Olivier Dorvelus
jemansfield
CS1300AA
Program4
Chapter 4- Factorial Numbers
Due Date:4/12/2016

Program Goal:

The goal of the program is to take a nonegative interger input and output the factorial of that number. This is now possible due to the use of loops. The program has to ensure that the factorial code only runs when the user inputs a nonnegative integer and puts the user back into the loop if that condition is not met.

Preconditions:

User has to input a number
Temporary variable
Counter

Source Code:

```
#include <iostream>
#include <iomanip>

using namespace std;

int main()
{
   int n;
   int counter = 0;
   double factorial = 1;

   cout << "Please enter a nonnegative integer number: ";
   cin >> n;
   while(n < 0) {
   cout << "Invalid integer, please try again: ";
   cin >> n;
   }
   int temp = n;
   while(counter < n) {
      if(n == 0)
         factorial = factorial;
      factorial = factorial * temp;
      temp--;
      counter++;
```

```
}
   cout << fixed << setprecision(0) <<  n << "! = " << factorial << endl;

  //  else {
    //    cout << "Please enter a valid number next time" << endl;
   // }
    return 0;
}
```

Postconditions:

The program will output the factorial of whatever integer the user inputs.

Results/Output:

If user enters a valid integer:

Please enter a nonnegative integer number: 5
5! = 120

If user enters a non-valid integer:

Please enter a nonnegative integer number: -4
Invalid integer, please try again: -16
Invalid integer, please try again: -15
Invalid integer, please try again: -1
Invalid integer, please try again: 8
8! = 40320

Reflection

Overview:

The program takes an integer input from the user and checks if its a nonnegative integer. If the integer is negative, the program will prompt the user until the user inputs a valid integer. Once a valid integer is passed, the program will calculate the factorial of that number and output that result.

Problems & Solutions:

The slight problem I ran into was trying to figure out how to ensure the loops after the first iteration stores the last value and multiply it by whatever number the factorial is currently up to. I realized I had to use a temp variable to mimic n so n is not affected. If n were to be affected by subtracting 1 from it after every iteration, it will affect the while loop and end result of the calculation.

Lessons Learned:

I read the problem and thought it was fairly easily so I rushed my code and found myself with calculations error. I thought about it more and fixed it in about five minutes so never rush your code even if the problem seems really simple.