



C++ PROGRAMMING

(335)

REGIONAL – 2015

Production Portion:

Program 1: Approximating the Value of π _____ (350 points)

TOTAL POINTS _____ (350 points)

Judge/Graders: Please double check and verify all scores and answer keys!

Property of Business Professionals of America.
May be reproduced only for use in the Business Professionals of America
Workplace Skills Assessment Program competition.



Contestants were asked to run the program for n at 1,000 and n at 100,000. The grader should run the contestant program to the results below to about 14 degrees of precision depending on the compiler. At times the compiler will not always round as indicated in the code provided by the contestant. When the value for n is changed, the compiler can throw more values of precision than is indicated. Please keep this in mind when grading. At this point, if the grader so chooses, the code should be checked to make certain precision was set and points may be granted.

Execution for n at 1,000

```
Enter an integer n greater than 0: 1000
Computing Pi Series Summation by ML Formula
=====
The value of n in the Leibniz Series: 1000
The Leibniz Series approximation of PI at n: 3.142591654339544

The known value of PI to 15 decimal points: 3.1415926535897931
The difference between PI and the Leibniz Series: 0.000999000749751

The approximation of PI using 22/7: 3.142857142857143
The difference between PI using 22/7 and the Leibniz Series: 0.000265488517599

The approximation of PI using 355/113: 3.141592920353983
The difference between PI using 355/113 and the Leibniz Series:0.000998733985562
```

Execution for n at 100,000

```
Enter an integer n greater than 0: 100000
Computing Pi Series Summation by ML Formula
=====
The value of n in the Leibniz Series: 100000
The Leibniz Series approximation of PI at n: 3.1416026534897203

The known value of PI to 15 decimal points: 3.1415926535897931
The difference between PI and the Leibniz Series: 0.000009999899927

The approximation of PI using 22/7: 3.142857142857143
The difference between PI using 22/7 and the Leibniz Series: 0.001254489367422

The approximation of PI using 355/113: 3.141592920353983
The difference between PI using 355/113 and the Leibniz Series:0.000009733135738
```



```
// C++ code for regional 2015 c++ programmer
// Programmer: _____
// This program calculates the value of PI using the Libnitz series

#include <iostream>
#include <math.h>
#include <iomanip>
#include <fstream>

using namespace std;

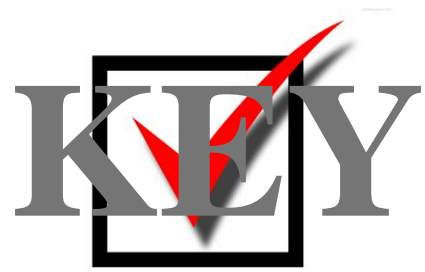
double promptForLimitValue(void);
double valueOfLibnizSeries(double);

int main()
{
    int counter = 1;
    do
    {
        // assign values to constants
        double knownValueOfPi = 3.141592653589793;
        double valueOfPi22by7= 22.0/7.0;
        double valueOfPi355by113 = (double)(355.0/113.0);

        // declare function prototypes
        int nLimit = promptForLimitValue();
        double LibApprox = valueOfLibnizSeries(nLimit);

        // Create the report
        cout << "The value of n in the Leibniz Series: " <<nLimit <<endl;
        cout << "The Leibniz Series approximation of PI at n: "
            << setprecision(16) <<LibApprox <<endl <<endl;
        cout.setf(ios::fixed);
        cout << "The known value of PI to 15 decimal points: "
            << setprecision(16) << knownValueOfPi <<endl;
        cout << "The difference between PI and the Leibniz Series: "
            << setprecision(15) <<abs(knownValueOfPi - LibApprox) <<endl <<endl;
        cout << "The approximation of PI using 22/7: "
            << setprecision(15) <<valueOfPi22by7 <<endl;
        cout << "The difference between PI using 22/7 and the Leibniz Series: "
            << setprecision(15) <<abs(valueOfPi22by7 - LibApprox)<<endl <<endl;
        cout << "The approximation of PI using 355/113: "
            << setprecision(15) <<valueOfPi355by113 <<endl;
        cout.setf(ios::fixed);
        cout << "The difference between PI using 355/113 and the Leibniz Series:"
            << setprecision(15) <<abs(valueOfPi355by113 - LibApprox)<<endl <<endl;
        cin.get();
    } while(counter < 5);
    return 0;
}

double promptForLimitValue(void)
{
    // Prompt for value of n
    int nLimit = 0;
    while(1)
    {
        cout << "Enter an integer n greater than 0: ";
        cin >> nLimit;
        if (nLimit > 0.0 )
            break;
        else
            { cout << endl << "n must an integer > 0" <<endl;}
    }
    return nLimit;
}
```



```
}
```

```
double valueOfLibnizSeries(double nlimit)
{ // Calc libniz series based upon value prompted nlimit
  double approx = 0.0;
  for(double i = 0; i <= nlimit; i++)
  {
    approx += 4 * ((pow(-1, i))/((2 * i) + 1));
  }
  return approx;
}
```



Your application will be graded on the following criteria:

Solution and Project

- The project is present on the flash drive _____ 10 pts
- The project is named according to the naming conventions _____ 10 pts

Program Execution

- Code copied to USB drive and program runs from USB _____ 20 pts

If the program does not execute, then the remaining items in this section receive a score of zero.

- The program rejects negative numbers when requesting a value for n _____ 15 pts
- The program runs and produces correct output for n = 1,000 _____ 25 pts
- The program runs and produces correct output for n = 100,000 _____ 25 pts
- The output matches the sample output in format and alignment _____ 25 pts
- The program displays the results of the Leibniz approximation to 15 decimals _____ 10 pts
- The program displays the 22/7 approximation to 15 decimals _____ 10 pts
- The program displays the 355/113 approximation to 15 decimals _____ 10 pts
- The program displays the difference between the Leibniz approximation and the known value of π to 15 decimals _____ 10 pts
- The program displays the difference between the 22/7 approximation of π and the known value of π to 15 decimals _____ 10 pts
- The program displays the difference between the 355/113 approximation of π and the known value of π to 15 decimals _____ 10 pts

Source Code Review

- Code is commented at the top, for each function, and as needed _____ 15 pts
- Code uses reasonable and consistent variable naming conventions _____ 15 pts
- A function is used to obtain the number of terms to use in the Leibniz series _____ 30 pts
- A function is used to calculate the Leibniz approximation of PI that has the signature of: `double Leibniz(int terms)` _____ 30 pts
- The program uses output formatting to limit the difference between the Leibniz approximation of π and the known value of π to 10 digits _____ 10 pts
- The program uses output formatting to limit the difference between the 22/7 approximation of π and the known value of π to 10 digits _____ 10 pts
- The program uses output formatting to limit the difference between the 355/113 approximation of π and the known value of π to 10 digits _____ 10 pts
- All differences are calculated as absolute values _____ 10 pts
- Output formatting is used to limit the Leibniz series approximation of π to 15 Decimal positions _____ 10 pts
- Output formatting is used to limit the 22/7 approximation of π to 15 decimal positions _____ 10 pts
- Output formatting is used to limit the 355 / 113 approximation of π to 16 digits _____ 10 pts

Total Points: 350 pts