CSC 103: How Computers Work

Spring 2016
Smith College
Prof. Sheehan
Class 3: March 28
Outline

• Review of terminology

• Begin: parts of a computer

• More practice with logic gates and truth tables
Boolean algebra

- Named after George Boole
Boolean algebra

- Named after George Boole

- A “boolean” variable is either true (1) or false (0)

George Boole (1815-1864)
Credit: wikipedia
Boolean algebra

- Named after George Boole

- A “boolean” variable is either true (1) or false (0)

- Boolean logic and algebra is a way to describe functions and computation involving boolean variables

George Boole (1815-1864)
Credit: wikipedia
Boolean algebra

• Named after George Boole

• A “boolean” variable is either **true** (1) or **false** (0)

• Boolean logic and algebra is a way to describe functions and computation involving boolean variables

• Example:

\[ f(a, b, c, d) = (a \text{ and } b) \text{ or } (c \text{ and } d) \]
Binary numbers and bases

- Computer stores *everything* as binary numbers
  - Text data (Word documents, etc)
  - Pictures, video, audio
  - Instructions for programing
Binary numbers and bases

- Computer stores *everything* as binary numbers
  - Text data (Word documents, etc)
  - Pictures, video, audio
  - Instructions for programming

- Anything we can do in decimal, we can do in binary!
Binary numbers and bases

- Computer stores *everything* as binary numbers
  - Text data (Word documents, etc)
  - Pictures, video, audio
  - Instructions for programming

- Anything we can do in decimal, we can do in binary!

- “Special” bases:
  - **Binary**: base 2  
    - digits: **01**
  - **Octal**: base 8  
    - digits: **01234567**
  - **Decimal**: base 10  
    - digits: **0123456789**
  - **Hexadecimal**: base 16  
    - digits: **0123456789ABCDEF**
Logic gates

NOT

AND

OR

Image Credit: http://www.sectorfej.net/
Logic gates

\[ \bar{A} \quad A \cdot B \quad A + B \]
Logic gates

- All operations can be built from these 3 gates

\[
\begin{align*}
\bar{A} & \quad A \cdot B & \quad A + B
\end{align*}
\]

Image Credit: http://www.sectorfej.net/
Logic gates

- All operations can be built from these 3 gates
- Boolean functions, logic gate circuits, and truth tables are all equivalent!
Other logic gates

**NAND:** \( \text{not}(A \text{ and } B) \)
Other logic gates

**NAND**: not(A and B)

**NOR**: not(A or B)
Other logic gates

**NAND**: \( \neg (A \text{ and } B) \)

**NOR**: \( \neg (A \text{ or } B) \)

**XOR**: \( A \text{ xor } B \)
Other logic gates

NAND: $\text{not}(A \text{ and } B)$

NOR: $\text{not}(A \text{ or } B)$

XOR: $A \text{ xor } B$

XNOR: $\text{not}(A \text{ xor } B)$
Q: How do these gates work in a computer?
Enter: Claude Shannon

- Demonstrated that Boolean algebra could be encoded using electrical switches.

- Through this mechanism, any logical computations could be performed.

Claude Shannon (1916-2001)
Credit: wikipedia
How computers do logic gates

(1)
How computers do logic gates

NOT

Image Credit: Dominique Thiebaut
How computers do logic gates

(1) NOT

(2)
How computers do logic gates

(1) \[ \text{NOT} \]

(2) \[ \text{AND} \]
How computers do logic gates

(1) NOT

(2) AND

(3)
How computers do logic gates

1. NOT
2. AND
3. OR

Image Credit: Dominique Thiebaut
Transistors

• These switches form a model for **transistors**

• Transistors are the building blocks of **integrated circuits (ICs)**

• Integrated circuits can have billions of transistors in the size of a dime!

• Transistors can also have other functions, such as amplification

• Use **semiconductor** material
Integrated Circuit

Image Credit: Dominique Thiebaut
Apollo guidance computer

- Developed for the Apollo program in the 1960s
- First computer to use integrated circuits
- 4,100 ICs, each with a single NOR gate

Credit: wikipedia