

АКАДЕМИЯ НАУК ЭСТОНСКОЙ ССР  
ИНСТИТУТ КИБЕРНЕТИКИ

На правах рукописи

ПЕИЯЛ Янус Раймодович

УДК 681.3.06

СПЕЦИФИКАЦИИ ФОРТ-ПРОГРАММ И ИХ ПРИМЕНЕНИЕ  
В СИСТЕМАХ ПОСТРОЕНИЯ ТРАНСЛЯТОРОВ

05.13.11 - математическое и программное обеспечение  
вычислительных машин и систем

Автореферат диссертации на соискание  
ученой степени кандидата технических наук

Таллин - 1986

Работа выполнена в Тартуском ордена Трудового Красного Знамени и ордена Дружбы народов государственном университете

Научный руководитель - кандидат физико-математических наук,  
профессор КААЗИК Ю.Н.

Официальные оппоненты - доктор физико-математических наук,  
СЕМЕНОВ А.Л.

- кандидат технических наук,  
ВООГЛАЙД А.О.

Ведущая организация - Ленинградский ордена Ленина и ордена  
Трудового Красного Знамени государственный  
университет имени А.А.Жданова

Защита состоится "....".....1987 г. в ..... часов  
на заседании специализированного Совета К 017.03.01 при  
Институте кибернетики АН ЭССР по адресу: 200108, ЭССР,  
г. Таллин, Академия тез, 21.

С диссертацией можно ознакомиться в библиотеке Института  
кибернетики АН ЭССР.

Автореферат разослан "....".....198.. г.

Ученый секретарь специализированного совета,  
кандидат технических наук МЫТУС Л.Л.



.....

## ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность работы. Развитие вычислительной техники и расширение области её применения требуют повышения производительности труда в программировании. Одним средством решения этой задачи является разработка и внедрение инструментальных систем и систем построения трансляторов (СПТ).

На данном этапе актуальна проблема создания подобных систем для персональных ЭВМ, т.к. именно персональные ЭВМ применяются в самых разных областях человеческой деятельности. В то же время ресурсы широко распространенных микро-ЭВМ (объем оперативной памяти, быстродействие и т.п.) обычно весьма ограничены. Поэтому непосредственное использование тех систем и методов, которые разработаны для больших и средних ЭВМ, затруднено, но тем не менее нужно учитывать накопленный в этой области опыт.

В условиях быстрого развития микропроцессорной техники нецелесообразно разрабатывать программное обеспечение заново для каждого нового типа ЭВМ. Таким образом, переносимость программного обеспечения является актуальным требованием.

В практике последних лет средством создания мобильного и компактного программного обеспечения микро-ЭВМ (в том числе для разработки СПТ) служили расширяемые языки типа ФОРТ.

Актуальной задачей является изучение и совершенствование этого инструмента в целях повышения надежности программного обеспечения. Необходимо обратить внимание на улучшение методов описания и реализации языков программирования в ФОРТ-ориентированных СПТ на основе имеющегося опыта.

Цель работы. Основными целями данной диссертационной работы являются:

1. Исследование и разработка методов описания языков программирования в ФОРТ-ориентированных СПТ.
2. Разработка методов, позволяющих повысить надежность системы программирования ФОРТ и созданных на её основе СПТ.

Методы исследования. В работе используются методы теории формальных языков и методы теории полугрупп, а также некоторые положения технологии программирования.

Научная новизна.

1. Определено понятие спецификации ФОРТ-программы и разработан новый формальный аппарат - алгебра спецификаций языка ФОРТ. Изучены свойства этой алгебры.
2. На основе алгебры спецификаций определено понятие корректной ФОРТ-программы и предложено решение проблемы статического контроля типов для языка ФОРТ.
3. Проанализированы существующие методы описания языков в ФОРТ-ориентированных СПТ и введено определение корректности таких описаний.
4. Разработаны методы, позволяющие проверить корректность описания исходного языка относительно вышеуказанного определения и выявить некоторые ошибки в этом описании на самых ранних этапах реализации языка.

