

## Теория компиляторов

Карпов В.Э.(ктн., доцент, вице-президент Российской ассоциации искусственного интеллекта)

1. Введение. Предмет курса и его связь со смежными дисциплинами.
  - a. Архитектура ЭВМ. История развития вычислительной техники. Автоматически управляемые вычислительные машины.
  - b. Организация памяти. Элементная база ЭВМ. Структура и принципы организации машин фон-неймановской архитектуры. Современные тенденции развития вычислительной техники. Машины не-фон-Неймановской архитектуры. Параллельные машины. Нейрокомпьютеры. Машины искусственного интеллекта.
2. Основные понятия и определения. Терминология. Ассемблер и автокод. Логическая структура компилятора. Процесс компиляции. Основные части компилятора. Понятие синтаксического и семантического анализа. Многопроходные компиляторы.
3. Формальные системы и языки программирования. Принципы описания и задания языков. Формальные грамматики.
4. Иерархия Хомского. Регулярные и ОК-грамматики. Форма Бэкуса-Наура.
5. Регулярные грамматики. Конечные автоматы. Формальное определение. Детерминированные и недетерминированные конечные автоматы
6. Программирование сканера. Регулярные выражения и конечные автоматы. Структура сканера.
7. Организация таблиц символов. Хеш-функции, хеш-адресация. Способы решения задачи коллизии. Рехеширование.
8. Контекстно-свободные грамматики. Ок-грамматики. Синтаксический анализ и генерация промежуточных форм представления программы.
  - a. Схема синтаксически управляемого перевода.
  - b. Автоматы с магазинной памятью.
  - c. Операторные грамматики (грамматики простого предшества).
  - d. Матрицы переходов.
9. Внутренние формы представления исходной программы. Польская форма. Тетрады.
10. Оптимизация программ. Оптимизация линейных участков, циклов и условий.
11. Интерпретаторы. Принципы построения. Особенности интерпретируемых языков. Компиляторы компиляторов.
12. Генерация объектного кода. Генерация кодов арифметических выражений. Система адресации. Объектные модули и компоновка. Объектный язык. Регистры. Модификация адресов. Принципы построения общесистемных ассемблеров. Основные конструкции и алгоритмы.
13. Отладчики. Общие принципы отладки программ. Технологии отладки. Трассировка и контрольные точки. Отладка в интегрированных средах.
14. Макросы. Макрокоманды и макроопределения. Макроязык и макропроцессор. Реализация макропроцессоров. Однопросмотровые и двупросмотровые алгоритмы. Макроассемблеры.
15. Загрузчики. Принципы организации памяти. Выделение памяти различным типам данных. Динамическое распределение. Схемы загрузки. Абсолютные и настраиваемые загрузчики. Связывание подпрограмм. Принципы проектирования загрузчиков. Алгоритм работы загрузчика.

## Объектно-ориентированное программирование

1. Современные системы управления сложными объектами основаны на применении высокотехнологичных приемов и методов программирования, одним из которых является объектно-ориентированное программирование (ООП). Несмотря на то, что история ООП насчитывает уже более 20 лет, основные тенденции современного программирования остаются неизменными. Изучению общих принципов объектно-ориентированного программирования и получению навыков практического применения ООП при создании сложных программных комплексов и посвящена настоящая дисциплина.
2. Курс разбит на 2 части. Первая часть посвящена основам программирования на, пожалуй, единственном "настоящем" на данный момент времени объектно-ориентированном языке – языке Смолток.
3. Язык Смолток – очень элегантен и прост с точки зрения синтаксиса (попробуйте найти еще один язык, в котором было бы всего три оператора). Тем не менее это одно из наиболее эффективных средств для моделирования сложных систем, на этом языке создаются надежные сверхсложные программные комплексы.
4. Во второй части курса рассматриваются принципы объектно-ориентированного программирования на языке C++. Рассматриваются основные ООП-конструкции, приемы и методы.
5. На основании изучения этой дисциплины студент должен уметь применять приемы и методы ООП в своей практической деятельности, знать основные принципы организации сложных объектно-ориентированных систем, иметь представление об основных тенденциях развития современных информационных технологий. Этот курс является составной частью цикла специальных дисциплин, определяющих подготовку студентов в области современных информационных технологий.

## Рекомендуемая литература

1. Донован Дж. Системное программирование. /Пер.с англ. -М.:Мир, 1975
2. Грис Д. Конструирование компиляторов для цифровых вычислительных машин. /Пер.с англ. -М.:Мир, 1975.
3. Ульман Дж., Ахо А.В., Уиман Д.Д. Теория синтаксического анализа, перевода и компиляции, 2т.
4. Фути К., Судзуки Н. Языки программирования и схемотехника СБИС: Пер. с япон. -М.:Мир, 1988. -224 с.
5. Бородич Ю.С. и др. Паскаль для персональных компьютеров: Справ.пособие /Ю.С.Бородич, А.Н.Вальвачев, А.И.Кузьмич. -Мн.:Выш.шк.: БФ ГИТМП "Ника", 1991.-365 с.
6. 3. Эллис М., Строуструп В. Справочное руководство по языку программирования C++ с комментариями: Пер.с англ.-М.:Мир, 1992. -445с.
7. Бадд Тимоти. Объектно-ориентированное программирование в действии /Пер. с англ. СПб.:Питер, 1997. -464 с.