1x-x13 Recitation: How to Succeed in x13

August 30, 2024

Agenda

- Introduction
- Course Details
- Office Hours
- x13 Advice from TAs
- Data Lab

Introduction

- Welcome to 15-213/15-513/14-513!
- Recitations are for…
    - Reviewing lectures
    - Discussing homework problems
    - Interactively exploring concepts
    - Previewing future lecture material

- Please, **please** ask questions!

Course Details

- How do I get help?
  - Course website: http://cs.cmu.edu/~213
  - Office hours
  - Piazza
  - *Definitely* consult the course textbook
  - **Carefully read the assignment writeups!**
- All labs are submitted on Autolab
- All labs should be worked on using our **shark machines**

Office Hours

- Office Hours start Tuesday!
- Queue link: https://ohq.eberly.cmu.edu/#/courses
- Please locate the TA in the specified location!
- Semester’s OH schedule (subject to change)
  - Will be a pinned post in Piazza

OH Etiquette

- Office hours are for getting ideas on how to debug or better approach your homework!

- Please try to narrow down your problem area as much as possible to help TAs help you!

- **Write a description!** If you don’t have a description, you may be frozen/removed from the queue. Make sure to use the tags!

OH Etiquette (contd)

- TAs will only spend 10 minutes per student and then you can rejoin the queue.

- We will close the queue early so everyone can be helped so please keep this in mind!

How to Succeed at 213

Some advice from your friendly TAs ;)

What is success in x13?

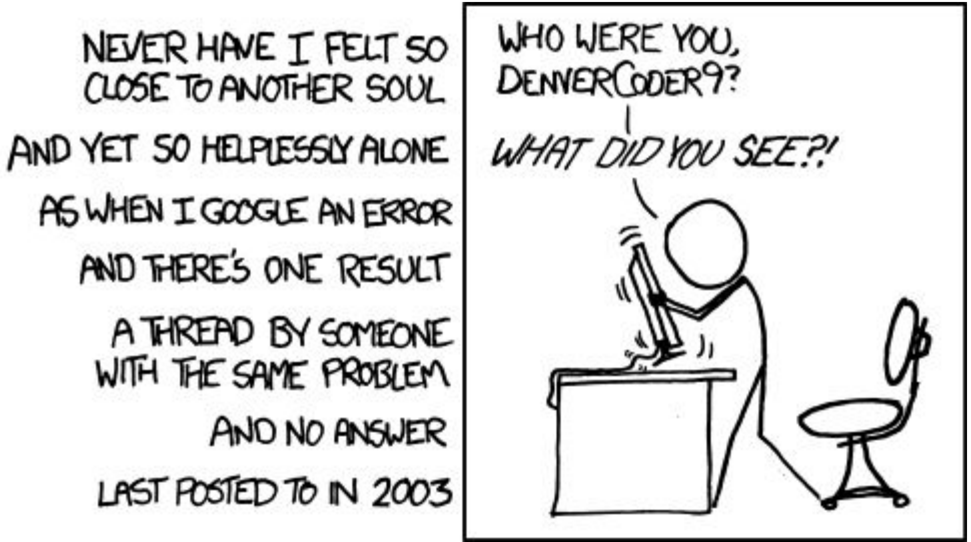
- Some of you (probably most) see success as an A
- … buuuutttttt you can still succeed without getting an A, in fact, true success in x13 is **learning the material**
- And this can be difficult because we will cover *a lot* of different topics, many of which will probably be new to you (and that’s okay!)

How do I learn the material then?

- Engage with the topics in lecture
- Read the textbook
- Learn the material before having to apply it
- Ask questions!

I’ve tried that, but I’m confused. Now what?

- It’s okay to be confused! These topics can be difficult and take time to truly understand
- (Some) online resources are okay to use, but a general google search probably won’t give you helpful results . . .


NEVER HAVE I FELT SO CLOSE TO ANOTHER SOUL
AND YET SO HELPLESSLY ALONE
AS WHEN I GOOGLE AN ERROR
AND THERE'S ONE RESULT
A THREAD BY SOMEONE WITH THE SAME PROBLEM
AND NO ANSWER
LAST POSTED TO IN 2003
WHO WERE YOU, DENVERCODER9?
WHAT DID YOU SEE?!

https://xkcd.com/979/

I need help with a concept

- Read the textbook
- Come to OH and ask your TAs ;)
- Come to Prof. OH (they don’t bite, we promise)
- Ask on piazza
- Ask your recitation TAs to cover the topic again
    - *cough cough wink wink*

I need help with a problem or bug

- Step away and come back after a small break

- Try to solve on your own (debugging for an hour is not that long)
    - Generally, give yourself a day to mull over the problem (your brain will continue to think about it while you do other tasks!)

