



Содержание

Языки и эволюция технологий программирования

Языки программирования высокого уровня
 Язык и его реализация
 Компилятор, интерпретатор, конвертор
 Метаязыки

Генеалогия языков программирования

Первое поколение языков
 Фортран
 Алгол-60
 Кобол

Две попытки объять необъятное
 ПЛ/1
 Алгол-68

Интерактивное программирование для всех
 Бейсик

Структурное программирование
 Основы структурного программирования
 Паскаль
 Язык Си

Модульность, надежность, абстракция
 Ада
 Модуль-2
 Абстрактные типы данных

Объектно-ориентированное программирование

Язык программирования Си++
 Язык программирования Оберон
 Язык программирования Ява
 Язык программирования Си#
 Примеры использования объектной технологии

Языки-концепции

Форт
 Лисп
 Пролог
 Смолток

Языки Интернета

HTML
 Ява и апплеты
 Скриптовые языки
 Языки CGI-программирования
 Языки активных серверных страниц
 Языки Интернета: повторение пройденного

Какой язык лучше. Сравнительная оценка языков программирования

Арифметика синтаксиса
 Экспертная оценка языков программирования
 Важнейшие языки

Теоретические основы трансляции

Формальные языки и грамматики

Основные термины и определения
 Примеры языков
 Порождающие грамматики (грамматики Н. Хомского)
 Еще несколько определений
 Дерево вывода
 Задача разбора
 Для чего надо решать задачу разбора
 Домино Де Ремера
 Разновидности алгоритмов разбора
 Эквивалентность и однозначность грамматик
 Иерархия грамматик Н. Хомского

Автоматные грамматики и языки

- Граф автоматной грамматики
- Конечные автоматы
- Преобразование недетерминированного конечного автомата (НКА) в детерминированный конечный автомат (ДКА)
- Таблица переходов детерминированного конечного автомата
- Программная реализация автоматного распознавателя
- Дерево разбора в автоматной грамматике
- Пример автоматного языка
- Синтаксические диаграммы автоматного языка
- Регулярные выражения и регулярные множества
- Эквивалентность регулярных выражений и автоматных грамматик
- Для чего нужны регулярные выражения
- Регулярные выражения как языки
- Расширенная нотация для регулярных выражений

Контекстно-свободные (КС) грамматики и языки

- Однозначность КС-грамматики
- Алгоритмы распознавания КС-языков
- Распознающий автомат для КС-языков
- Самовложение в КС-грамматиках
- Синтаксические диаграммы КС-языков
- Определение языка с помощью синтаксических диаграмм
- Синтаксический анализ КС-языков методом рекурсивного спуска
- Требование детерминированного распознавания
- LL-грамматики
- Левая и правая рекурсия
- Синтаксический анализ арифметических выражений
- Включение действий в синтаксис
- Обработка ошибок при трансляции
- Табличный LL(1)-анализатор
- Рекурсивный спуск и табличный анализатор

Трансляция выражений

- Польская запись
- Алгоритм вычисления выражений в обратной польской записи
- Перевод выражений в обратную польскую запись
- Интерпретация выражений
- Семантическое дерево выражения
- Упражнения для самостоятельной работы

Трансляция языков программирования**Описание языков программирования**

- Метаязыки
- БНФ
- Синтаксические диаграммы
- Расширенная форма Бэкуса-Наура (РБНФ)
- Описания синтаксиса языков семейства Си
- Описания синтаксиса языка Ада
- Определение синтаксиса Кобола и ПЛ/1

Язык программирования «О»

- Краткая характеристика языка «О»
- Синтаксис «О»
- Пример программы на «О»

Структура компилятора

- Многопроходные и однопроходные трансляторы

Компилятор языка «О»

- Вспомогательные модули компилятора

Лексический анализатор (сканер)

- Виды и значения лексем
- Лексический анализатор языка «О»

Синтаксический анализатор**Контекстный анализ**

- Таблица имен
- Контекстный анализ модуля
- Трансляция списка импорта
- Трансляция описаний
- Контекстный анализ выражений
- Контекстный анализ операторов