Практическая ценность и реализация результатов. Предложенные в работе методы позволяют повысить надежность системы программирования ФОРТ и созданных на её основе СПТ. Пользователям ФОРТ-ориентированной СПТ предлагаются средства для создания и отладки описаний реализуемых языков.

Все полученные результаты применяются в СПТ ТАРТУ, но статический контроль типов для языка ФОРТ можно использовать и отдельно.

СПТ ТАРТУ реализована на языке fig-FORTH и внедрена на ЭВМ СМ-4, Apple -II и Искра-226.

Личное участие автора. СПТ ТАРТУ реализована коллективом разработчиков (Томбак М.О., Соо В.К., Пейял Я.Р.).

Все теоретические результаты, изложенные в главах 2 и 3 диссертационной работы и не имеющие ссылки на литературу, получены автором лично. Программы, которые приводятся в приложениях диссертации, тоже разработаны автором.

Апробация работы. Основные результаты исследований по теме диссертации докладывались на:

- Всесоюзной конференции по методам трансляции и конструирования программ, Новосибирск, октябрь, 1984,
- Всесоюзной конференции "Диалог" - 84 - микро", Ленинград, ноябрь, 1984,
- III Всесоюзной конференции "Автоматизация производства систем программирования", Таллин, декабрь, 1986,
- Всесоюзном семинаре РГ по реализации языков программирования, Тарту, февраль, 1984,
- Всесоюзном семинаре по языку ФОРТ, Таллин, апрель, 1984,
- Всесоюзном семинаре "Программное обеспечение мини- и микро-ЭВМ семейства СМ ЭВМ", Москва, октябрь, 1984,
- Всесоюзном семинаре по языку ФОРТ, Миасс, февраль, 1985,
- Всесоюзном семинаре по языку ФОРТ, Ленинград, окт., 1985,
- Всесоюзном семинаре по языку ФОРТ, Тарту, май, 1986,
- Всесоюзном семинаре "Расширяемые средства программирования", Кишинев, июнь, 1986,
- семинаре ВЦ Таллинского политехнического ин-та, 1983,
- семинарах кафедры программирования Тартуского госуниверситета, 1983, 1984, 1985, 1986,
- семинаре кафедры алгебры и геометрии Тартуского госуниверситета, февраль, 1986,
- семинаре сектора программного обеспечения Института кибернетики АН ЭССР, октябрь, 1986.

Публикации. По теме диссертации опубликовано 5 работ (2 из них в соавторстве).

Структура и объем работы. Диссертационная работа состоит из введения, трех глав, заключения, списка литературы (64 наименования) и двух приложений. Работа изложена на 92 страницах машинописного текста, в том числе 2 рисунка.

## СОДЕРЖАНИЕ РАБОТЫ

Во введении обосновывается актуальность темы исследования, конкретизируется круг рассматриваемых вопросов, определяются цели и задачи исследования. Здесь кратко изложены полученные результаты и содержание глав диссертации.

В первой главе рассматриваются инструментальные системы программирования на базе языков типа ФОРТ. Дается характеристика таких языков как инструмента реализации компактного и мобильного программного обеспечения. Анализируются недостатки языка ФОРТ и недостатки существующих ФОРТ-ориентированных СПТ. Описывается СПТ TARTU, в контексте которой применяются основные результаты данной диссертации. В результате анализа уточняются конкретные задачи исследования:

В разделе I.1 обсуждаются некоторые вопросы, связанные с областью автоматизированной реализации языков программирования. Рассматриваются мобильность программного обеспечения, выбор промежуточного языка в трансляторах, создание набора абстракций в виде хорошо структурированной совокупности модулей и т.п. Приводятся примеры СПТ, ориентированных на конкретный язык высокого уровня. Делается вывод, что язык ФОРТ является перспективным инструментом при реализации языков программирования.