I need help with a problem or bug (contd)

- If general bug, try some *reputable* sites to find similar problems (see next slide)
- Come to OH!
- Post on piazza!
- Rubber duck method

Actually good online resources

- https://itsfoss.com/linux-man-page-guide/
- https://man7.org/linux/man-pages/
- https://en.cppreference.com/w/c
  - Make sure to use the C (not C++) version!
- https://www.cs.virginia.edu/~evans/cs216/guides/x86.html
- https://beej.us/guide/
- http://www.stackgrowsdown.com/

Other helpful advice!

- Learn GDB early ***before*** you have to rely on it to debug
- Read the writeups (yes, there can be, *and will be*, relevant material on all 20 pages of a writeup)
- Don’t start labs late
- Save some grace days for malloc (~40 hours is average)

Other helpful advice (contd)!

- You don’t have to pass every test case of every assignment

- Be comfortable with the command line (it's not that scary!)

- Be comfortable with different editors (I’m looking at you VScode 👀)

- If you need help, ask! We are here to help YOU!

Data Lab: Getting Started

- Download the handout from autolab
  - Method 1:
    - `scp <path to datalab.tar> <andrewid>@shark.ics.cs.cmu.edu:<my course directory>`
    - `ssh <andrewid>@shark.ics.cs.cmu.edu`
    - `cd to the datalab.tar file`
    - `tar -xf datalab.tar`

Data Lab: Getting Started

- Download the handout from autolab
  - Method 2:
    - `autolab download 15213-s24:datalab`
    - `cd into the datalab folder`
    - `tar -xf datalab.tar`

Data Lab: Getting Started

- Upload `bits.c` file to Autolab for submission
  - `make submit`

Data Lab: Running your code

- `dlc`: a modified C compiler
- `btest`: runs your solutions on random values
- `bddcheck`: exhaustively tests your solutions
  - Checks all values, formally verifying the solution
- `driver.pl`: Runs both dlc and bddcheck
  - Exactly matches Autolab's grading script
  - You will likely only need to submit once
- For more information, **read the writeup**
  - Available under autolab as "**View writeup**"
  - **Read the writeup please!**

Data Lab: Reminders

- Casting between **int** and **long** is ok, in either direction

- Be aware of operations and their types!
  - ! returns an **int** *even if the argument is a long*

- Good idea to append “L” suffix to every integer constant
  - `(1`**`L`**` << 63)` is not the same as `1 << 63`
  - `(!x << 63)` is not the same as `((`**`long`**`)!x) << 63`

If there’s time…

Let’s do an activity :)

Form Groups of 3 - 4

- Series of exercises
  - Operators
  - Puzzle
- There’s a handout on the website :)

Divide and Conquer (Bit Count)

Let’s count how many bits are set in a number. For each challenge, you can use any operator allowed in the integer problems in datalab.

Using 1 operator, we return the number of bits set in a 1-bit number:

```
int bitCount1bit(int x) {return x;}
```

Divide and Conquer (cont.)

How about if there are two bits in the input? (4 ops max)

```
int bitCount2bit(int x) {
    int bit1 = ________________ & ________________;
    int bit2 = ________________ & ________________;
    return ________________ + bit1 ;
}
```

Divide and Conquer (cont.)

How about if there are four bits? (8 ops max)

```
int bitCount4bit(int x) {
    int mask = ______________________________;
    int halfSum = ______________________________;
    int mask2 = ______________________________;
    return ___________________ + ___________________ ;
}
```

Divide and Conquer (cont.)

How about if there are eight bits? (12 ops max)

```
int bitCount8bit(int x) {
    int mask = ______________________________;
    int quarterSum = _____________________________;
    int mask2 = ______________________________;
    int halfSum = ___________________________________;
    int mask3 = _____________________________;
    return ___________________ + ____________________ ;
}
```

Questions?

- Remember
  - cprogramminglab is due Tuesday!
    - You really should have started already!
  - datalab is due Sep 10!
    - We recommend you start just a BIT early!
- Read the lab writeup!

"That's all Folks!"