News

- reminder: no class next time: this Friday Jan 8!
- UBC CS news

Correction / Recap:

- Prerequisites:
  - Math 12 is the prerequisite
  - If you have not taken it, you will be dropped from the course
  - You cannot get credit for 111 if you take 101 after or concurrently with 111

Recap: Processes, Procedures, and Programs

- process: what happens when a computer follows a procedure - it's a procedure in execution

- procedure: collection of instructions in some meaningful order that results in useful behavior on behalf of the device that executes the instructions

- program: when instructions are written in symbolic language that can be executed by a computer

Recap: Procedures and Algorithms

Here's why we get frustrated when we start to learn to write programs to make computers do stuff:

An algorithm is:
- a finite procedure
- written in a fixed symbolic vocabulary
- governed by precise instructions
- moving in discrete steps, 1, 2, 3, ...
- whose execution requires no insight, cleverness, intuition, intelligence, or perspicuity
- and that sooner or later comes to an end

We don't have a lot of practice at being stupid!

Why Being Precise/Stupid Isn't Easy

- human languages are very different from computer languages: they're ambiguous
  - Humans bring huge amounts of knowledge to understanding meaning of sentence
  - We apply it automatically and unconsciously
  - Many meanings per word
  - Sentence structure
  - Context of conversation
  - How the world works
  - How language is used
  - You count on listener to disambiguate without even noticing
  - We can get away with relatively short and imprecise sentences

Physical Hardware

- "Computer science is how to harness the physical world to help us think." - Alan Hu

- Harnessing the physical world to help us think:
  - How to get things that have computational behavior?
  - Technology dependent:
    - Sticks, gears, relays, vacuum tubes, transistors, DNA, ...
    - How to control that behavior to do interesting things

Computer Design

- It's hard to figure out how to make things do computation
- All digital computers for over 50 years have had:
  - Same basic organization
  - Binary representation of data
  - Numerically addressed memory
  - Fetch-decode-execute operation cycle
- We'll only have a brief glance here

Binary Data Representation

- All programs and data on a computer are represented using only symbols 0 and 1
- This simple binary system is encoded in all of our digital hardware devices:
  - Magnetic disks: magnetic material can be polarized to one of two extremes (north or south) to represent 0 or 1
  - Memory: each byte consists of 8 bits; each bit is a kind of electronic switch that is either off or on representing 0 or 1

Memory

- Some of computer programming is resource management
- As beginning programmers, the resource that you'll be concerned with most is memory
- Most programming languages do a lot of the work for you
  - More on this soon
Memory
- Memory consists of a series of locations, each having a unique address, that are used to store programs and data.
- When data is stored in a memory location, the data that was previously stored there is overwritten and destroyed.
- Each memory location stores one byte (or 8 bits) of data.
  - More on this soon

Units of Memory Storage
We measure units of memory in terms of bytes:

<table>
<thead>
<tr>
<th>Unit</th>
<th>Symbol</th>
<th># of bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>byte (8 bits)</td>
<td>2^8 = 1</td>
<td></td>
</tr>
<tr>
<td>kilobyte (KB)</td>
<td>2^10 = 1024</td>
<td></td>
</tr>
<tr>
<td>megabyte (MB)</td>
<td>2^20 = 1024^2</td>
<td></td>
</tr>
<tr>
<td>gigabyte (GB)</td>
<td>2^30 = 1024^3</td>
<td></td>
</tr>
<tr>
<td>terabyte (TB)</td>
<td>2^40 = 1024^4</td>
<td></td>
</tr>
</tbody>
</table>

What Can Be Represented By A Byte?
- 256 different characters from your keyboard
- Java actually uses 2 bytes to represent a character
- how many characters is that?
- 256 different shades of gray in a black and white image
- 256 colors or shades of color in a color image
- 256 frequencies or tones to be played through a speaker
- 256 of anything that can be represented as discrete entities
  - part of an instruction for a computer

What is Computer Programming?
Central processing unit
- CPU executes instructions in a continuous cycle
  - known as the “fetch-decode-execute” cycle
- CPU has dedicated memory locations known as registers
  - One register, the program counter, stores the address in memory of the next instruction to be executed

Memory
- Data values are stored in memory locations – more than one location may be used if the data is large.

Mass storage/long-term memory
- Macintosh SE in 1987
- 1 megabyte (MB) of memory
- MacBook Pro in 2008
- 2 gigabytes (GB) of memory
- 1000 times more memory capacity in 20 years
- 1000 times greater processing speed
- Approximately the same price

What Can Be Represented By A Byte?
- 256 of anything that can be represented as discrete entities
- part of an instruction for a computer

Central processing unit
- CPU executes instructions in a continuous cycle
  - known as the “fetch-decode-execute” cycle
- CPU has dedicated memory locations known as registers
  - One register, the program counter, stores the address in memory of the next instruction to be executed

Instructions
- Implication of the fetch-decode-execute cycle
  - we control the computer to make it do what we want by giving it a sequence of little steps for it to do
  - these steps are the instructions in a programming language

Programming Languages
- Objectives
  - Understand what is meant by computer programming.
  - Understand the difference between machine/assembly language and a high-level computer language.
  - Understand what compilers and interpreters are, and why we use them.
  - Write, compile, and run a simple Java program.

