14 лекций в 1 семестре

14 лекций во 2 семестре

Ольга Викторовна  
Основы алгоритмизации и программирования

Задание на 1 лабораторную работу:

* Алгоритм – определение, понятие
* Свойства алгоритма
* Способы описания алгоритма
* Основные алгоритмические конструкции
* Как их можно изобразить при описании
* Ещё что-нибудь

Изучаем С++ – императивный язык высокого уровня

**Парадигмы программирования**

Парадигма – это совокупность идей и понятий, определяющих стиль написания компьютерных программ (подход к программированию)

В 40-х годах 20 века Джон Фон Нейман предложил следующую архитектуру компьютера:

1. Он выделил в компьютере 2 главных модуля: процессор, ответственный за вычисления; модуль хранения данных (память)
2. Программы и данные располагаются в одной и той же памяти (концепция хранимой программы) Но она всё равно делится на сегменты

Первый модуль определяет действия, подлежащие выполнению путём считывания команд из второго модуля, т.е. программа для Фон Неймановской машины состоит из набора команд, которые выполняются одна за другой.

Адрес очередной ячейки памяти, из которой следует брать команду, указывается счётчиком команд. Следовательно, области памяти могут быть проименованны, и, к заполненным в них значениям, можно впоследствии обращаться, менять их во время выполнения программы с использованием присвоенных имён.

Переменная – центральное понятие языков высокого уровня.

Данные, с которыми работает программа, могут включать в себя переменные.

Основной цикл выполнения (цикл Фон Неймана) состоит из следующих шагов:

1. Получить адрес результата или операндов
2. Получить значения операнда (оттуда, где он расположен)
3. Вычислить значения результата из значения операндов
4. Сохранить значения результата (по его адресу)

Императивные языки программирования были разработаны после Фон Неймановской машины, чтобы абстрагироваться от компонентов и действий машины, чтобы облегчить программирование вычислений.

Абстрагироваться – отбрасывание ненужного и выделение чего-то существенного (нужного для нас в данный момент)

Абстракция – существенные характеристики некого объекта, которые отличают его от всех других, или наоборот, которые объединяют его с другими. В процессе абстрагирования мы имеем скорее обобщённую модель, а не в деталях.

1. Первые языки программирования были чисто императивными (машинные коды, языки ассемблера, ранние высокоуровневые языки (типо FORTRAN))

Сейчас эту парадигму поддерживают Python, Pascal, С++, Java и т.д.

1. Структурное программирование (Дейкстра, Николос Вирт)

Были введены понятия, объединяющие часто используемые шаблоны написания императивного кода.

Шаблоны, позволяющие избежать составления программы по типу «блюдо спагетти» Например, понятие составной инструкции/оператора/блока; инструкции ветвления и цикла. Благодаря этому можно отказаться от оператора go to (безусловный переход)

Программа типа «Блюдо спагетти»

1. Процедурное программирование

Растёт сложность программ, появилось понятие процедуры, появилась новая методология. Исходная задача разбивается на меньшие (с помощью процедур), пока решение всех задач не становится тривиальным (понятным и простым)

Процедуры – самостоятельный участок кода, который можно выполнить как одну инструкцию (как один оператор) Связанные понятия – вызов, аргументы (параметры), возврат, рекурсия, перегрузка и т.д. (написать определения понятий)

1. Модульное программирование

После него появились классы

Модуль – функционально законченный фрагмент программы, оформленный в виде отдельного файла с исходным кодом.

Класс – это шаблон кода, по которому создаётся какой-то объект.

1. Объектно-ориентированная парадигма (ООП)

Программы состоят из объектов, каждый из которых принадлежит определённому классу, которые организованны на принципах наследования, инкапсуляции, полиморфизма. Объекты взаимодействуют между собой в программе, посылают друг другу сообщения, рождаются и умирают.

С++ – ООП. Он развивался вместе с парадигмами