CS 10 Review (Basics)

Wagner Truppel
Lecturer, Dept. of Computer Science & Engineering
UC Riverside

wagner@cs.ucr.edu
http://www.cs.ucr.edu/~wagner

http://www.cs.ucr.edu/cs12

Today's Topics

- Number Systems
- Computer Organization
- Computer Programming

Number Systems

- The difference between a **quantity** and its **numerical representation**
- Eleven dots: ••••••••••
- 11 ?
- I say 32…
- Or, perhaps, 23…
- Or, if you prefer, 21…
- But *also* 11.
- How come ?

Number Systems

- The difference between a **quantity** and its **numerical representation**
- Eleven dots: ••••••••••
- 3 ••• and 2 • ⇨ 32_3
- 2 •••• and 3 • ⇨ 23_4
- 2 ••••• and 1 • ⇨ 21_5
- 1 •••••••••• and 1 • ⇨ 11_{10}
- 32_3, 23_4, 21_5, and 11_{10} are different numerical representations of the **same** quantity

Number Systems

- Eleven dots: ••••••••••
- 2 ••••• and 1 • ⇨ 21_5
- 1 •••••••••• and 1 • ⇨ 11_{10}
- The subscripted numbers are called the **base** of the number system in question
- •••••••••• = 21 in base 5 = 11 in base 10
- Base b uses ***only*** the digits 0, 1, 2, …, (b-1)
- Examples:
  Base 10 uses 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  Base 3 uses 0, 1, 2

Number Systems

- Example: Base 3
  ⇨ 0_3
  • ⇨ 1_3
  •• ⇨ 2_3
  ••• ⇨ 10_3
  ••• • ⇨ 11_3
  ••• •• ⇨ 12_3
  ••• ••• ⇨ 20_3
  ••• ••• • ⇨ 21_3
  ••• ••• •• ⇨ 22_3
  ••• ••• ••• ⇨ ??

Number Systems

- Example: Base 3
  ••• ••• ••• ⇨ ?

Number Systems

- Example: Base 3
  ••• ••• ••• ⇨ 30_3 ?

Number Systems

- Example: Base 3
  ••• ••• ••• ⇨ 30_3 ? **No !**

Number Systems

- Example: Base 3
  ••• ••• ••• ⇨ 30_3 ? **No !**
  ••••••••• = 1 group of nine
  = 1 group of 3 squared
  = $\mathbf{100_3}$
- Why ***3 squared***?
- Just like 100_{10} = 1 group of **10** ***squared***
- Just like 1000_{10} = 1 group of **10** ***cubed***

Number Systems

- 21843_{10} =
  $3 \times 10^0 + 4 \times 10^1 + 8 \times 10^2 +$
  $1 \times 10^3 + 2 \times 10^4$
- 21201_3 =
  $1 \times 3^0 + 0 \times 3^1 + 2 \times 3^2 +$
  $1 \times 3^3 + 2 \times 3^4 =$
  $1_{10} + 0_{10} + 18_{10} + 27_{10} + 162_{10}$
  $= 208_{10}$

Number Systems

- 2 1 8 4 3_b
  b^0 units
  b^1 units
  b^2 units
  b^3 units
  b^4 units
- Remember to read from **right to left** !
- Remember that the **exponents start at 0** !

Number Systems

- **Important: Base 2 (Binary System)**

	⇨	0000_2	⇨ 0_{10}
•	⇨	0001_2	⇨ 1_{10}
••	⇨	0010_2	⇨ 2_{10}
•• •	⇨	0011_2	⇨ 3_{10}
••••	⇨	0100_2	⇨ 4_{10}
•••• •	⇨	0101_2	⇨ 5_{10}
•••• ••	⇨	0110_2	⇨ 6_{10}
•••• •• •	⇨	0111_2	⇨ 7_{10}
••••••••	⇨	1000_2	⇨ 8_{10}

Number Systems

- The digits 0_2 and 1_2 are called **B**inary dig**its** or **bits**
- 8 bits form one **byte**
- Why should we care about **base 2** ?
- Because:
  - Electricity **not** flowing ⇨ 0
  - Electricity flowing ⇨ 1
- Thus, we can represent numbers with electrical devices
- And that's what an electronic computer is !

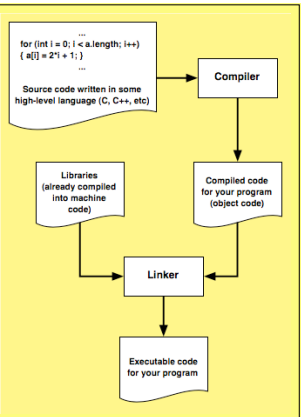
Computer Organization

- Ok, there's more to a computer than just that…
  - Memory
  - ALU (**A**rithmetic and **L**ogical **U**nit)
  - CPU (**C**entral **P**rocessing **U**nit)
  - Clock
  - I/O devices
  - And more…

Computer Programming

- What *really* happens when you write and compile a computer program ?

Computer Programming

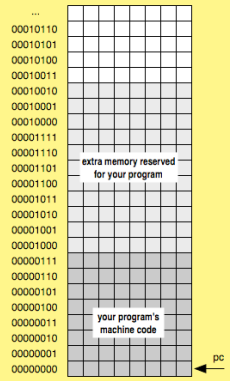


Computer Programming

- Where does your program reside ?
  - When it's not running, it's stored in the hard disk
  - When it's running, it's stored in memory
  - Why ?
- What really happens when you run a computer program ?

Memory Organization

- Memory contents
- Addresses
- Program counter



Computer Programming

- What really happens when you run a computer program ?
- We'll continue to look into that question in the next few lectures…
- And you'll learn the complete story when you take *CS 61: Machine Organization and Assembly Language Programming*

Reminder

- Make sure to add yourself to the course mailing list
  (accessible from the CS 12 course web page)
- Make sure to read handout 1: **intro_comp_org.pdf**
  It has more material than what I covered here today
  (also accessible from the CS 12 course web page)

Reminder

- For next lecture, read handout 2: **stack_frames.pdf**
- Review **chapters 1, 2, 3, 4** of Savitch's book
- It's easy reading, since it's all CS 10 material