What is a Computer?
- How is a computer different from a video game console? Or a DVD player? Or a telephone? Or a bank machine?
- The computer is general. It can be all of the other devices.
- Making the computer do what we want is called programming the computer.

What is Computer Programming?
- How is a computer different from a video game console? Or a DVD player? Or a telephone? Or a bank machine?

Computer Programming
- Input Devices
- Output Devices
- Central Processing Unit
- Memory
- Mass Storage Devices
Computer Programming

- You can make the computer do anything that it's capable of. The only limits are space, time, I/O devices, and your skill and creativity.
- It takes work:
  - The biggest program you'll write in 111 will be a few hundred lines long.
  - Windows XP is 40 million lines long.
- You have to write in a language the computer understands.

George and Stephen go to France

- George is American. He knows only English.
- Stephen is Canadian. He is bilingual in English and French.
- How can George communicate in France?

How can George communicate in France?

1. If he wants to communicate quickly, then Stephen can interpret — translating French to English and English to French on-the-fly.
2. If there's a lot of stuff to translate (e.g., a speech, or a long document), then Stephen can translate the whole thing at once. Now, George can read it whenever he wants.

Aside – Binary vs. Decimal Numbers

- We are used to representing numbers in the decimal system
  - have digits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  - For example:
    - 4763
- Means
  - $3 \cdot 10^3 + 7 \cdot 10^2 + 6 \cdot 10^1 + 3 \cdot 10^0$
- Note how the exponents count up from 0!

Aside – Binary vs. Decimal Numbers

- With binary numbers, we use 2 as the base
  - have digits 0, 1
  - Example: 10010111
    - $2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$
    - $256 + 64 + 32 + 16 + 8 + 4 + 2 + 1$
    - We have just converted a binary number to decimal
    - more details, decimal to binary conversion in CPSC 121

Health Education in Remote Areas

- In remote areas of the world, there are languages spoken by small groups of people, and also a national language spoken by the mainstream, e.g.:
  - Many native languages vs. Spanish in Latin America
  - Minority languages vs. Mandarin in China
  - Regional languages vs. Hindi or English in India
- How do you provide health info (e.g., in English) to the isolated (e.g., in Latin America)?

Assembly Language

- Assembly language program converted into corresponding machine language instructions by another program called an assembler

Machine Language

- First programming languages: machine languages
  - Most primitive kind
- Sample machine language instruction
  - add r1, r2, r6
    - 00000000000101010011000000100000
      - in this register
      - add what is in this register
      - in this register
      - to what
      - in this register
      - in this register
      - in this register
      - in this register
      - in this register
      - in this register
      - in this register
      - in this register
      - in this register
      - in this register
  - Difficult to write programs this way
  - People created languages that were more readable

Aside – Other Bases

- The same principle works for other bases
  - For example, hexadecimal (base 16)
    - uses digits 0, 1, 2, 3, 4, 5, 6, 7, 8, A, B, C, D, E, F
    - A-F correspond to values 10-15
  - Example: C350
    - $15 \cdot 16^2 + 3 \cdot 16^1 + 5 \cdot 16^0$
    - $5120 + 480 + 5 = 5605$
    - $5 \cdot 16^2 + 3 \cdot 256 + 12 \cdot 16^0$
Assembly Language
- Both machine and assembly languages pose big challenges for programmers
  - Difficult to read and write
  - Difficult to remember
- Each instruction does very little
  - Takes lots of instructions just to get something simple done
- Every machine or assembly language good for only one type of computer
  - Different to program IBM than Honeywell than Burroughs...

High-Level Language
- Next step: development of high-level languages
  - You may have heard of some
    - Fortran, COBOL, Lisp, BASIC, C, C++, C#...
    - Ada, Perl, Java, Python, Ruby, Javascript
  - High-level languages intended to be easier to use
    - still a long way from English.
  - A single high-level instruction gets more work done than a machine or assembly language instruction.
  - Most high-level languages can be used on different computers

High-Level Language
- Must be translated into machine language so the computer can understand it.
  - High-level instruction: A = B + C becomes at least four machine language instructions!
  - How?
    - You could translate it as you go (interpreter).
    - You could translate it in advance (compiler).

Interpreters and Compilers
- An interpreter translates the high-level language into machine language on-the-fly, executing the instructions as it goes.
  - A compiler translates the high-level language program all at once in advance.
  - Both compilers and interpreters are themselves computer programs.
  - Which is better?
    - Remember George and Stephen in France?

Java
- Java is the high-level language we’ll use.
  - Modern, widely used, portable, safe.
  - Developed by Sun in early 1990s
    - Originally intended for set-top boxes
    - Retargeted for the Web

Java Does Both!
Recursive toString method
Method String Object
- "{ "name": "World", "counter": 0 }"