В разделе I.2 дается обзор материалов по языкам типа ФОРТ, рассматриваются вопросы реализации систем на основе "сшитого кода", области применения подобных систем и пути их развития, в том числе создание т.н. ФОРТ-машин. Описываются основные концепции и свойства наиболее характерного представителя семейства языков "сшитого кода" - языка ФОРТ. Выделяются такие качества как расширяемость, открытость, гибкость, мобильность, интерактивность, структурность, лаконичность и эффективность. Характерной чертой языка ФОРТ является использование магазинов (стеков) и постфиксной записи.

Расширяемость языка ФОРТ позволяет настраивать язык на

нужную проблемную область и создавать понятия достаточно высокого уровня. Поэтому актуальной задачей будущего является выявление базовых понятий для конкретных проблемных областей и их реализация, вплоть до создания проблемно-ориентированных виртуальных ФОРТ-машин.

Обзор завершается перечислением наиболее существенных недостатков языка ФОРТ, среди которых основным считается отсутствие средств контроля типов передаваемых через магазин параметров.

Раздел I.3 посвящен изучению ФОРТ-ориентированных СПТ. Известно, что языки "сшитого кода" использовались при непосредственной реализации языков UCSD-Pascal, Extended Basic, Pascal, Lisp (несколько реализаций). В дальнейшем идея использования языка ФОРТ в качестве промежуточного языка транслятора развивалась и приводила к созданию ФОРТ-ориентированных СПТ. В данной работе рассматриваются четыре системы:

- 1) система "ШАГ" (Агамирзян И.Р. - ИТА АН СССР, Ленинград),
- 2) система, разработанная в ЛГУ (авторы Кириллин В.А., Клубович А.А., Ноздрунов Н.Р. - Ленинград),
- 3) система, разработанная в РГУ (авторы Литвиненко А.Н., Демин С.П. - ВЦ РГУ, Ростов-на-Дону),
- 4) СПТ TARTU (Томбак М., Соо В., Пейал Я. - ТГУ, Тарту).

Все эти системы основываются на языке ФОРТ (хотя на разных версиях ФОРТ-а), используя ФОРТ как целостную технологическую среду. Средства описания синтаксиса в этих системах разные и не рассматриваются подробнее. Общей чертой является использование расширенного (ФОРТ-ориентированного) метода семантических процедур, при котором "процедурами" являются любые средства языка ФОРТ - подпрограммы, типы данных, макро-средства и т.п. Эти т.н. семантические понятия образуют своего рода лексикон языка, элементы которого связываются с (абстрактной) синтаксической структурой языка. Несмотря на различия в методах установления такой связи можно сказать, что рассматриваемые системы создают средства для перевода с исходного языка на язык семантических понятий.

Далее анализируются недостатки ФОРТ-ориентированных СПТ. Отмечается, что особую роль в СПТ играют средства, позволяющие проверить корректность некоторых аспектов описания исходного языка (напр. алгоритм Кнута для проверки корректности системы правил вычисления значений атрибутов). Наличие таких средств повышает надежность создаваемых трансляторов и дает возможность обнаружить некоторые ошибки в описании языка на самых ранних этапах разработки. В первых трех системах упомянутые средства обнаружения ошибок в описании языка практически отсутствуют (но имеются средства отладки, которые позволяют выявить подобные ошибки позже).

Наиболее трудным вопросом в рассматриваемых системах является согласование набора семантических понятий с синтаксической структурой языка. Первым шагом при решении этой проблемы является создание средств, позволяющих по описанию семантических понятий и описанию синтаксической структуры проверить те аспекты их согласованности, которые являются самыми важными. От выбора таких аспектов зависит аппарат описания семантических понятий.

В разделе 1.4 рассматривается конкретный представитель ФОРТ-ориентированных систем - СПТ ТАРТУ. В этой системе для перевода используется довольно простой механизм - несколько модифицированный синтаксически управляемый (СУ-) перевод. Вводится понятие режима перевода. Пользователю предоставляется возможность расширить имеющийся набор режимов. Метод перевода иллюстрируется конкретным примером.

