COMP 110
Introduction to Programming

Fall 2015
Time: TR 9:30 – 10:45
Room: AR 121 (Hanes Art Center)

Jay Aikat
FB 314, aikat@cs.unc.edu

Previous Class

• What did we discuss?
Today

• Announcements
  • Still registering...
  • TA Office hours begin tomorrow (Fri, Aug 21)
  • See class webpage for TA office hours
  • Join Piazza:
piazza.com/unc/fall2015/comp110
  • Assignment1: due Fri, Aug 28 at 11:55 PM
  • Check Sakai, Piazza and class webpage regularly
• Computer and Programming basics
• Your first program

Software

• Java and Eclipse
  -- See the course website for detailed installation instructions
• How many got this installed?
Computers – not Magic


Hardware vs Software (concretely)

• Hardware
  – CPU, Memory, disks

• Software
  – Windows 7, Google Chrome, Games, Eclipse
  – Microsoft Word
Hardware vs Software (abstractly)

• Software
  – An organized collection of instructions

• Hardware
  – Circuits that execute, store and interact with instructions
    • Execution: CPU
    • Storage: Memory
    • Interaction: Peripherals, like keyboards, monitors, networks

Software Categories

• Operating System
  – first program to load when a computer is turned on
  – controls all machine activities
  – provides the user interface to the computer
  – manages computer resources, such as the CPU, memory, and hard drive
  – Examples?

• Application
  – generic term for any other kind of software
  – examples: word processors, missile control systems, games
Operating System (OS)

- OS monitors overall activity of the computer and provides services
- Written using programming language
- Example services:
  - memory management
  - input/output
  - storage management

Application Programs

- Written using programming languages
- Perform a specific task
- Run by the OS
- Example programs:
  - Browsers
  - Word Processors
  - Spreadsheets
  - Games
Questions

Classify the following pieces of software as operating system or application:
1. Microsoft Windows 8
2. Microsoft PowerPoint
3. Linux
4. Your COMP 110 programs

Instructions

• An instruction is a sequence of 0's and 1's that represents a single operation on the computer

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Data</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000101</td>
<td>00000001</td>
<td>00000010</td>
</tr>
<tr>
<td>Add</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
| What is the output? -- 3

• These 0's and 1's are called bits
  - Why only 0 and 1?
    - Because it is easier to make an electrical device that has only two stable states
A Computer

Memory

Motherboard (Highway)

CPU (Brain)

RAM (Short Term Memory)

Hard Drive (Long Term Storage)

DVD Drive (Removable Media)
CPU (Central Processing Unit)

• It is the “brain” of the computer
  – CPU executes the instructions
  – CPU’s working routine
    • read instructions and data from memory
    • do calculation
    • write calculation results back to memory

• Intel Core i7 3.4 GHz
  – Executes at most 3,400,000,000 instructions per second

Central Processing Unit (CPU)

• Control Unit (CU)
  – "the brain" of the CPU
• Program Counter (PC)
  – points to the next instruction to be executed
• Instruction Register (IR)
  – holds the currently executing instruction
• Arithmetic Logic Unit (ALU)
  – carries out all arithmetic and logical ops
• Accumulator (ACC)
  – holds the results of the operations performed by the ALU
Memory

- Holds instructions and data for the computer
  - How much the “brain” can remember
- Main Memory
  - For intermediate calculations (program you are running)
  - Disappears when you shut down your computer
- Secondary Memory
  - Hard drives, CDs, Flash drives
  - Exists until you delete it

GB? MB? KB?

