CPSC 1070
Lab Project
Sept. 23, 2019

This is a two day project to be completed in lab by Wednesday. If you have not finished by the end of lab on Wednesday, the TA will grade you on what you have completed.

From the lab schedule page, please download the file `broken–calculator.zip`, and unzip it. You should now have a directory named `broken–calculator`. Use `cd` to go into this directory and use `make` to compile the `calculator` program.

This is the same program that we did in class on Friday, but with its own routine called `str2float()` to convert a string to a floating point number. Using this routine is supposed to better than using `atof()` or `sscanf()` to convert the number, because it does complete error checking on the string to make sure that it is a valid floating point number. Unfortunately, the program does not work. If you run it, you will see that it does not compute correctly.

Your task is to use the `gdb` debugger to find and fix any problems with the program. Type

```
gdb calculator
```

to enter the debugger. Then, set any breakpoints that you want to set and try running the program with a set of command line arguments. For example

```
run 3.25 + 2.75
```

I would suggest first setting a breakpoint in the `str2float()` function, and stepping through this function to see what it is doing. You can use `gdb`'s `print` command to print the contents of variables as you go to see what is happening. Fix problems as you find them, `quit` from `gdb`, recompile, and rerun, again using `gdb`, to see if your fixes have worked.

When you think that you have found and fixed all problems in `str2float()`, does the program now work? Remember to check all operators. If it does not always work correctly, try setting breakpoints in main to see if you can track any problems down and fix them. When you are convinced everything is working, Have the TA check you out.