COMP 302 System Programming, Spring 2021
Instructor: Zafer Aydın
Homework 1

Submit your answers to questions below in a text file (i.e. Word document). Submit your codes and files for the calculator program (no need to submit the diction program) as a zip file (you can compress the folder that contains files related to the calculator program).

1. Read chapter 23 of textbook about compiling programs. Do the steps and run the commands described in this chapter. What does the `diction` program do?

2. Run the `diction` program and demonstrate one of the functions of this program.

3. Write a simple C program that implements a simple calculator. The format for calling your script can be

```
./calculator  <first-number> <operator> <second-number>
```

where <first-number> and <second-number> can be any real-valued number, <operator> can be + or – only (i.e. only addition and subtraction).

Example calls for your calculator could be,

```
./calculator  3.0 + 4.0
./calculator  3.0 - 4.0
```

4. Read the following link about preparing a configure script

https://robots.thoughtbot.com/the-magic-behind-configure-make-make-install

First prepare `configure.ac` and `Makefile.am` files as described in the link. Then execute the following commands in Ubuntu

```
aclocal
autoconf
automake --add-missing
```

which will produce `configure` and `Makefile.in` files. Then run the `configure` command. Set the installation directory to the directory that contains your calculator program's source codes. You can read the following thread for this purpose: https://stackoverflow.com/questions/3239343/make-install-but-not-to-default-directories

Include the configure command you used in your report.

5. Run `make` and `make install` commands. State which files are produced after running these commands.

6. Check the executable permissions of your program using the `ls -l` command and make changes if necessary using the `chmod` command so that your program is executable. Include the output of `ls -l` command showing that the calculator program is executable.

7. Run and test your program. Include the outputs to your report for the following sample executions

```
./calculator  3.0 + 5.0
./calculator  3.0 - 5.0
```