```
//  TECH 3044
//  Calc + Analytical Geometry
//  Spring 2013
//  The University of Memphis
//
//  Assignment:    MacLauren Series
//  Developer:     John McDonald
//  Version:       1.0
//  Compiler:      Codeblocks 13.12
//  Date:          March 1, 2013
//
//  Program Description:
//
// Calculate Maclauren series
//

/******************************/
/*  PRE-PROCESSOR DIRECTIVES  */
/******************************/
#include <iostream>
#include <conio.h>
#include <math.h>
#include <assert.h>
#include <cmath>
using namespace std;

/******************************/
/*  FUNCTION PROTOTYPES       */
/******************************/
void DescribeProgram(void);
void RunProgram(void);
float Maclauren_Series(float);
float Library_Function(float);
bool AreEqual(float, float);
void WaitForKeypress(void);
int factorial(int);

/******************************/
/*  main() FUNCTION           */
/******************************/
int main(void)
{
    //  only function call permitted in 'main()'...
    RunProgram();
}

/******************************/
/*  DescribeProgram()         */
/*                            */
/*  Function used to describe */
/*  the program.              */
/*                            */
/*  Parameters:  None         */
/*  Returns:     void         */
/******************************/
void DescribeProgram(void)
{
    cout << "Calculate the Sine of 30 degrees using" << endl;
    cout << "the Maclauren series." << endl;
    cout << "Compare that result with a library function.\n" << endl;
}

/******************************/
```

```
/*  RunMyProgram()            */
/*                            */
/*  Function used to call all */
/*  necessary code for prog.  */
/*  execution.                */
/*                            */
/*  Parameters:  None         */
/*  Returns:     void         */
/******************************/
void RunProgram(void)
{
    float angle;
    //  describe the program...
    DescribeProgram();

    cout << "Sine of 30 Degrees...\n";
    angle = 30.0;
    //cout << "Please enter an angle: ";
    //cin >> angle;
    // convert to radians...
    float pi = 3.14159265;
    angle = angle * (pi/180);
    float lib_sin = Library_Function(angle);
    float mac_sin = Maclauren_Series(angle);
    cout << "\n\n";
    cout << "C Library Function Returned: ";
    cout << lib_sin;
    cout << "\n\n";
    WaitForKeypress();
}

/******************************/
/*  Maclauren_Series(float)   */
/*                            */
/*  Calculate Sine            */
/******************************/
float Maclauren_Series(float angle)
{
    float n = 1.0, x = 0.0, calc_sin = 0.0, lib_sin = 0.0;
    float sign = 1.0;
    int counter = 0;

    lib_sin = Library_Function(angle);

    cout << "Now calculating Maclauren Series...\n";
    while(AreEqual(lib_sin, calc_sin) == false)
    {
        calc_sin += sign*((pow(angle, n))/(factorial(n)));
        n = n+2.0;
        sign = sign*(-1.0);
        cout << calc_sin;
        cout << "\n";
        counter++;
    }
    cout << "\nMaclauren Series Returned: ";
    cout << calc_sin;
    cout << " using ";
    cout << counter;
    cout << " terms.";
    return calc_sin;
}

int factorial(int n)
```

```
{
    if (n == 0)
       return 1;
    return n * factorial(n - 1);
}

// here you can adjust the accuracy level (decimal places)
bool AreEqual(float a1, float a2)
{
    if(max(fabs(a1 - a2), fabs(a1 - a2) / a1) <= pow(10,-6))
        return true;
    else if(max(fabs(a1 - a2), fabs(a1 - a2) / a1) <= pow(10,-3))
        return false;
    else
    return false;
}

/******************************/
/*  Library_Function(float)   */
/*                            */
/*  Calculate Sine            */
/******************************/
float Library_Function(float angle)
{
    float lib_sin = 0.0;
    lib_sin = sin(angle);
    return lib_sin;
}

/******************************/
/*  WaitForKeypress()         */
/*                            */
/*  Function used to pause    */
/*  until key has been        */
/*  pressed.                  */
/******************************/
void WaitForKeypress(void)
{
    // wait for the user to press key to quit...
    cout << "\nPress any key to quit...";
    getch();
}
```