В разделе 1.5 приводятся основные выводы первой главы:

- 1) анализ применяемых методов описания языков в ФОРТ-ориентированных СПТ указывает на необходимость повышения технологичности этих методов;
- 2) одним из путей для достижения этой цели является создание средств автоматизированной проверки наиболее важных аспектов согласованности лексикона семантических понятий и синтаксической структуры языка;
- 3) ввиду того, что семантические понятия реализуются на языке

ФОРТ, существенным аспектом при описании семантических понятий является спецификация использования магазина ими;

4) для исследования такого аспекта целесообразно создать формальный аппарат;

5) на основе этого аппарата необходимо уточнить понятие корректности описания языка и разработать методы проверки такой корректности.

Во второй главе определяется понятие спецификации ФОРТ-программы и разрабатывается новый формальный аппарат - алгебра спецификаций языка ФОРТ. На основе алгебры спецификаций определяется понятие корректной ФОРТ-программы и предлагается решение проблемы контроля типов передаваемых через магазин параметров - статический контроль типов. Изучаются алгебраические свойства полугруппы спецификаций и свойства частичного порядка на этой полугруппе.

ФОРТ-понятия обмениваются информацией через магазин данных. Ориентированное на человека описание языка всегда содержит спецификации использования магазина для всех понятий в виде:

типы входных параметров --- типы выходных параметров.

Списки типов упорядочены - верхний элемент магазина является последним в списке. Количество типов и их имена выбираются в зависимости от проблемной области. Например, спецификация понятия "store" (запись значения по адресу) пишется в виде  $[val\ ad\ \text{---} ]$ , а спецификация понятия "fetch" (разыменование) - в виде  $[ad\ \text{---}\ val]$ . Основой при создании формального аппарата является именно такое, привычное для ФОРТ-программиста понятие спецификации.

В разделе 2.1 формализуется понятие спецификации и изучаются алгебраические свойства этого формализма. Сначала вводятся некоторые обозначения и определения:

- $A$  - алфавит (конечное множество символов),
- $A^*$  - множество слов в алфавите  $A$ ,
- $\Lambda$  - пустое слово ( $\Lambda \in A^*$  при любом  $A$ ),
- $ab$  - конкатенация слов  $a$  и  $b$ .

Т.н. нулевая спецификация обозначается через  $0$  и множество спецификаций над алфавитом  $A$  определяется следующим образом:

$$\Phi(A) = A^* \times A^* \cup \{0\}.$$

Пара  $(\lambda_1, \lambda_2) \in A^* \times A^*$  обозначается через  $[\lambda_1 \text{ --- } \lambda_2]$ .

Пустая спецификация определяется как пара

$$(\Lambda, \Lambda) = [ \text{---} ] \in A^* \times A^*$$

и обозначается через  $\mathbb{1}$ .

Произведение спецификаций определяется следующим образом:

- 1)  $\forall \lambda \in \Phi : \lambda 0 = 0 \lambda = 0,$
- 2)  $\forall \lambda, t \in \Phi \setminus \{0\} : \lambda t = [\lambda_1 \text{ --- } \lambda_2] [t_1 \text{ --- } t_2] =$   
 $= \begin{cases} [\alpha \lambda_1 \text{ --- } t_2], & \text{если } t_1 = \alpha \lambda_2, \\ [\lambda_1 \text{ --- } \beta t_2], & \text{если } \lambda_2 = \beta t_1, \\ 0, & \text{в остальных случаях.} \end{cases}$

Нулевая спецификация характеризует ошибочную ситуацию - несогласованность типов.

