CSE 4252: Lab1 (10 Points)

Due Date: September 12th (11:59 PM)

Overview

These exercises will allow you to have some practice with the Unix/Linux environment, the g++ compiler and basic common situations when developing programs.

Objectives

- Practice with developing environment and programming fundamentals
  - Review of Unix/Linux working environment
  - Practice with g++ compiler
  - Practice with a basic C++ program modification

Instructions

Follow the instructions given in the description of each exercise and refer to the tutorial file for using Unix commands (https://www.tutorialspoint.com/unix/unix_tutorial.pdf). Create a new folder named Lab1 in your Linux working directory. The exercises below should all be created within that new folder.

Exercise 1

Login into your Unix/Linux or follow the instructions here (https://cse.osu.edu/about/remote-access) to login into the stdlinux environment. You can go over the tutorial to practice the UNIX commands and get yourself familiar with your working directory and environment.

Create a new text file (with a 5 lines of text) using any editor like vim, nano, emacs or gedit. Save it. Make a copy of the file and name it differently. Then open the copy with another editor (i.e. different than what you used to create the file) and modify it and save it.

Combine (concatenate) both files into one file named *mergedex1.txt*. Submit the file.

How can you save the terminal process you do in a text file to submit those results?

To record your session in a file, use the command *script*. This command sends everything you do in the terminal to a file. Read the man page (*man script*).
Try:

script lab1ex1.txt (it starts recording)

When finished, type *exit* to terminate the typescripting ("recording") of the terminal session. Look at the file created (use cat command) and make sure the output was recorded. You are going to use this mechanism to record, save and then submit your Labs results in future too. In case your script recorded file is showing junk characters when you do cat (these are control characters), copy paste the terminal's contents to txt file and submit.

Exercise 2

Create your first C++ program file *lab1.cpp* and place it into the Lab1 folder. Compile it and run it by following the instructions below.

Code:

```
#include <iostream>
using namespace std;
int main()
{
  cout << "Difficulty Levels\n\n";
  cout << "1 - Easy\n";
  cout << "2 - Normal\n";
  cout << "3 - Hard\n";

  int choice;
  cout << "Choice: ";
  cin >> choice;
  switch (choice)
  {
    case 1:
            cout << "You picked Easy.\n";
            break;
    case 2:
            cout << "You picked Normal.\n";
            break;
    case 3:
            cout << "You picked Hard.\n";
            break;
    default:
            cout << "You made an illegal choice.\n";
  }
  return 0;
}
```

Compilation:

Save your program as lab1.cpp and compile it with g++ by entering the following command in the terminal:

```
g++ -o lab1 lab1.cpp
```

Note: The executable name is renamed as lab1 by specifying -o option. If you skip it and type *g++ lab1.cpp* instead, the executable is named a.out by default. You can do *ls -al* with both commands to see what executable is generated.

Running:
Run your program by entering the following command (way to run the executable):
lab1
(or ./lab1)
Script records your terminal process by using *script lab1ex2.txt* and submit this file.

Exercise 3

Copy your first program file into *lab1ex3.cpp* in the same folder. Introduce errors in the file. The script is going to show me the errors that was introduced.

Script records in a file *lab1ex3.txt*. Submit this file together with the cpp file.

Exercise 4

Write a C++ Program *lab1ex4.cpp* that accepts temperature in Fahrenheit and display it in Celsius.

Script records few inputs and outputs to *lab1ex4.txt*. Submit this file together with the cpp file.

Submission Instructions

Make sure your programs compile and run correctly before submitting. Submit the zipped folder Lab1.zip that contains the files from the different exercises.

<u>Point Distribution:</u>
(1) **mergedex1.txt** [1 point]
(2) **lab1ex1.txt** [2 points]
(3) **lab1ex2.txt** [1 point]
(4) **lab1ex3.cpp** [2 points]
(5) **lab1ex3.txt** [1 point]
(6) **lab1ex4.cpp** [2 points]
(7) **lab1ex4.txt** [1 point]