Aryan Aziz

SID 932431149

CS165

Assignment 1 Report

**Understanding –**

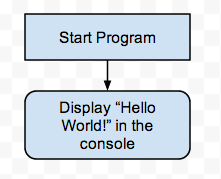
As these are the first 2 programs we are writing, they both seem pretty straightforward.

For the first program we are creating a simple “Hello World!” implementation that, when ran, displays the text “Hello World!” in the console. There are no external inputs, and only 1 output.

The second program dives a bit deeper and wants to request the user for a value (in this case, an integer) and then display that value back to the user.

**Design –**

**hello\_world.cpp**

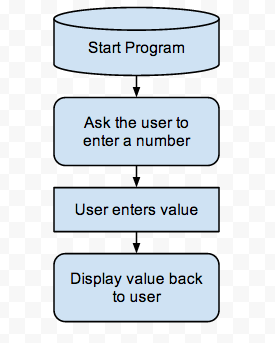


It is never nice to place all of your code on a single line (unless you’re minifying it to conserve space) so I decided to keep it clean and neat and break everything out.

The design itself (following the theme of making the code easy to read) should only be 5 lines (or 4 lines if you want to move line 5 at the end of like 4 below).

1. Including IOStream
2. Defining the main function
3. Outputting the Text
4. Ending the main function/Returning 0
5. Closing Bracket

**echo.cpp**



This one is a little more in-depth and it requires us to take the 5 lines outline above and add in 3 extra lines (not in any specific order):

1. Defining the variable
2. Asking for User Input
3. Outputting User Input

I decided on using an integer as my variable because who doesn’t like clean, whole numbers?

**Testing –**

**hello\_world.cpp**

The only testing involved here was to run the program and see if it output the correct text.

**echo.cpp**

Other than the output text, I had to test to make sure that it was requesting the value and when entered it was correctly giving back the value entered. However, it is important to also check what happens if the user enters a value you’re not expecting, in this case, anything other than an integer.

Input – Double: Rounds the number down to the nearest whole number.

Input – String: Assigns the value of 0 to the variable.

Input – Boolean: To my understanding there is no way for the user to enter a Boolean value in this case. Entering 0 or 1 will pass because they are still integers, and entering “true” or “false” will be treated as a string and fail.

**Reflection –**

I became pretty intrigued with the idea that my echo.cpp program was flawed. Although it’s meant to be treated as an early program to help learn syntax and usage, I wanted to define it further if the user enters a value it wasn’t expecting.

I played around with an if statement that checked if cin failed (!std::cin). This produced….mixed results. It definitely noticed if the input came as a string instead of a number of some sort and displayed the cout message in my if statement, but if the input was a double (ie 1.0) then it still did its normal thing.

After more research, I found that there are a few steps that must be done including the cin.clear and cin.ignore functions. I’m not entirely sure why you would need to use them both in conjunction, to my understanding if you’re clearing it why do you need to ignore the first maximum amount of the value? Something I definitely need to look into more.

Nonetheless, my loop hits a weird stop. With everything, I was able to get it to ask for the value again, but if I enter the value incorrectly twice then my loop suddenly ends and the program closes. Another thing I want to look into more.

For future reference, the code including the if statement (not my assignment submission): https://gist.github.com/anonymous/ca173965334a52200225