

WiT & ACM present

DonsHack!

Theme: **UrbanTech**

Up to **\$2000** in prizes!

Hear from alumni!

Sponsored by:



Alani

PRIME
HYDRATION

April 5th-7th

Register here!



Notes

Textbook?

Final exam time possibilities:

- Weds May 15, 10am-12pm
- Weds May 15, 12:30-2:30pm
- Weds May 15, 5:30-7:30pm

You will vote on your choices as part of **Lab14**. Please do this even if you ignore the rest of **Lab14**!

Show "weird.c" in /CProgs

CS112 –
Java
Programming

Spring 2024

Copyright 2023 Paul Haskell. All rights reserved.

Debugging

Debugging

I am among the world's greatest debuggers...because I am very humble :)

Lessons:

- When there is a bug, at least one of your assumptions is wrong
- Divide the system into smaller pieces that are easier to understand
- Make theories but test them
- Keep separate lists of Things You Suspect vs Things You Verified from Testing
- "There is never just 1 bug"

NTL story: Timestamps and buggy test equipment

- Processing time changed the timestamp time
- All test equipment was buggy!
- Graphs, eventually found "file mode".

Debugging

So far I expect you have all been doing what is called "println debugging"

- Add lots of `println()` statements to your code
- Verify your assumptions by confirming values of key data at key points in the code
- Excellent and widely used strategy, but you have to keep on adding `println()` statements to your code, recompiling, and rerunning
- And sometimes bugs takes weeks to reproduce

Graphical Debugger

A program that runs your program

- Can pause and later continue program at one or more points in code
- Can print out values at these "breakpoints" without having to add println's

Eclipse includes great debugging tools

(VSCode and packages, and JARs...)

Debug **Plagiarism.java**

CS221 profs complained my students don't know how to use a debugger. We will practice that today.

Lab14

Just 1 program: **ReverseString.java** . Finish and push to GitHub by the end of class today!

IN PARALLEL pass out midterms