GDB and SSH

Joel Willoughby

COP 3503

Outline

Agenda

- Discuss the process of debugging in general
- Talk about gdb from both the command line and Eclipse platform
- Talk about ssh and ftp and how you can compile your projects on these machines from home

Outline

None this lab

At least that I could think of while writing this

Outline

Debugging

- Debugging the process of removing bugs and errors from your code
- These errors can be fatal, or they can be logical
  - Fatal errors halt your program execution (Ex. Seg Fault)
  - Loogical errors are just incorrect behavior in your program (Ex. functions not returning correct values)
- In general, logical errors are more difficult to track down and fix

What you are probably familiar with

- The easy way to debug is to just use print statements everywhere
- This is usually sufficient but not exactly efficient
- Also, problems with the console stream can lead to incorrect printing out (why we use endl and flush)
- Process of tracking down the error is long and laborious. You have to keep moving print statements, recompile, re-run, etc.
- Debuggers (like gdb) give us a better platform for all of this

A few notes on gdb

- Works only for specific languages and compilers (g++ for C++ and gcc for C are the most popular)
- It provides a command line interface, much like the unix shell you should (hopefully) be used to
- GUIs are built for it (we will see with Eclipse)
- **In order to use gdb to debug your program, you must compile it with the -g option**

```
1 g++ -g blah.cpp blah2.cpp -o blah
```

Outline

Questions

???