Генерация кода

- Виртуальная машина
- Архитектура виртуальной машины
- Программирование в коде виртуальной машины
- Реализация виртуальной машины
- Генератор кода
- Распределение памяти
- Генерация кода для выражений
- Генерация кода для операторов
- Завершение генерации
- Назначение адресов переменным

Трансляция процедур

- Расширенный набор команд виртуальной машины
- Процедуры без параметров и локальных переменных
- Процедуры с параметрами-значениями без локальных переменных
- Процедуры с параметрами-значениями и локальными переменными.
- Простейшая оптимизация кода
- Процедуры-функции с параметрами-значениями и локальными переменными
- Трансляция оператора RETURN
- Особенность трансляции параметров-переменных
- Пример программы на языке «О с процедурами»

Конструкция простого ассемблера

- Язык ассемблера виртуальной машины
- Реализация ассемблера

Автоматизация построения и мобильность трансляторов

- Автоматический анализ и преобразование грамматик
- Автоматическое построение компилятора и его частей
- Использование языков высокого уровня
- Самокомпилятор. Раскрутка
- Примеры раскрутки
- Унификация промежуточного представления

Язык программирования Оберон-2

От переводчика

1. Введение
2. Синтаксис
3. Словарь и представление
4. Объявления и области действия
5. Объявления констант
6. Объявления типов
 - 6.1 Основные типы
 - 6.2 Тип массив
 - 6.3 Тип запись
 - 6.4 Тип указатель
 - 6.5 Процедурные типы
7. Объявления переменных
8. Выражения
 - 8.1 Операнды
 - 8.2 Операции
9. Операторы
 - 9.1 Присваивания
 - 9.2 Вызовы процедур
 - 9.3 Последовательность операторов
 - 9.4 Операторы IF
 - 9.5 Операторы CASE
 - 9.6 Операторы WHILE
 - 9.7 Операторы REPEAT
 - 9.8 Операторы FOR.
 - 9.9 Операторы LOOP
 - 9.10 Операторы возврата и выхода
 - 9.11 Операторы WITH
10. Объявления процедур
 - 10.1 Формальные параметры
 - 10.2 Процедуры, связанные с типом
 - 10.3 Стандартные процедуры
11. Модули

Приложение А: Определение терминов

- Целые типы
- Вещественные типы
- Числовые типы
- Одинаковые типы
- Равные типы
- Поглощение типов
- Расширение типов (базовый тип)
- Совместимость по присваиванию
- Совместимость массивов
- Совместимость выражений
- Совпадение списков формальных параметров

Приложение В: Синтаксис Оберона-2**Приложение С: Модуль SYSTEM****Приложение D: Среда Оберон**

- D1. Команды
- D2. Динамическая загрузка модулей
- D3. Сбор мусора
- D4. Смотритель
- D5. Структуры данных времени выполнения

Текст компилятора языка «О» на Паскале**Текст компилятора языка «О» на Обероне**

- Отличия версий для компиляторов JOB и XDS
- Изменение обозначений
- Изменения в структуре компилятора

Текст компилятора языка «О» на Си/Си++**Текст компилятора «О» на языке программирования Ява****Текст компилятора «О» на языке программирования Си#****Сравнение компиляторов**

- Компиляторы языка «О»
- Компиляторы компилируют компиляторы

Ассемблер виртуальной О-машины**Литература**

С. З. Сverdlov

Памяти моего отца

От автора

Программист не может не интересоваться языками программирования. Даже если в своей работе или учебе вы пользуетесь каким-то одним языком, иметь представление о других, их свойствах, областях применения, истории и современном состоянии полезно и важно.