- 1 bit = 0 or 1
- 1 byte = 8 bits
  - Smallest addressable unit of memory
- Kilo, Mega, Giga, Tera
  - 1 KB = 1,000 bytes (1 thousand bytes)
  - 1 MB = 1,000 KB = 1,000,000 bytes (1 million bytes)
  - 1 GB = 1,000 MB = 1,000,000,000 bytes (1 billion bytes)
    - The same for GHz (1 Giga Hertz)
  - 1 TB = 1,000 GB = 1,000,000,000,000 bytes!
Main Memory

- Memory address
  - To locate certain memory positions
  - CPU fetches data according to memory address
- Another interesting fact: characters are also saved in bits, and so is everything else
Main Memory with 100 Cells

Each memory cell has a numeric address, which uniquely identifies it.

CPU and Main Memory

All programs must be brought into main memory before execution.

Primary storage area for programs and data that are in active use (RAM).

Chip that executes program instructions.
Secondary Storage

Secondary memory devices provide long-term storage.

Information is moved between main memory and secondary memory as needed.

Secondary Storage

- Provides permanent storage for information
- Retains information even when power is off
- Examples of secondary storage:
  - Hard Disks
  - USB Drives
  - DVDs
  - CDs
  - Tapes
**Peripherals**

- **Input devices**
  - Keyboard, mouse, game controller......
  - When they get input, they save that at certain memory addresses

- **Output devices**
  - Monitor, speaker, printer......
  - They are projected to certain memory addresses
  - When CPU wants to output, it writes to those addresses

- **CPU sees everything as memory**

**Input/Output Devices**

I/O devices facilitate user interaction
Opening Notepad

- Use the mouse to select Notepad
- The CPU requests the Notepad application
- Notepad is loaded from the hard drive to main memory
- The CPU reads instructions from main memory and executes them one at a time
- Notepad is displayed on your monitor

Programs

- Set of instructions for a CPU to follow
  – Also known as software
- You will be writing programs
  – We will write one today
- Our programs will be in Java
Programming Languages

• Why do we need languages when we have instructions?
  – Too hard for humans to write bits directly


Programming Languages

• Different languages are good at different aspects
  - C/C++: close to instructions, runs fast
  - Matlab: good at scientific computation
  - Python: relatively easy, fast development

• We choose Java
  – Not because...
    - It’s the best language (there is no such thing!), or
    - It’s easiest to learn (not!)
  – But because
    - It’s widely used, incorporates (most) modern features
Programming languages

- **Java** - General purpose. Best for writing larger programs.
- **Javascript** - No relationship to Java! Very specialized. Runs in your web browser and adds advanced behavior to web pages.
- **Python** - General purpose but a scripting language. Much easier to write small programs (compared to Java), but much less appropriate for larger ones.
- **Matlab** - Very different from the other 3 languages. Very powerful but highly specialized. Excellent for solving equations, graphing data, etc. since much less programming is required.

From Languages to Instructions

- The translator is called a **compiler**
  - It is also a program
  - From human-readable to machine-readable
From Java to Machine Language

- Computers understand only 0 and 1 (*machine language*)
- *Compiler* translates source code into machine code

- Java *compiler* translates source code (file ending in *java*) into *bytecode* (file ending in *class*)
  - bytecode is portable (not machine-specific)
- Java *interpreter* reads and executes bytecode
  - different Java interpreters for different types of CPUs and operating systems (OS)
    - Intel/Windows, Motorola/Mac OS X, Intel/Linux

Programming Languages

- Programming languages have rules of grammar just as English does

  - **syntax rules** - which statements are legal and which are not
  
  - **semantic rules** - determine the meaning of the instructions

- **token** - smallest individual unit of a program
  - special symbols
  - word symbols
  - identifiers
Special Symbols

+    -    *    /

.    ;    ?    ,

<=   !=   ==   >=

Word Symbols (aka reserved words or keywords)

• int
• float
• double
• char
• void
• public
• static
• throws
• return

• reserved words are always all lowercase
• each word symbol is considered to be a single symbol
• cannot be used for anything other than their intended purpose in a program
Next class (Tue, Aug 25)

• More Programming Basics
• Program in class: Hello World!
• Assignment0 DUE Fri, Aug 28
→ Reading Assignment: Chapter 1