COMP 201 – Fall 2021

Lab Exercise 4: GDB

Due: 7 November 2021 23:59

1 Introduction

Debugging is an essential skill for any software developer. The simple and naive approach of using debug print statements is inefficient, error-prone, and unreliable in general. Real debugging requires a more sophisticated tool called a *debugger*. In this lab, we'll introduce you to the basic usage of GDB, GNU's Debugger tool.

1. Download the buggy.c code from the Blackboard and upload it to the LinuxPool.

2. To get started, we'll need to compile the program specifically for GDB by using the -g flag. You can use the following command for compiling your code where the "OUT FILE NAME" is your desired name of executable file.

```
gcc -g buggy.c -o OUT FILE NAME
```

3. You can run your code by simply calling your executable file. For example if you use gcc -g buggy.c -o main you will get a "main" file in the current working directory in case the compile is successful. So, you can run the file name "main" as below:

```
./main
```

4. Check out the code to get an idea of what the program is trying to compute, but do not change anything yet. Try running the program; it takes 1 command line argument: an integer representing number of the first prime numbers. For debugging that's better to keep it small. The default value is 10 if you don't pass any argument variable.

   For e.g.:

   ```
   ./main //Using default values
   ./main 3 //for first 3 numbers
   ```

5. To start the GNU debugger run your program with command line arguments you can use:

   ```
   gdb --args main 5
   ```

   Alternatively, you can start GDB without args using gdb main and once it has started, you can set (and reset) the argument(s) using:

   ```
   set args 5
   ```

6. After you have GDB prompt and you were able to run the code using debugger, start looking for bugs and fix them. Check the slides and GDB reference sheets online, you have many options like setting breakpoints, executing line by line, checking the variable values, and etc.

7. In the end, please submit your fixed code to the Blackboard.

2 Test Cases

You can use the following test cases to make sure that you've successfully debugged the code.

```
./main
The product of the first 10 primes is 6469693230

./main 30
The product of the first 30 primes is 7581744426003940878
```