

# Introduction to Computer Science and Programming in C

Session 2: September 4, 2008

Columbia University

# Review

- Went over syllabus
- Algorithms - systematic methods to solve problem.
  - Embarrassing addition example
- Characteristics of C: high-level, compiled, etc.

# Today

- Very brief history of computers
- Basic Architecture of the Modern Computer
- Cunix tutorial

# Links

- <http://www.columbia.edu/acis/history/>

# Early Computers

- **Analog** computers.
- vs. **digital**

# Early Digital

- First half of 20th century, punch cards.  
<http://en.wikipedia.org/wiki/Image:Punch-card-blue.jpg>
- Relays
- Vacuum tubes
- Programming: physically rearrange wires.

# Modern Computers

- Stored-program
- von Neuman Architecture
- Magnetic storage, optical storage, etc.
  - non-volatile

# Modern Computers

- Volatile memory (vs. non-volatile)
  - **Random Access Memory (RAM)**
  - Faster, more expensive
- **Central Processing Unit (processor)**
- **Input/Output (I/O)**



# Modern Computers

- **Operating systems (OS)** manage for us
  - Unix, Linux, DOS, Windows, Mac OS, etc.
- Programs access disk, RAM and I/O through OS
- Virtual memory

# Binary Representation

- **Binary** - taking two values
- **bit** = 0 or 1
- **byte** = 8 bits
- kilobyte, megabyte, gigabyte, terabyte

# Binary Representation

- Using one bit, we can represent true / false.
- Using one byte, how to represent the numbers 0 through 10?
  - Base-2
  - Addition and multiplication still work!
  - We can represent  $2^8$  values (counting 0)

# Binary Representation

- [0,255]
- or [-127,127]
- What about characters?  
a-z, A-Z, 0-9, punctuation...
- American Standard Code for Information Interchange (ASCII)