**Теорема 1.** Множество  $\Phi$  образует полугруппу с единицей и нулем, т.е.

- 1)  $\forall \lambda, t \in \Phi : \lambda t \in \Phi,$
- 2)  $\forall \tau, \lambda, t \in \Phi : (\tau \lambda) t = \tau (\lambda t),$
- 3)  $\forall \lambda \in \Phi : \lambda \mathbb{1} = \mathbb{1} \lambda = \lambda,$
- 4)  $\forall \lambda \in \Phi : \lambda 0 = 0 \lambda = 0.$

Пусть  $\lambda \in \Phi$ . Тогда элемент  $\lambda^{-1} \in \Phi$  определяется следующим образом:

- 1) если  $\lambda = 0$ , то  $\lambda^{-1} = 0,$
- 2) если  $\lambda = [\lambda_1 \text{ --- } \lambda_2]$ , то  $\lambda^{-1} = [\lambda_2 \text{ --- } \lambda_1].$

Из  $\Phi$  выделяются следующие подмножества:

$$L = \{ [a \text{ ---}] \mid a \in A^* \},$$

$$R = \{ [ \text{---} b ] \mid b \in A^* \},$$

$$E = \{ [c \text{ ---} c] \mid c \in A^* \} \cup \{0\}.$$

**Теорема 2.** Справедливы следующие утверждения:

- 1)  $E$  является коммутативной подполугруппой с единицей и нулем, причем элементы множества  $E$  и только они являются идемпотентами полугруппы  $\Phi$ ,
- 2)  $L$  и  $R$  являются подполугруппами с единицей, причем  $\lambda \in L$  тогда и только тогда, когда  $\lambda^{-1} \in R$ ,

- 3)  $\lambda t \in L$  влечет  $t \in L$  и  $\lambda t \in R$  влечет  $\lambda \in R,$
- 4)  $LR = \Phi \setminus \{0\}$  и  $RL \subset R \cup L \cup \{0\},$
- 5) если  $\lambda \in L$  и  $t \neq 0$  или  $\lambda \neq 0$  и  $t \in R$ , то справедливо  $\lambda t \neq 0.$

Из утверждения 1 теоремы 2 следует, что  $\Phi$  является т.н. инверсной полугруппой.

Подполугруппа  $R$  и свободная полугруппа  $A^*$  изоморфны:

$$[ \text{---} c ] [ \text{---} d ] = [ \text{---} cd ].$$

Установлено, что если  $\lambda \in L \setminus \{\mathbb{1}\}$ , то множества вида  $\lambda \Phi$  являются собственными главными правыми идеалами полугруппы  $\Phi$ , а множества вида  $\Phi t$ , где  $t \in R \setminus \{\mathbb{1}\}$  - собственными главными левыми идеалами. Более того,  $\Phi$  является 0-бипростой полугруппой, в которой отсутствуют 0-минимальные идеалы.

В разделе 2.2 определяется частичный порядок на полугруппе  $\Phi$ , изучаются его свойства и связанные с ним понятия.

Сначала определяется отношение  $\leq$ , полагая  $\lambda \leq t$  тогда и только тогда, когда  $\lambda t^{-1} = \lambda \lambda^{-1}$ . Так как  $0 t^{-1} = 0 0 = 0$ , то  $0 \leq t$  для любого  $t \in \Phi$ . Из того, что  $\lambda \leq t$  следует, что  $\lambda \tau \leq \tau t$ ,  $\lambda \tau \leq t \tau$  и  $\lambda^{-1} \leq t^{-1}$ .

**Теорема 3.** Следующие утверждения равносильны:

- 1)  $[\lambda_1 \text{ --- } \lambda_2] \leq [t_1 \text{ --- } t_2],$
- 2)  $\exists a \in A^* : [\lambda_1 \text{ --- } \lambda_2] = [a t_1 \text{ --- } a t_2],$
- 3)  $[ \text{---} \lambda_1 ] [t_1 \text{ --- } t_2] [ \text{---} \lambda_2 ] = [ \text{---} ],$
- 4)  $[ \text{---} \lambda_1 ] [t_1 \text{ --- } t_2] = [ \text{---} \lambda_2 ],$
- 5)  $[t_1 \text{ --- } t_2] [ \text{---} \lambda_2 ] = [ \lambda_1 \text{ ---} ].$

Так как отношение  $\leq$  является частичным порядком, нас интересует, какие элементы сравнимы между собой.

