Содержание курса лекций «Языки программирования» (2014 год)

1. Введение. Основные понятия языков программирования

Определение языков программирования (ЯП), ЯП и основные парадигмы программирования. Исторический очерк развития ЯП. Основные позиции при рассмотрении ЯП. Схема рассмотрения ЯП: базис, средства развития и средства защиты. Основные понятия языков программирования: данные, операции и связывание. Понятие о виртуальной машине языка

2. Базис языков программирования: базисные типы и структуры данных, операторный базис

Арифметические типы данных: целые, плавающие, фиксированные. Проблемы представления чисел и способы их решения в ЯП. Символьные и логические типы данных. Порядковые типы: диапазоны и перечисления. Особенности реализации перечислений в современных ЯП. Ссылки и указатели. Управление памятью. Автоматическая сборка мусора. Объектно-референциальная модель в современных ЯП. Составные типы данных. Массивы и их особенности в современных ЯП. Записи. Недостатки системы типов в традиционных ЯП. Объединения как средство преодоления этих недостатков. Проблемы, связанные с объединениями. Ассоциативные массивы и записи.

Понятие о структурном программировании. Разновидности управляющих конструкций в современных языках программирования. Условные операторы и многовариантные развилки. Циклы. Оператор перехода, связанные с им проблемы и способы их решения в современных ЯП. Особенности реализации циклов-итераторов в современных ЯП

3. Определение новых абстракций. Процедурные абстракции, определение новых типов данных, классы. Инкапсуляция и абстрактные типы данных

Сопрограммы и подпрограммы. Сопрограммы в современных ЯП (на примере итераторов в C# и Python).

Процедурные абстракции. Процедуры и функции в современных ЯП. Передача параметров: семантика и способы реализации. Статический полиморфизм и перегрузка имен подпрограмм. Подпрограммные типы данных. Делегаты и события в языке C#. Анонимные функции. Лямбда-функции и замыкания в современных ЯП (C#, Python, C++).

Понятие логического модуля. Использование модулей для определения новых типов данных. Особенности понятия модуля в современных ЯП. Импорт и экспорт имен. Видимость имен: непосредственная и потенциальная. Управление видимостью. Области видимости и пространства имен. Модульность и технология программирования: проектирование «сверху-вниз» и «снизу-вверх».

Понятие класса. Класс как тип данных. Члены класса: функции, данные. Статические и нестатические члены. Члены - вложенные классы. Статические и нестатические классы. Классы и области видимости.

Понятие специальных функций-членов. Проблема инициализации объектов и способы ее решения. Конструкторы, деструкторы, операторы using и try-finally.

Преобразование типов и классы. Явные и неявные преобразования. Управление преобразованиями в современных ЯП: проблемы и способы их решения. Классы и перегрузка имен. Перегрузка встроенных знаков операций. Итераторы и индексаторы. Классы и стандартные библиотеки. Встроенные классы стандартной библиотеки.

Понятие инкапсуляции. Понятие абстрактного типа данных (АТД) и его достоинства. Инкапсуляция и логические модули. Управление видимостью. Реализация АТД в модульных языках программирования (Ада, Оберон, Модула). Инкапсуляция и классы. Управление видимостью и управление доступом. Пространства имен и инкапсуляция. Реализация АТД с помощью понятия класса. Принцип разделения определения, реализации и использования (РОРИ). Эволюция принципа РОРИ в современных ЯП.

Модульность и раздельная трансляция.

4. Объектно-ориентированные свойства языков программирования. Наследование и динамический полиморфизм. Абстрактные классы и интерфейсы.

Иерархии типов, статические и динамические типы в объектно-ориентированных ЯП. Наследование и области видимости имен. Замещение, перегрузка и скрытие имен при наследовании. Наследование и инкапсуляция. Управление видимостью и доступом при наследовании. Запрещение наследования для классов и методов. Наследование и специальные функции. Понятие о множественном наследовании.

Статическое и динамическое связывание методов. Динамический тип данных и динамическое связывание. Замещение функций и динамическое связывание. Особенности динамического связывания в современных ЯП. Достоинства и недостатки динамического связывания. Снятие динамического связывания. Понятие о мультиметодах. Понятие абстрактного класса (АК). Необходимость понятия АК при проектировании иерархий классов. Воплощение концепции АК в современных ЯП. Абстрактные классы и интерфейсы. Интерфейс как языковая конструкция. Связь интерфейсов и других языковых конструкций (итераторов, сохраняемых объектов и т.д.). Интерфейсы и иерархии классов. Множественное наследование интерфейсов. Реализация интерфейсов и ее особенности современных ЯП. Явная и неявная реализация интерфейсов.

Понятие о динамической идентификации типа.

5. Обобщенное программирование и статический полиморфизм

Понятие о статической параметризации и родовых объектах. Достоинства статической параметризации. Статическая параметризация и ООП. Родовые модули и подпрограммы в языке Ада. Механизм шаблонов в языке Си++. Шаблоны-классы и шаблоны-функции. Параметры шаблонов. Вывод параметров шаблонов. Генерация кода по шаблонам. Полная и частичная специализация шаблонов. Обобщенное программирование на языке Си++. Шаблоны и функциональное программирование. Сравнение механизма шаблонов Си++ и родовых объектов Ады. Особенности статического полиморфизма в языках C# и Java.