Assignment #1
Due: Thursday January 30th, 2020 at 11:59:00 pm

Instructor	Dr. Stephen Perkins
Office Location	ECSS 4.702
Office Phone	(972) 883-3891
Email Address	stephen.perkins@utdallas.edu
Office Hours	Tuesday and Thursday 3:00pm – 5:00pm and by appointment
TA	TBD

Purpose

Demonstrate the ability to use text editors in the shell. Demonstrate the ability to create and execute shell scripts. Demonstrate the ability to create and compile multi-file C++ programs. Demonstrate the ability to access the command line arguments from within a C++ program. Demonstrate proficiency in logging your console input/output using the `script` command.

Assignment

Log all the work you do in the bash shell using the `script` command (see the "**Session Logging"** section below). You will need to get logging working BEFORE you perform any of the steps listed below. You will need to turn in the session log files. See "**Deliverables**" below.

With `script` logging on and working, write a program that consists of two C++ source files and a header file (.h file). The first C++ file contains the main() routine and the second file contains a C++ function. The function prototype should be included in the header file.

> From within **main**(), you must display the count of command line arguments to *stdout*. Further, you must display the value of each command line argument to *stdout*. Finally, you must call the procedure that is defined in the second file.
>
> From within the procedure defined in the second file, you must display a log message that states you are inside the procedure. This log message must be sent to *stderr*.

You are then to create a shell script for compiling your code. The script will individually compile each file and generate the intermediate object file. Once the object files are created, the script will call the linker and link the object files into an executable. Your shell script should echo what it is doing at each step to *stdout*.

You are then to create a second shell script that will run your program several times with varying command line arguments. Each time you call your program, you should **append** the *stdout* to an output file called ***stdout.log*** and you should **append** the *stderr* to an output file called ***stderr.log***. Your shell script should echo what it is doing at each step to *stdout*.

File Headers

All semester you will be turning in files. Each file that that you create that is not "generated output" or fixed format output (log files, output files from your program, fixed format input files, etc.) should contain a file header that lists important information. While each class you attend (and each company you work for) may have a standard that you must adhere to, this class requires a header that looks like:

```
/*
 * Filename      Your filename
 * Date          The date
 * Author        Your Name
 * Email         Your UTD Email address
 * Course        CS 3377.XXX Spring 2019
 * Version       1.0  (or correct version)
 * Copyright     2019, All Rights Reserved
 *
 * Description
 *
 *     A (possibly multi-line) description of
 *     the file contents.
 */
```

Depending on the type of the file you are documenting, you may need to substitute the C++ comment characters for the comment character required for proper usage. For instance, bash shell scripts would need to use the # character for as a comment character.

Session Logging

Session logging will be performed using the Linux *script* command. For details, see the man page for *script* (i.e. **man script**). Make sure that you are careful to NOT OVERWRITE any older script files you may have previously created. *script* will overwrite a file if you give it a file name that already exists. See the *-a* option to protect against this. You may exit the *script* program by typing:

CTRL-D.

You can either keep one long log file that spans multiple login sessions, or you can create multiple log files (or both).

Deliverables

You must submit your homework through ELearning. You must include your .cc files, .h files, shell scripts, stdout.log, stderr.log, and your session log file/files generated by the *script* command. All source files and shell scripts need to have your name, email, and course number commented at the top as discussed in the **File Header** section. All files must be combined into a single .zip file that is then uploaded into ELearning.

Notes

If the compiler complains that /tmp is full ("No space left on device"), then set your TMPDIR environment variable to /scratch. You can do this in your bash shell and in your shell scripts like this

TMPDIR=/scratch; export TMPDIR

All editing must be done within the shell using a text editor.

All work must be performed on cs1.utdallas.edu or cs2.utdallas.edu.

No late homework is accepted.

Example Output

```
$> compile.sh

Setting TEMPDIR environment variable to /scratch
Compiling file1.cc
Compiling file12.cc
Linking files to create executable hw1
Done



$> run.sh

Running 'hw1' with 0 arguments:
     stdout appended to stdout.log
     stderr appended to stderr.log

Running 'hw1' with 1 argument:
     stdout appended to stdout.log
     stderr appended to stderr.log

Running 'hw1' with 5 arguments:
     stdout appended to stdout.log
     stderr appended to stderr.log


$> more stdout.log

argc was: 1
./hw1
Done!
argc was: 2
./hw1
abc
Done!
argc was: 6
./hw1
a
b
```

```
c
d
e
Done!


$> more stderr.log

Inside proc1() as stderr
Inside proc1() as stderr
Inside proc1() as stderr
```