .

на $\alpha 21;$

Нестандартный выход из процедуры.

$\psi 4:$

конец;

в процедура $3 \text{ П } (b1, p);$

Пересылка фактического заголовка списка в заголовок фиктивного списка.

начало $\mathcal{Q} := b1;$

$\alpha 51:$ сим; $c = 'x'?$ да $\alpha 52;$

$I(\epsilon, \alpha); \alpha := y;$ на $\alpha 51;$

Выписка из ЛАБ в список ϵ .

$\alpha 52:$ $\ell := 0;$

$\alpha 53:$ $b > \beta?$ да $\alpha 54;$ $\ell := \ell + 1;$

Выписка текста в список ϵ .

$\alpha := \alpha + 1;$ $e[\alpha] := \mathcal{Q}[\ell];$

на $\alpha 53;$

$\alpha 54:$

конец;

в процедура $\text{И } (\psi, \gamma);$

начало $d := \psi;$

Пересылка заголовка списка из ψ в заголовок фиктивного списка.

$c_\alpha := c;$ $z := 1;$

$\alpha 67:$ $c_\alpha[z] = 0?$ да $\alpha 68;$

$y := y + 1;$ $d[y] := c_\alpha[z];$

Запись очередного символа.

$\alpha 68:$ $z = 3?$ да $\alpha 78;$

Проверка на конец;

$z := z + 1;$ на $\alpha 67;$

Подготовка к проверке следующего символа.

$\alpha 78:$

конец;

в процедура $\text{СОС};$

начало на $\alpha 37;$

$\alpha 42:$ $(500, 0, 0, 3);$

Подготовка текста печати.

$\alpha 40: (500, 0, 0, 0); (112, 12, \alpha 40, 1);$
на $\alpha 44;$
 $\alpha 37: b := \uparrow \Delta \varepsilon [or]; b := b - 5;$
 ПЕЧ [2] := or;
 ПЕЧ [3] := $\Delta \varepsilon [or];$
 $(14, 130, b, b); (13, \alpha 42, b, \alpha 40);$
 $(32, 0, \alpha 40, 0);$
 $\alpha 44: (50, 500, 0, 15);$ Печать подготовлен-
 ной информации.
 $\alpha 31: (70, 0, 0, 0);$
 $\alpha 45: c = '?'; ? \text{ да } \Omega 5; or := or + 1;$ Пропуск символов из
 списка $\Delta \varepsilon$ до
 символа '?'.
 $c := \Delta \varepsilon [or];$
на $\alpha 45;$
 $\Omega 5:$ на $\alpha 0;$ Выход из процедуры.
в процедуру П ;
начало на $\alpha 72;$
 $\alpha 57: (0, 0, 0, 0);$ Печать текста, по-
 $\alpha 58: (70, 0, 0, 0);$ лученного в списке
 $\alpha 59: (0, 0, 0, 0)$ ε .
 Выдача полученного
 текста на перфо-
 карты.
 $\alpha 60: (70, 0, \alpha 59, 0);$
на $\alpha 73;$
 $\alpha 72: m1 := \uparrow \varepsilon [\alpha]; (13, \alpha 61, m1, \alpha 57);$ Подготовка к печати
 $(13, \alpha 66, m1, \alpha 59); \alpha := \alpha + 1;$ и перфорации. Здесь:
 $\varepsilon [\alpha] := 0; \alpha := \alpha + 1; \varepsilon [\alpha] := 0;$ $\alpha 61: (50, 500, 0, 13);$
 $\alpha := \alpha + 1; \varepsilon [\alpha] := 0; \alpha := \alpha + 1;$ $\alpha 65: (100, 0, 0, 1);$
 $\varepsilon [\alpha] := 0; \alpha := 0;$ $\alpha 66: (50, 200, 0, 13);$
 $(52, 0, 0, 0); (13, \alpha 65, m1, \alpha 69);$ Чистка последней
 $\alpha 69: (0, 0, 0, 0); (112, 12, \alpha 69, 1);$ перфокарты, в кото-
 рой стоит код суммы.
на $\alpha 57;$
 $\alpha 73:$
конец;

Л и т е р а т у р а

1. ПЕТРОВА Л.Т. Схемно-списочная символика в языке ДЕЛЬТА-ЭПСИЛОН. - В кн.: Оптимизация, вып. 6 (23), Новосибирск, 1972, с. 65-79.
2. ПЕТРОВА Л.Т. Трансляция укрупненных конструкций языка ДЕЛЬТА-ЭПСИЛОН. Там же, с. 80-93.
3. КАТКОВ В.Л., МОРОЗОВ В.П., ПОТТОСИН И.В., РАР А.Ф., СЕМЕНОВА Л.Я., ХОПЕРСКОВ А.Е. ЭПСИЛОН-система автоматизация программирования задач символьной обработки. Новосибирск, "Наука", 1972.

Поступила в ред.-изд. отд.

17. XII. 1974 г.