

# Week 0: Intro to Computers and Programming

Gaddis: Sections 1.1-3 and 2.1

CS 1428  
Fall 2014

Jill Seaman

1

## 1.1 Why Program?

Computer – programmable machine designed to follow instructions

Program – instructions in computer memory to make the computer do something

Programmer – person who writes instructions (programs) to make computer perform a task

SO, without programmers, no programs; without programs, a computer cannot do anything

2

## 1.2 Computer Systems: Hardware and Software

- Hardware:  
the physical components that a computer is made of.
- Software:  
the programs that run on a computer

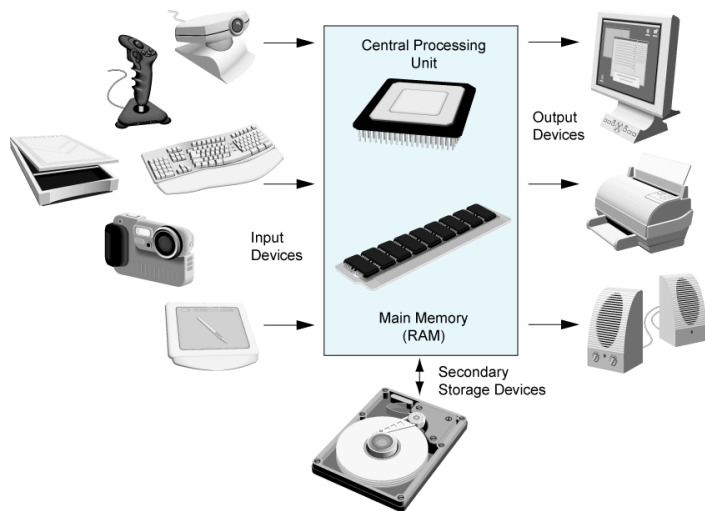
3

## Hardware Components

- **Central Processing Unit (CPU)**
- **Main Memory (RAM)**
- **Secondary Storage**
- **Input Devices**
- **Output Devices**

4

## Hardware Components Illustrated



5

## Central Processing Unit (CPU)

The part that actually RUNS the programs

Comprised of:

- Control Unit
  - Retrieves and decodes program instructions
  - Coordinates activities of all other parts of computer
- Arithmetic & Logic Unit (ALU)
  - Performs high-speed numeric calculations
  - Makes true/false decisions (comparisons)

6

## Main Memory

- Computer's working memory: programs must be stored in main memory when the program is running
- It is volatile: Main memory is erased when program terminates or computer is turned off
- Also called Random Access Memory (RAM)
- Addresses – Each location in memory is identified by a unique number known as an address.

7

## Main Memory

0	1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16 149	17	18	19
20	21	22	23 72	24	25	26	27	28	29

- The number 149 is stored in the location with the address 16, and the number 72 is stored at address 23.

(149 and 72 are actually represented in memory using binary: 0's and 1's)

8

## Secondary Storage

- Where documents and data (and programs) are stored.
- Non-volatile: data retained when program is not running or computer is turned off
- Comes in a variety of media:
  - ▶ magnetic: floppy disk, hard drive
  - ▶ optical: CD-ROM, DVD
  - ▶ Flash drives, connected to the USB port

9

## Input and Output Devices

- **Input Devices:** send information to the computer from outside
  - ▶ Keyboard, mouse, scanner, digital camera, microphone, touch screen/pad
  - ▶ Disk drives, CD drives, and DVD drives
- **Output Devices:** send information from the computer to the outside world
  - ▶ Monitors/screens, printers, speakers
  - ▶ Disk drives, CD drives, and DVD drives

10

## Software

- Programs that run on the hardware
- Operating Systems (System software):
  - programs that manage the computer hardware and the programs that run on them.
  - Unix, MS-DOS, Linux, Windows, Mac OS X
  - Time machine, printer drivers, compilers
- Application Programs (Apps):
  - Solve specific problems and provide services to the user
  - Word, Excel, iTunes, Firefox, Angry Birds, Photoshop

11

## 1.3 Programs and Programming Languages

- A program is a set of instructions that the computer follows to perform a task
- An algorithm:
  - ▶ A set of well-defined steps for performing a task or solving a problem.
  - ▶ A step by step ordered procedure that solves a problem in a finite number of precise steps.
- An algorithm can be in any language (English, C++, machine code, etc).

12

## Example (algorithm)

1. Display on screen: "how many hours did you work?"
2. Wait for user to enter number, store it in memory
3. Display on screen: "what is your pay rate (per hour)?"
4. Wait for user to enter rate, store it in memory
5. Multiply hours by rate, store result in memory
6. Display on screen: "you have earned \$xx.xx" where xx.xx is result of step 5.

**Note:** Computer does not speak English, it only understands its own "machine language"

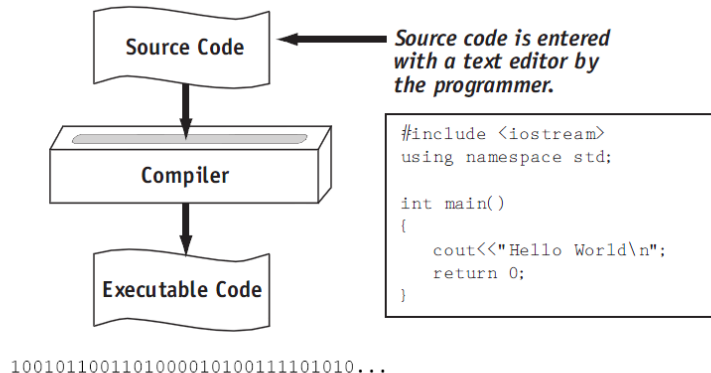
13

## Programming Languages

- Machine Language:
  - Instructions are encoded as a sequence of 1's and 0's
  - Machine specific
- Low Level Languages: Assembly Language
  - Letters and digits (codes)
  - Direct correspondence to Machine Language
- High Level Languages (like C++):
  - Words, symbols, numbers
  - Easier for humans to read and use
  - Must be translated to Machine Code

14

## Translation Process



10010110011010000101001111101010...

Tony Gaddis, Starting out with C++: From Control Structures Through Objects 7th ed.

15

## 2.1 The Parts of a C++ Program

```
// sample C++ program
#include <iostream>
using namespace std;

int main()
{
    cout << "Hello, world!";
    return 0;
}
```

16

## Parts of a C++ Program

- **Comment:** `//...`
  - ▶ ignored by compiler
  - ▶ notes to human reader
- **Preprocessor Directive:** `#include <iostream>`
  - ▶ compiler inserts contents of file `iostream` here
  - ▶ required because `cout` is defined in `iostream`
- `using namespace std;`
  - ▶ allows us to write `cout` instead of `std::cout`

17

## Parts of a C++ Program

- `int main ()`
  - start of function (group of stmts) named `main`
  - the starting point of the program
- `{}`
  - contains the body of the function
- `cout << "Hello, world!";`
  - statement that displays message on screen
- `return 0;`
  - sends value of 0 to OS (means success!)

18