A Computer

We need:
- Input Devices
- Output Devices
- Memory and Storage

All controlled by the CPU:
- Datapath
- Control

Introduction

- Rapidly changing field:
  - vacuum tube -> transistor -> IC -> VLSI (see section 1.4)
  - doubling every 1.5 years: 
    memory capacity
    processor speed  *(Due to advances in technology and organization)*
- Things you'll be learning:
  - how computers work, a basic foundation
  - how to analyze their performance (or how not to!)
  - issues affecting modern processors (caches, pipelines)
- Why learn this stuff?
  - you want to call yourself a "computer scientist"
  - you want to build software people use (need performance)
  - you need to make a purchasing decision or offer “expert” advice

Abstraction

Examine only the essential details.

The same object can be examined at different levels of detail.
*But usually one level at a time.*

Essential to understanding any complex thing.
*Computers in our case.*
Abstraction: Web Pages

- Web Page
- HTML
- Characters
- Code numbers
- Bits
- Electronic signals

Abstraction: Computer Programs

- High-Level Code: \( m = m + 5; \)
- Assembler Code:
  
  \[
  \begin{align*}
  \text{lw} &\quad \text{$t1$, $12($sp$)} \\
  \text{addi} &\quad \text{$t1$, $t1$, $5$} \\
  \text{sw} &\quad \text{$t1$, $12($sp$)}
  \end{align*}
  \]
- Binary Coding:
  
  - 0x8fa9000c
  - 0x21290005
  - 0xfa9000c

How Are Instructions Coded?

Part of the CPU Design.

Each CPU model is different. 

So you can’t run your Mac programs on your PC

Determines:

- What instructions are available.
- Code numbers for each operations.
- Size and format of each instruction.
- Representation and coding of each argument.

Instruction Set Architecture

- A very important abstraction
  - interface between hardware and low-level software
  - standardizes instructions, machine language bit patterns, etc.
  - advantage: different implementations of the same architecture
  - disadvantage: sometimes prevents using new innovations

  True or False: Binary compatibility is extraordinarily important?

- Modern instruction set architectures:
  - 80x86/Pentium/K6, PowerPC, DEC Alpha, MIPS, SPARC, HP
Where we are headed

- Performance issues (Chapter 2) \textit{vocabulary and motivation}
- A specific instruction set architecture (Chapter 3)
- Arithmetic and how to build an ALU (Chapter 4)
- Constructing a processor to execute our instructions (Chapter 5)
- Pipelining to improve performance (Chapter 6)
- Memory: caches and virtual memory (Chapter 7)
- I/O (Chapter 8)

Key to a good grade: reading the book!