Kyle Zalewski

CS202

Programming Project 1

The use of GDB in troubleshooting project one has been very helpful in quickly tracking down runtime errors, namely segmentation faults. My usual procedure begins with a command line run of my executable in GDB to discover which part of code is faulting. With this knowledge, I then open a new instance of GDB with the split-screen user interface, using the switch -tui at the command line. This allows me to set break points at the function call where the segmentation fault was encountered, and step through that function to see where my code went wrong. I found this to be an extremely effective and fast method for detecting the source of a fault type that is normally painful to deal with.

Additionally, I took the approach of making small adjustments to my code between compiles. This allowed me to always be only a short step away from a properly running program at every compile and run. By extension, this meant that I was usually only chasing down one problem in GDB at a time and was able to use the tool more efficiently (i.e. with the lowest number of runs). I was effectively able to control the number of successive errors encountered in a debugger run and thereby ensure I was looking at the problem I *thought* I was looking at.