Programming Methodology

By:
Anh Duong

Programming Methodology

By:
Anh Duong

Online:
< http://cnx.org/content/col10450/1.1/ >

C O N N E X I O N S

Rice University, Houston, Texas

Table of Contents

Chapter 1

Introduction

1.1 PC Architecture

1.1.1 computer architecture[1]

1.1.1.1 Computer Hardware

A computer may be divided into six logical units.

Input Unit

- Obtain information from input devices: keyboards and mouse devices.
- Place the information at the disposal of the other units to be processed.

Output Unit

- Take information that has been processed.
- Place it on output devices: displayed on screens, printed on paper.

Memory Unit

- RAM (random access memory) is volatile, stores program and data.
- ROM (read only memory) is non-volatile, contains fundamental instructions.

Arithmetic and Logic Unit (ALU)

- Perform all the arithmetic and logic operations: addition, subtraction, comparison, etc..

CPU

- Tell the input unit when information should be read into the memory unit.
- Tell the ALU when information from the memory should be used in calculations.
- Tell the output unit when to send information from the memory unit to certain output devices.

Secondary Storage.

- Permanent storage areas for programs and data: magnetic tapes, magnetic hard disks, floppy disk, CD ROM

---

[1]This content is available online at <http://cnx.org/content/m15039/1.1/>.

1.1.1.2 Computer Software

- A computer program: set of instructions used to operate a computer to produce a specific result.
- Computer programming: writing computer programs.
- Programming languages: languages used to create computer programs.

Machine Languages
  Example: 0101010 000000000001 000000000010

- The lowest level of computer languages.
- Programs consist of entirely of 1s and 0s.
- Programs can control directly to the computer's hardware.
- Machine language instructions consist of two parts:

  · Instruction part (opcode) is the leftmost group of bits and tells the computer the operation to be performed.
  · Address part specifies the memory address of the data to be used in the instruction.

Assembly Languages
  Example:
  LOAD BASEPAY
  ADD OVERPAY
  STORE GROSSPAY

- Perform the same tasks as machine languages, but use symbolic names for opcodes and operands.
- An assembly language program must be translated into a machine language program.

Translation program (assembler)Machine languageprogramAssembly languageprogram

- Machine languages and assembly languages are called low-level languages since they are closest to computer hardware.

High-level Programming Languages

- Create computer programs using instructions that much easier to understand: English-like included with mathematical notations.
- Programs written in high-level languages must be translated into a low level language using a program called a compiler.
- Each line in a high-level language program is called a statement.

Ex: Result = (First + Second)*Third.
  Application and System Software

- Application software: perform particular tasks required by the users.
- System software: must be available to any computer system to operate. The most important system software is the operating system (MS-DOS, UNIX, MS WINDOWS, MS WINDOWS NT)

- Multitasking systems: operating systems allow user to run multiple programs.

Chapter 2

Basic C++

Index of Keywords and Terms

Keywords are listed by the section with that keyword (page numbers are in parentheses). Keywords do not necessarily appear in the text of the page. They are merely associated with that section. *Ex.* apples, § 1.1 (1) **Terms** are referenced by the page they appear on. *Ex.* apples, 1

Attributions

Programming Methodology

This course provides students with methodology in programming, using C++ in almost examples.

About Connexions

Since 1999, Connexions has been pioneering a global system where anyone can create course materials and make them fully accessible and easily reusable free of charge. We are a Web-based authoring, teaching and learning environment open to anyone interested in education, including students, teachers, professors and lifelong learners. We connect ideas and facilitate educational communities.

Connexions's modular, interactive courses are in use worldwide by universities, community colleges, K-12 schools, distance learners, and lifelong learners. Connexions materials are in many languages, including English, Spanish, Chinese, Japanese, Italian, Vietnamese, French, Portuguese, and Thai. Connexions is part of an exciting new information distribution system that allows for **Print on Demand Books**. Connexions has partnered with innovative on-demand publisher QOOP to accelerate the delivery of printed course materials and textbooks into classrooms worldwide at lower prices than traditional academic publishers.