

АННОТАЦИЯ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ «Теория алгоритмических языков и трансляторов»

по основной профессиональной образовательной программе по направлению подготовки
09.03.04 «Программная инженерия» (уровень бакалавриата)

Направленность (профиль): Разработка программно-информационных систем

Общий объем дисциплины – 5 з.е. (180 часов)

Форма промежуточной аттестации – Экзамен.

В результате освоения дисциплины у обучающихся должны быть сформированы компетенции с соответствующими индикаторами их достижения:

- ПК-1.1: Выбирает средства для моделирования и анализа программного обеспечения;
- ПК-1.2: Применяет методы формализации и моделирования при конструировании программного обеспечения;

Содержание дисциплины:

Дисциплина «Теория алгоритмических языков и трансляторов» включает в себя следующие разделы:

Форма обучения очная. Семестр 7.

1. Формальные грамматики и языки. Средства для моделирования и анализа программного обеспечения. Методы формализации и моделирования при конструировании программного обеспечения.

Понятие языка и грамматики. Типы грамматик.

КС-грамматики: правила вывода, дерево грамматического разбора, методы преобразования..

2. Автоматы. Конечные автоматы и автоматные грамматики. Синтез конечных автоматов.

Минимизация и детерминизация конечных автоматов. Регулярные выражения.

Автоматы с магазинной памятью и КС-языки..

3. Структура транслятора. Понятие компиляции и интерпретации.

Лексический, синтаксический и семантический уровни языка программирования. требования к КС-грамматикам языков программирования.

Контекстные условия.

Лексика языка программирования.

Типы лексических единиц. Понятие лексемы. Конечный автомат лексики языка.

Реализация лексического уровня. Класс лексического анализатора и структура проекта приложения..

4. Синтаксический анализ. Нисходящий и восходящий синтаксический анализ.

Рекурсивные и магазинные методы синтаксического анализа.

Метод рекурсивного спуска. Понятие синтаксической диаграммы.

Построение и преобразование синтаксических диаграмм.

Программа синтаксического анализа..

5. Семантический анализ. Контекстные условия языков программирования.

Понятие семантического дерева и особенности его реализации.

Хранимая информация. Семантические подпрограммы и их реализация..

6. Методы интерпретации языков программирования. Понятие процесса интерпретации. Способы представления вычисляемых значений.

Интерпретация выражений. Процесс приведения типов.

Интерпретация управляющих структур.

Интерпретация функций и их вызовов, реализация рекурсивных вызовов..

7. Препроцессорный уровень языка программирования. Понятие макрогенерации. Структура языка препроцессорного уровня: лексика, синтаксис, примеры.

Принцип реализации препроцессорного уровня на основе интерпретации.

Синтаксические диаграммы, дерево макроопределений.

Макроподстановка и генерация текста программы..

8. Методы компиляции. Задача компилятора. Понятие промежуточного кода. Типы

промежуточного кода.

Формирование триад для выражений и операторов управления ходом вычислительного процесса.

Методы оптимизации промежуточного кода.

Генерация ассемблерного кода, сегменты констант, данных и кода.

Выделение памяти.

Понятие объектного кода. Задача редактора связей..

Разработал:
профессор
кафедры ПМ

Е.Н. Крючкова

Проверил:
Декан ФИТ

А.С. Авдеев