CSC 101 **Name:** Christian Morgan

Lab Five **Date:** 2/17/22

Objectives:

- Continue understanding of basic output and input (cin/cout)
- Continue understanding of C++ mathematical operators
- Mathematical Library Functions
- Understanding conditional branching – if statements

Procedures:

1. Boot up your Ubuntu Linux OS and log in.

2. Open a terminal. At the command prompt, examine the man page for g++. What command did you use to do this?

man g++

3. While in the g++ man page, hit the / key. A forward leaning slash should appear at the bottom left side of the screen. This enables the search function of the manual system. Type the word Wall at the forward leaning slash and hit enter. You will notice that the manual skips to the first instance of the word Wall in the man page. Hitting the n key will skip to the next instance of the searched for word.

4. Use the manual system to research the -Wall command line option. What does the -Wall option do, and what sort of flags does it enable?

This enables all the warnings about constructions that some users consider questionable, and that are wasy to avoid (or modifyto prevent the warning), even in conjuction with macros.

-Wall turn on warning flags: - Waddress-bounds=1, -Warray-parameter=2, etc....

5. Use the manual system to research the -Wextra command line option. What is the function of this option?

This enables some extra warning flags that are not enabled by -Wall.

6. Use the manual system to research the -O command line options. What is the function of this option? Briefly explain the levels available for this option.

the optimization level

Place the primary output in file file. This applies to whatever sort of output is bein produced, whtether it can be na executable file , object file, an assembler file ot prepreprocesses C code.

7. Examine the program listing below:

```
#include<iostream>
#include<cmath>
using namespace std;

int main()                                                    OUTPUT
{
    int x = 5, y = 8, z = 10;

    cout << ( x < y ) << endl;                                1
    cout << ( y != y ) << endl;                               0
    cout << ( x == z ) << endl;                               0
    cout << ( y == y ) << endl;                               1
    cout << ( x + y > z ) << endl;                            1
    cout << ( y > x + 5 ) << endl;                            0
    x = ( z * y - x * z ) / x;
    cout << ( x != z ) << endl;                               1
    cout << ( x / z > y - 5 ) << endl;                        0
    cout << ( x + 4 <= z - 5 ) << endl;                       0
    cout << ( x * y <= pow(x,2.0) ) << endl;                  0
    cout << ( y * z <= pow(x,2.0) )  << endl;                 0
    cout << ( x >= sqrt(z) ) << endl;                         1
    cout << ( x * z % y < 5 ) << endl;                        1

    if (x > 100)
        cout << "a" << endl;
        cout << "b" << endl;

    if (x < y)
        cout << "one" << endl;                                1
    else
        cout << "two" << endl;                                b

    if (x + y >= y + z)
    {
        cout << "three" << endl;                              one
    }
    else
    {
        cout << "four" << endl;                               four
    }

    return 0;
}
```

Use the spaces to the right of the listing to show the output of the program. Only list output for statements that are actually executed by the program.

8. Create a new directory called prog5. Type in the program using a text editor and save the source file as 'prog5a.cpp' in the new directory. Compile the program with all warnings enabled. What command did you use to accomplish this?

9. Run the program, comparing its output to your estimate in the previous step. Explain any difference between your estimation and the actual output of the program.

1, 0, 0, 1

1, 0, 1, 0

0, 0, 0, 1

1, b, one, four

10. Copy the Makefile from the previous lab and alter it so that it will compile prog5a.cpp. List the contents of the new Makefile below:

```
all: prog5a

prog4a: prog5a.o
        g++ prog45.o -o prog5a

prog5a.o: prog5a.cpp   g
        ++ -c prog5a.cpp

clean:
        rm -rf *.o prog5a
```

11. Examine the Fahrenheit to Celsius program you created in Lab 4. Alter this program so that it can convert both Fahrenheit to Celsius as well as Celsius to Fahrenheit. Prompt the user for which conversion to make, then input the temperature from the keyboard, make the proper conversion (whichever the user selects) and output the converted temperature. Supply a flowchart that shows the logic of your altered program and attach it to this lab sheet. As usual, make sure you write a complete C++ program and add copious comments. Print out the source code and attach it to this lab sheet.

12. Write a complete C++ program that prints all the real solutions to the quadratic equation:

$$ax^2 + bx + c = 0$$

for a given set of coefficients. Declare variables (of an appropriate type and using meaningful variable names) to represent the coefficients a, b, and c, as well as the solutions. Then read the values for a, b, and c from the keyboard and use the quadratic formula:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

to generate the solutions. Print out a message when the user supplies values for which the roots are undefined.

Run the program several times, with different inputs. Try to explain any anomalous results.

Make sure you write a complete C++ program. You may use additional variables for the calculations as necessary. Add judicious comments to document your code.

Include a header, like the one from previous labs, displaying your name, the date, the course number, and a short description of the program.

When you have finished, print out the code and attach it to this lab sheet.