**Теорема 4.** Следующие утверждения эквивалентны:

- 1)  $\lambda \neq 0, t \neq 0$  и  $\lambda$  сравним с  $t$ .
- 2) существует  $\tau \neq 0$  такой, что  $\tau \leq \lambda$  и  $\tau \leq t$ ,
- 3) существует  $u \in \Phi$  такой, что  $\lambda \leq u, t \leq u$  и либо  $\lambda t^{-1} \neq 0$ , либо  $\lambda^{-1} t \neq 0$ .

Рассматриваются понятия верхней и нижней грани подмножеств полугруппы  $\Phi$ . Всякое двухэлементное подмножество

$\{s, t\} \subset \Phi$  имеет нижнюю грань:

$$\inf \{s, t\} = \begin{cases} s & \text{если } s \leq t, \\ t & \text{если } t \leq s, \\ \emptyset & \text{если } s \text{ несравним с } t. \end{cases}$$

Если для ненулевых элементов  $s, t \in \Phi$  существуют  $a, b, c, d, e \in \mathcal{A}^*$  такие, что  $s = [abd \dots abe]$  и  $t = [cbd \dots cbe]$  (причем последний символ слова  $a$  не совпадает с последним символом слова  $c$ ), то существует

$$\sup \{s, t\} = [bd \dots be].$$

В разделе 2.3 вводятся понятия корректной ФОРТ-программы и замкнутой ФОРТ-программы. Сначала на множестве рассматриваемых ФОРТ-понятий  $\Delta$  определяется отображение  $\mathcal{L}: \Delta^* \rightarrow \Phi$ :

- 1)  $\forall \Pi \in \Delta: \mathcal{L}(\Pi) \in \Phi \setminus \{0\}$  является спецификацией понятия  $\Pi$ ,
- 2)  $\mathcal{L}(\Lambda) = \mathbb{1}$ ,
- 3)  $\forall \omega \in \Delta^*, \forall \Pi \in \Delta: \mathcal{L}(\omega\Pi) = \mathcal{L}(\omega)\mathcal{L}(\Pi)$ .

Последовательность ФОРТ-понятий (ФОРТ-программа)  $\omega \in \Delta^*$  называется корректной, если  $\mathcal{L}(\omega) \neq 0$ , и замкнутой, если  $\mathcal{L}(\omega) = \mathbb{1}$ . Понятия, из которых состоит корректная ФОРТ-программа, согласуются по типам входных и выходных параметров. Корректная программа является замкнутой, если у нее нет ни входных, ни выходных параметров.

В третьей главе рассматривается применение алгебры спецификаций в СПТ. Определяется корректность СУ-схемы, выходным языком которой является ФОРТ. Доказывается основная теорема о корректности перевода. На базе этой теоремы создаются алгоритмы проверки корректности. Рассматривается практическое применение предложенного метода спецификации.

В разделе 3.1 проблема проверки корректности перевода приводится к проблеме разрешимости некоторой системы неравенств.

Схема СУ-перевода - это пятерка  $T = (N, \Sigma, \Delta, R, S)$ , где  $N$  - нетерминальный алфавит,  $S \in N$  - фиксированный начальный символ (аксиома),  $\Sigma$  - входной алфавит,  $\Delta$  - выходной алфавит, а  $R$  - множество правил перевода вида

$$A_0 \rightarrow x_0 A_1 x_1 \dots x_{p-1} A_p x_p, z_0 B_1 z_1 \dots z_{p-1} B_p z_p$$

( $x_i \in \Sigma^*, z_i \in \Delta^*, A_i, B_i \in N$ ), где вектор  $(B_1, \dots, B_p)$  является некоторой перестановкой вектора  $(A_1, \dots, A_p)$ .

