CS 150
Introduction to Computer Science 1

Professor: Chadd Williams
chadd@pacificu.edu
What is CS150?

- CS150 is a programming course
- You will learn
  - The mechanics of writing programs in C++
  - How to solve complex problems using C++
  - How to break a large problem into smaller, more manageable problems
  - How to formulate algorithms to solve problems
- You do not need any previous programming or computer skills to take this course
How to Succeed in CS150

• Don’t miss class. It is very difficult to pick up any material that you miss

• Try and read ahead even if you don’t understand much

• Start programming assignments early

• Do as much on your own as possible. The more help you get the less sure of yourself you will become
How to Succeed in CS150

- Read the assignments carefully and follow all directions
- See me as soon as possible about any in class information that you are unclear on
- Attack the computer, you can’t hurt a thing!
Course Schedule

- The course schedule I have given you is tentative. I expect to follow this schedule, but I may have to adjust it from time to time.
- The online schedule will be accurate and up to date. That is the schedule that you should refer to when studying or revising.
Introduction to Computers and Programming

Chapter 1
Topics

• What are computers?
• A little bit of history
• Computer basics
• Programming languages
What is a Computer?

- What is your definition?

- The most important thing to remember is that a computer is a machine that follows directions. In the case of programming, the machine is following your directions exactly.

- You need to be very specific about what you want the computer to do.
History

- First electronic digital computer
  - Late 1930’s at Iowa State
  - Dr. John Atanasoff and Clifford Berry
  - Mathematical computations for nuclear physics

- First large-scale, general purpose computer
  - ENIAC in 1946 at U. Penn. for US Army
  - J. Presper Eckert and John Mauchley
  - Weighed 30 tons and occupied 1500 sq. ft.
  - Cost $500,000 to develop and build
  - Used for calculating ballistics tables, predicting weather and making atomic energy calculations
Picture of ENIAC
Von Neumann Architecture

- Dr. John von Neumann proposed the concept of a stored-program computer
- In ENIAC data is stored in memory, so why not a program
- The von Neumann architecture is the basis of the digital computers we know today
Today

- Most of us use microcomputers
  - First developed in 70’s
  - Small processor
  - Mac’s and PC’s are examples
Hardware

- Physical components of a computer
  - Central Processing Unit (CPU)
  - Main Memory (RAM)
  - Secondary Storage
  - Input Devices
  - Output Devices

- Let’s look at each of these in detail
CPU

- A CPU
  - Fetches instructions
  - Follows instructions
  - Produces results

- A CPU consists of
  - Control unit: coordinates computer operations
  - ALU: performs arithmetic operations
Memory

<table>
<thead>
<tr>
<th>Address</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-27.2</td>
</tr>
<tr>
<td>1</td>
<td>354</td>
</tr>
<tr>
<td>2</td>
<td>0.05</td>
</tr>
<tr>
<td>3</td>
<td>-26</td>
</tr>
<tr>
<td>4</td>
<td>H</td>
</tr>
<tr>
<td>5</td>
<td>400</td>
</tr>
<tr>
<td>6</td>
<td>RTV 001</td>
</tr>
<tr>
<td>7</td>
<td>ADD 003</td>
</tr>
<tr>
<td>8</td>
<td>STO 005</td>
</tr>
<tr>
<td>9</td>
<td>X</td>
</tr>
<tr>
<td>10</td>
<td>1005</td>
</tr>
</tbody>
</table>

- Memory is a sequence of storage cells
- Memory cells are 1 byte in size
- Bytes are groups of bits (8 usually)
- Bits are 0 or 1
- Each memory cell has unique address
- Contents can be data or instruction
- Everything stored as strings of 0s & 1s
- RAM is volatile
Secondary Storage

- Not volatile

- Disk drives
  - Hard disks
  - Floppy disks
  - Zip disks

- Optical drives
  - CDs
  - DVDs
Input/Output Devices

- **Input:** sends information to the computer from outside
- **Output:** sends information from the computer to outside
- **Examples?**
Software

- Operating System
- Application Software
Question

• Can computers think?

• Computers need a list of instructions to perform operations

• These instructions are *programs*
Program

- Program
  - Set of instructions directing a computer to perform a task

- Programming language
  - A language used to write programs
  - Examples?
Programming Language

• Machine language
  - Zeroes and ones
  - Machine dependent

• High level language
  - Instructions look like everyday English
  - Each instruction can perform many machine language instructions
C++

- Based on the C programming language
- C++ is a high level programming language
- One of today’s most popular programming languages
- Used extensively in industry
Summary

• Today we have looked at:
  o The history of computers
  o The hardware of computers
  o The software of computers
  o Concept of programming

• Next time we will:
  o Start coding

• Completed sections 1.1 - 1.3 from the book
  o Pages 1-9