15-213 Recitation: Data Lab

___TA___
Jan 28, 2019

Agenda

- Course Details
- Data Lab
  - ANSI C
- Floating Point

Course Details

- How do I get help?
  - Course website: http://cs.cmu.edu/~213
  - Office hours: **5-9PM** from Sun-Fri, **5-7PM** Sat in Wean 5207
  - Piazza
  - *Definitely* consult the course textbook
  - **Carefully read the assignment writeups!**
- All labs are submitted on Autolab.
- All labs should be worked on using the **shark machines.**

Data Lab: Logistics

- How do I get started?
  - Use link in writeup to create git repository
  - From command line: *git clone <url>*

- Use this lab to get good at git
- Check the Datalab FAQ on Piazza
- isPallindrome has an operator limit of 40
- Bootcamp Slides: http://www.cs.cmu.edu/~213/activities/linux-bootcamp/linux-bootcamp.pdf

Data Lab: What is ANSI C?

Within two braces, all *declarations* must go before any *expressions*.

This is *not* ANSI C.

```
unsigned int foo(unsigned int x)
{
    x = x * 2;
    int y = 5;

    if (x > 5) {
        x = x * 3;
        int z = 4;
        x = x * z;
    }

    return x * y;
}
```

Data Lab: What is ANSI C?

This is ANSI C.

```
unsigned int foo(unsigned int x)
{
    int y = 5;
    x = x * 2;

    if (x > 5) {
        int z = 4;
        x = x * 3;
        x = x * z;
    }

    return x * y;
}
```

This is *not* ANSI C.

```
unsigned int foo(unsigned int x)
{
    x = x * 2;
    int y = 5;

    if (x > 5) {
        x = x * 3;
        int z = 4;
        x = x * z;
    }

    return x * y;
}
```

Form Groups of 3 - 4

- Series of exercises
  - Operators
  - Floating point
  - Puzzle

FLOATING POINT

"Normalized" Values

$v = (-1)^s M 2^E$

- When: **exp** ≠ 000…0 and **exp** ≠ 111…1
- Exponent coded as a biased value: E = **exp** – Bias
  - exp: unsigned value of exp field
  - Bias = $2^{k-1} - 1$, where k is number of exponent bits
    - Single precision: 127 (**exp**: 1…254, E: -126…127)
    - Double precision: 1023 (**exp**: 1…2046, E: -1022…1023)

- Significand coded with implied leading 1: M = $1.xxx…x_2$
  - xxx…x: bits of frac field
  - Minimum when **frac**=000…0 (M = 1.0)
  - Maximum when **frac**=111…1 (M = 2.0 – ε)
  - Get extra leading bit for "free"

Denormalized Values

$v = (-1)^s M 2^E$
$E = 1 - Bias$

- Condition: exp = 000…0
- Exponent value: E = 1 – Bias (instead of **exp** – Bias) (why?)
- Significand coded with implied leading 0: M = 0.xxx…x_2
  - **xxx**…**x**: bits of **frac**
- Cases
  - **exp** = 000…0, **frac** = 000…0
    - Represents zero value
    - Note distinct values: +0 and –0 (why?)
  - **exp** = 000…0, **frac** ≠ 000…0
    - Numbers closest to 0.0
    - Equispaced

Special Values

- Condition: **`exp`** = **`111…1`**
- Case: **`exp`** = **`111…1`**, **`frac`** = **`000…0`**
  - **Represents value ∞ (infinity)**
  - Operation that overflows
  - Both positive and negative
  - E.g., 1.0/0.0 = −1.0/−0.0 = +∞, 1.0/−0.0 = −∞

- Case: **`exp`** = **`111…1`**, **`frac`** ≠ **`000…0`**
  - **Not-a-Number (NaN)**
  - Represents case when no numeric value can be determined
  - E.g., sqrt(–1), ∞ − ∞, ∞ × 0

Floating Point: Rounding

1.BBGRXXX

In the below examples, imagine the underlined part as a fraction.

- Guard Bit: the least significant bit of the resulting number
- Round Bit: the first bit removed from rounding
- Sticky Bits: all bits after the round bit, OR'd together

Examples of rounding cases, including rounding to nearest even number

- 1.10|11: More than ½, round up: 1.11
- 1.10|10: Equal to ½, round down *to even*: 1.10
- 1.01|01: Less than ½, round down: 1.01
- 1.01|10: Equal to ½, round up *to even:* 1.10
- 1.01|00: Equal to 0, do nothing: 1.01
- 1.00|00: Equal to 0, do nothing: 1.00

All other cases involve either rounding up or down - *try them*!

Questions?

- Remember, data lab is due this Thursday!
  - You really should have started already!
- Read the lab writeup.
  - **Read the lab writeup.**
    - ***Read the lab writeup.***
      - ***Read the lab writeup.***
        - **Please. :)**