Первая часть книги содержит обзор языков высокого уровня и связанных с их эволюцией технологий структурного, модульного и объектно-ориентированного программирования. Проводится сравнительный анализ языков, в том числе на основе объективных критериев, даются экспертные оценки. Подробно обсуждаются Ява и Си#. Отдельный раздел посвящен языкам Интернета. Изложение сопровождается большим числом примеров программ. Знакомство с этим обзором поможет составить предметное представление о свойствах основных языков, обоснованно подходить к их выбору.

Реальное применение языка программирования невозможно без соответствующей системы программирования, основу которой составляет транслятор. Конструирование трансляторов — вторая тема книги. Транслятор — весьма непросто устроенная программа, разработка которой невозможна без знакомства с элементами теории формальных языков и соответствующей техникой программирования.

Создание компилятора или интерпретатора — увлекательное дело, дающее возможность совершенствовать свои программистские навыки, глубже понять устройство самих языков. Конечно, не каждому программисту доводится участвовать в разработке транслятора и уж тем более в создании языка программирования. Но знание методов программирования трансляторов оказывается полезным в самых разных ситуациях: от решения олимпиадных задач до разработки и реализации входных языков прикладных систем.

В книге обсуждаются все элементы транслятора и этапы реализации языка от спецификации до формирования машинного кода. Приводится полный исходный текст компилятора на нескольких языках программирования. Рассматривается конструкция простого двухпроходного ассемблера. Обсуждаются возможности автоматизации построения трансляторов и способы повышения их мобильности.

Особую роль в изложении играет язык Оберон (и его расширение Оберон-2) — один из самых совершенных и современных языков программирования, разработанный Н. Виртом — автором Паскаля и Модулы-2. Оберон необычайно прост и в то же время содержит все необходимые средства структурного, объектно-ориентированного и модульно-компонентного программирования. В книге публикуется спецификация Оберон-2. Трансляция языков программирования рассматривается на примере разработки компилятора для подмножества Оберона. Надеюсь, что читатели с удовольствием познакомятся с этим языком, даже если не обязательно в дальнейшем будут использовать его.

Книга написана по материалам одноименного курса, который в течение ряда лет читается студентам факультета прикладной математики и компьютерных технологий Вологодского педагогического университета. В работе над курсом и книгой я руководствовался несколькими основными принципами: изложение должно быть понятным, конструктивным и интересным. В какой мере это удалось — судить вам. Некоторые суждения, высказанные в книге, заведомо субъективны. Вы можете с ними не согласиться, но, надеюсь, признаете, что они аргументированы.

Книга адресуется студентам вузов, специализирующимся по компьютерным технологиям, программистам-практикам и всем, кто интересуется программированием. Предполагается, что читатель имеет начальные навыки программирования и обладает математической подготовкой в пределах программы средней школы.

Благодарности

Мне доставляет удовольствие поблагодарить тех, кто помог советом и делом в работе над книгой. Хочу выразить признательность блистательному создателю Паскаля, Модулы и Оберона проф. Н. Вирту, который ответил на мои письма и вместе с проф. Х. Мёссенбёком дал согласие на опубликование русского перевода спецификации Оберона-2. На протяжении всей работы я имел удовольствие дружеского общения с Е. А. Зуевым, который оказал неоценимую поддержку и поделился множеством интересных соображений. Я благодарен рецензентам — проф. В. О. Сафонову, высказавшему множество ценных замечаний, и проф. А. Н. Терехову, который также был моим консультантом по Алголу-68.

В течение ряда лет студенты факультета прикладной математики и компьютерных технологий ВГПУ, проходившие курс, сделали немало предложений и замечаний, способствовавших его улучшению. Большое влияние на содержание книги и мое понимание проблематики языков программирования и методов трансляции оказал Ф. В. Меньшиков. Он, а также Я. А. Музыкантов и В. С. Губа стали первыми читателями рукописи. Их поправки были очень ценны.

Благодарю за консультации И. Р. Агамирзяна, за беспримерное терпение — главного редактора издательства «Питер» Е. В. Строганову и руководителя проекта И. В. Шапошникова, редакторов книги А. В. Пасечника и К. А. Кнопа.

27 декабря 2004 г.
Вологда