Схема СУ-перевода естественным образом определяет множество пар цепочек, выводимых из пары  $(S, S)$ . Первые компоненты всех выводимых пар составляют входной язык  $L_1$ , а вторые компоненты - выходной язык  $L_2$ . Если  $(x, y)$  является выводимой парой, то цепочка  $y$  называется переводом цепочки  $x$ . Предполагается, что грамматика  $G_1$  входного языка СУ-схемы  $T$  является приведенной, и рассматривается грамматика выходного языка

$$G_2 = (N, \Delta, P, S).$$

Выходной язык определяется как множество

$$L_2 = \{\omega \mid \omega \in \Delta^* \text{ \& } S \Rightarrow^+ \omega\}.$$

Также предполагается, что для выходных символов задан комплект спецификаций

$$\mathcal{L}(\Delta) \subset \Phi \setminus \{0\},$$

по которому вышеуказанным способом определяется гомоморфизм

$$\mathcal{L}: \Delta^* \rightarrow \Phi.$$

СУ-схема  $T$  называется корректной, если любая выходная последовательность  $\omega \in L_2$  является замкнутой, т.е.  $\mathcal{L}(\omega) = \mathbb{1}$ . Замкнутость (не просто корректность) объектных программ требуется потому, что магазины ФОРТ-системы скрыты от пользователя конкретного транслятора.

Система неравенств  $I(T, \mathcal{L})$  строится следующим образом:

- 1) каждому нетерминалу  $A \in N$  ставится в соответствие неизвестная  $Z(A) \in \Phi$ ,
- 2) правила выходной грамматики заменяются неравенствами - из правила вида  $A \rightarrow X_1 \dots X_k$  строится неравенство  $Z(A) \leq Y_1 \dots Y_k$ , где  $Y_i = \mathcal{L}(X_i)$ , если  $X_i \in \Delta$  и  $Y_i = Z(X_i)$ , если  $X_i \in N$ .

Если правая часть правила пуста, то  $Z(A) \leq \mathbb{1}$ .

- 3) добавляется неравенство  $\mathbb{1} \leq Z(S)$ , где  $S$  - начальный символ СУ-схемы  $T$ .

Для нетерминальных символов выходной грамматики определя-



ются следующие вспомогательные множества:

$$C(A) = \{ (u, v) \in \Delta^* \times \Delta^* \mid S \Rightarrow^* uAv \},$$

$$L(A) = \{ \omega \in \Delta^* \mid A \Rightarrow^+ \omega \}.$$

**Теорема 5** (основная теорема). Следующие утверждения эквивалентны:

1) все выходные последовательности СУ-схемы  $T$  замкнуты,

2) система неравенств  $I(T, \Delta)$  разрешима,

3) для каждого нетерминала  $A \in N$  существует величина  $m(A) = \sup \{ \Delta(\omega u) \}^{-1} \mid (u, v) \in C(A) \}$  и выполняется неравенство  $m(A) \leq \inf \{ \Delta(\omega) \mid \omega \in L(A) \}$ .

Доказательство этой теоремы основывается на результатах, изложенных во второй главе.

Практический метод проверки корректности описания перевода, разработка которого и является главной целью диссертации, создан на базе утверждения 3 теоремы 5 с учетом требования, что некорректные описания должны быть отвергнуты за конечное время.

В разделе 3.2 приводятся алгоритмы, позволяющие по СУ-схеме  $T$  и комплексу спецификаций семантических понятий  $\Delta(\Delta)$  установить их согласованность. Если СУ-схема некорректна относительно комплекта спецификаций, то работа завершается и дается дополнительная информация о причинах несогласованности. Обработка исходных данных делится на три этапа.

1. Вычисляется приближенное решение

$$\{ \ell(A) \mid A \in N \}$$

системы  $I(T, \Delta)$  такое, что для всех  $A \in N$  справедливо

$$\Delta(A) \leq \ell(A),$$

где  $\{ \Delta(A) \mid A \in N \}$  - решение системы  $I(T, \Delta)$ .

2. По комплексу  $\{ \ell(A) \mid A \in N \}$  вычисляются нижние границы  $c(A)$  компонент решения, т.е.

$$c(A) \leq \Delta(A) \text{ для всех } A \in N.$$

3. Приближенное решение уточняется, причем соблюдение условия  $c(A) \leq \ell(A)$  для всех  $A \in N$  гарантирует завершение работы в случае некорректной схемы.

