Floating point & Datalab

15-213 (18-213): Introduction to Computer Systems
Recitation 3, Jan. 28 2013

Jimmy Guo, Section H

FAQ page

- http://www.cs.cmu.edu/~213/faq.html
- **General/assignment issues**
  - E.g. Why do I get “permission denied” error?
  - Read through before you ask questions
  - Check back for any future update

Outline

- **Integer**
- **IEEE floating point**
  - Overview
  - Examples
- **Datalab**
- **Questions**

Integers

- **Power-of-2 Multiply/Divide with shift**
  - Multiply
    - Left shift by k
  - Divide
    - Negative number needs bias
    - (x + (1<<k) – 1) >> k

- **Representation in memory**
  - Big Endian vs. Little Endian
  - Keep in mind when doing Bomblab

Floating point

- **Encoding**
  - Sign
    - Symmetric on axis (thus there are both +0 and -0)
  - Exponent
    - 3(normalized, denormalized, special) cases based on exp
    - More bits --> wider range
  - Fraction (Mantissa)
    - Implied leading 1 (for normalized number)
    - More bits --> higher precision

Floating point

- **Bias**
  - Bias = $2^{k-1} - 1$, where k is number of exponent bits

- **Normalized *vs.* Denormalized**
  - exp?
  - Implied leading 1 *vs.* Implied leading 0
  - E = exp – Bias *vs.* E = 1 – Bias
  - Denser nearer the origin *vs.* evenly spaced

- **Special case (exp = 111…1)**
  - Infinity
  - NaN

Floating point

- **Rounding**
  - Round to even
  - Why?
    - Avoid statistical bias
  - How?
    - 1.1011 (All round to nearest 1/4)
    - 1.1010
    - 1.0101
    - 1.1110

- **Addition & Multiplication**
  - Associativity/distributivity may not hold
    - 3.14 + (1e20 - 1e20) *vs.* (3.14 + 1e20) - 1e20

Floating point (Examples)

1.a 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.**
- **There are n = 2 fraction bits.**

What is the…

- Bias?
- The largest denormalized number?
- The smalleset normalized number?
- Largest finite number it can represent?
- Smallest non-zero value it can represent?

Floating point (Examples)

1.b For the same problem, you are given some decimal values below, 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.

Value	Floating Point Bits	Rounded value
9/32	001 00	1/4
3		
9		
3/16		
15/2		

Floating point (Examples)

1.b For the same problem, you are given some decimal values below, 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.

Value	Floating Point Bits	Rounded value
9/32	001 00	1/4
3	100 10	3
9	110 00	8
3/16	000 11	3/16
15/2	110 00	8

Floating point (Examples)

2. Consider the following two 7-bit floating point representation based on the IEEE floating point format. Neither has a sign bit – they can only represent nonnegative numbers.

Format A

- **There are k = 3 exponent bits. The exponent bias is 3.**
- **There are n = 4 fraction bits.**

Format B

- **There are k = 4 exponent bits. The exponent bias is 7.**
- **There are n = 3 fraction bits.**

Convert these bit patterns to the closest value in Format B.

Format A	Format B
011 0000	0111 000
101 1110	
010 1001	
110 1111	
000 0001	

Floating point (Examples)

2. Consider the following two 7-bit floating point representation based on the IEEE floating point format. Neither has a sign bit – they can only represent nonnegative numbers.

Format A

- **There are k = 3 exponent bits. The exponent bias is 3.**
- **There are n = 4 fraction bits.**

Format B

- **There are k = 4 exponent bits. The exponent bias is 7.**
- **There are n = 3 fraction bits.**

Convert these bit patterns to the closest value in Format B.

Format A	Format B	
011 0000	0111 000	
101 1110	1001 111	
010 1001	0110 100	Round down
110 1111	1011 000	Round up
000 0001	0001 000	Denorm -> norm

Datalab Tips

- **Operator precedence**
  - z = x << 2 + y ?
- **Edge cases**
  - 0? T_{min}?
  - Shift by 32? (Undefined behavior!)
- **SubOK()?**
- **Use bddcheck & driver.pl**
  - Test thoroughly and provide more details on failure
  - Please do that early to avoid any grading surprise
  - Declare variables at very beginning of each function

Questions?