Welcome to recitation!

15-213: Introduction to Computer Systems
3rd Recitation, Sept. 10, 2012

Instructor: Adrian Trejo (atrejo)

Section H, 3:30p – 4:30p WEH5302

Outline

- General
- UNIX
- Integers
- IEEE Floating Point
- Datalab Tips
- Style
- Cheating
- Summary

General Stuff

- www.cs.cmu.edu/~213
- www.autolab.cs.cmu.edu

- Everything related to the course can be found on these two sites.

UNIX Basics

- In case you missed the boot camp:
- Use the shark machines!

- ssh/scp (on Mac/UNIX)
- PuTTY/Filezilla (on Windows)

- tar

- Learn to use an editor well (e.g. vim, emacs) and stick with it.

Integers

- Signed vs. Unsigned
- Two's complement representation
- Implicit casting between signed and unsigned

Floating Point

- Sign (one bit)
- Exponent (single precision: 8 bits; double precision: 11 bits)
- Fraction (Mantissa)

- Bias (2^(k-1) – 1, where k is the number of exponent bit)
- Normalized (E = Exp – Bias) vs. Denormalized (E = 1 – Bias)

- Special Values (Exp is all ones)

Floating Point (cont.)

- A favorite exam question:

Problem 2. (8 points):

Floating point encoding. Consider the following 5-bit floating point representation based on the IEEE floating point format. This format does not have a sign bit – it can only represent nonnegative numbers.

- There are $k = 3$ exponent bits. The exponent bias is 3.
- There are $n = 2$ fraction bits.

Recall that numeric values are encoded as a value of the form $V = M \times 2^E$, where E is the exponent after biasing, and M is the significand value. The fraction bits encode the significand value M using either a denormalized (exponent field 0) or a normalized representation (exponent field nonzero). The exponent E is given by $E = 1 - Bias$ for denormalized values and $E = e - Bias$ for normalized values, where e is the value of the exponent field `exp` interpreted as an unsigned number.

Below, you are given some decimal values, and your task it to encode them in floating point format. In addition, you should give the rounded value of the encoded floating point number. To get credit, you must give these as whole numbers (e.g., 17) or as fractions in reduced form (e.g., $3/4$). Any rounding of the significand is based on ***round-to-even***, which rounds an unrepresentable value that lies halfway between two representable values to the nearest even representable value.

Value	Floating Point Bits	Rounded value
$9/32$	`001 00`	1/4
2		
13		
$1/8$		
$15/4$		

Datalab Tips

- Signed negation
- -x == ~x + 1
- Always works except for x = Tmin

- Properties of Zero
- 0 & x = 0 for all x
- (0-1) & x = x for all x

- int x = 0
- int y = -x

- x and y are both positive since their MSBs is 0

Parity Example

- Let's write a function that takes an integer and returns 1 if it has an odd number of '1' bits, and 0 otherwise.

- How can we get the answer?
- If we XOR all the bits together, then we'll get the answer!

- 10011010 (function should return 0)
- 1 ^ 0 ^ 0 ^ 1 ^ 1 ^ 0 ^ 1 ^ 0 = 0

- 1001 ^ 1010 = 0011
- 00 ^ 11 = 11
- 1 ^ 1 = 0

Style

- http://www.cs.cmu.edu/~213/codeStyle.html
- Make sure you read through it since we'll use it as a rubric when we grade your labs.

- Look out for:
  - Comments
  - Magic numbers (without #define)
  - Line length
  - Consistency
  - etc.

Cheating policy

- Don’t do it!
- MOSS checker built into Autolab

Need help?

- Email: 15-213-staff@cs.cmu.edu
- Office hours: UMTWH 5:30p – 8:30p WeH 5207
- Recitation: bring us questions