В разделе 3.3 обсуждаются вопросы практического использования предложенной методики. Оказывается, что один и тот же аппарат может применяться в разных целях - для описания подпрограмм, макросредств и т.д. Приводится пример.

В заключении перечисляются основные результаты диссертационной работы и направления дальнейших исследований.

В приложении I приводится реализация алгебраических операций над ФОРТ-спецификациями.

В приложении 2 излагается реализация статического контроля типов для ФОРТ-программ.

Программы в приложениях написаны на языке fig-FORTH.

#### ЗАКЛЮЧЕНИЕ И ОСНОВНЫЕ ВЫВОДЫ

В диссертации получены следующие основные результаты:

1. Разработан метод спецификации ФОРТ-понятий, который согласуется с общепринятым способом описания входных и выходных параметров ФОРТ-программы. Создан соответствующий формальный аппарат - алгебра спецификаций языка ФОРТ.
2. Определено понятие корректности ФОРТ-программы относительно согласованности типов передаваемых через магазин параметров и реализованы средства для статической проверки такой корректности на основе алгебры спецификаций.
3. Проанализированы существующие методы описания и реализации языков программирования в ФОРТ-ориентированных СПТ. Изучены проблемы корректности описания перевода с исходного языка на язык семантических понятий.
4. Теоретическим исследованием свойств алгебры спецификаций проблема проверки корректности описания перевода приведена к проблеме разрешимости некоторой системы неравенств.
5. В виде конкретных алгоритмов реализован метод решения этой системы, позволяющий установить корректность описаний реализуемых языков в ФОРТ-ориентированных СПТ.

Полученные результаты позволяют повысить технологичность системы программирования ФОРТ и созданных на её основе инструментальных систем, особенно СИТ. Методы, предложенные в диссертации, оказались целесообразными и практичными при автоматизированной реализации языков программирования в СИТ ТАРТУ, а также при разработке программного обеспечения на языке ФОРТ. Предложенная алгебра спецификаций охватывает пока только один из существенных аспектов ФОРТ-программы - обмен информацией через магазин. Поэтому актуальной задачей будущего является усовершенствование метода спецификаций.

Основные положения диссертации изложены в следующих работах.

1. Пейал Я.Р. Опыт применения FORTH-ориентированной СИТ для СМ ЭВМ. - Информационный сборник / ЦНИИТЭИ приборостроения, ТС-12, вып. 4: Программное обеспечение мини- и микроЭВМ семейства СМ ЭВМ. - Москва, 1984, с. 57.
2. Пейал Я.Р., Соо В.К., Томбак М.О. СИТ для микро-ЭВМ. - В сб.: Индивидуальные диалоговые системы на базе микро-ЭВМ (персональные компьютеры). Диалог-84-микро: Тезисы докл. - Ленинград: Наука, 1984, с. 102-104.
3. Пейал Я.Р., Соо В.К., Томбак М.О. Использование расширяемого языка в СИТ. - Труды ВЦ Тартуского госунив. вып. 52. - Тарту, 1985, с. 39-54.
4. Пейал Я.Р. О свойствах спецификаций языка ФОРТ. - Труды ВЦ Тартуского госунив. вып. 54. - Тарту, 1986, с. 68-84.
5. Пейал Я.Р. Алгебра спецификаций языка ФОРТ и её применение в СИТ. - В сб.: III Всесоюзная конференция "Автоматизация производства систем программирования": Тезисы докл. - Таллин, 1986, с. 40-42.

*Peial*

Академия наук Эстонской ССР, Институт кибернетики.

П ё й а л Янус Раймондович. Спецификации ФОРТ-программ и их применение в системах построения трансляторов. Автореферат.  
На русском языке.

Подписано в печать 13.II.86 МВ- 10020 Бумага 60x84/16.  
Печатных листов 1,0. Условно-печатных листов 0,93. Учетно-из-  
дательских листов 0,74. Тираж 100. Заказ № 396. Бесплатно.  
Ротапринт АН ЭССР, 200001 Таллин, бульвар Эстония, 7.

Бесплатно.