Lab 02: Debuggers

September 17, 2020

Introduction

This lab will give you experience using `gdb`. `git clone` the lab repository, then make a solutions file named `answers.txt` to write your answers in (questions are denoted with Q).

Problem 1: Segfaults

1. Compile 'vector.cpp' and run it. Hopefully you get an segfault!
2. Open your executable in `gdb`.
3. Check the backtrace. Q1: What function and line is causing the segfault?
4. Check that function (you can print out bits of code using the `list` command.
5. Set a breakpoint before the segfault, then `start` or `run` your code again and step through it.
6. As you are stepping through, inspect the value of the variables at the breakpoint.
7. Q2: What is the bug in the code? Fix it.
8. Run your code and make sure you have fixed the segfault. You should have another problem now...

Problem 2: Loopy

1. Run the program, then press Ctrl+c to halt execution.
2. Check the backtrace.
3. Put a breakpoint on the loop, then restart the program.
4. `run` the code a few times and `print` the value of the variables.
5. Q3: What was causing the infinite loop? Fix the code.
6. Run the fixed code. Something doesn't add up...

Problem 3: Math is hard

1. The sum is incorrect. Set a breakpoint that lets you watch what the sum function is doing.
2. Inspect the local variables to see what is going on.
3. Q4: What was the cause? Fix the code. Run it to make sure your fix worked.

Epilogue

`git add` your `answers.txt` and corrected code and `git push`!