Contestant Number: ________________

Time: _________

Rank: _________

C++ PROGRAMMING
(335)

REGIONAL – 2015

Production Portion:

Program 1: Approximations of π	_______________	(350 points)
TOTAL POINTS	_____________	***350 points)***

Failure to adhere to any of the following rules will result in disqualification:

1. **Contestant must hand in this test booklet and all printouts. Failure to do so will result in disqualification.**
2. **No equipment, supplies, or materials other than those specified for this event are allowed in the testing area. No previous BPA tests and/or sample tests or facsimile (handwritten, photocopied, or keyed) are allowed in the testing area.**
3. **Electronic devices will be monitored according to ACT standards.**

No more than ten (10) minutes orientation
No more than 90 minutes testing time
No more than ten (10) minutes wrap-up

Approximating the Value of π

In 1836, Gottfried Wilhelm von Leibniz, a German mathematician and philosopher, rediscovered the 14th century infinite series that slowly converges upon the value of π. The series, called the Leibniz series, is shown below.

$$\frac{\pi}{4} = \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1}$$

When expanded, the series becomes:

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} \cdots \frac{1}{2n+1}$$

Write a program that approximates the value of π using the Leibniz series and compares the Leibniz series approximation against the known value of π and two other common approximations of π (22/7 and 355/113) and displays the difference between the approximations. The program requirements follow.

Requirements:

1. You must create a C++ or C# console application named CPP_335_ContestantNumber, where ContestantNumber is your BPA assigned contestant number (including dashes). For example, CPP_335_01_2345_6789.
2. Your name and contestant number must appear as a comment at the top of the main source code file.
3. The program must prompt the user to enter a positive integer that indicates the number of terms to calculate in the Leibniz series.
4. A function must be used to obtain the number of terms to use in the Leibniz series calculation. This function must reject negative numbers and request the user to enter a positive number or zero.
5. The Leibniz series must be calculated in a function with the prototype `double valueOfLibnizSeries(double nlimit).`
6. Display the known value of PI to 15 decimal places as 3.141592653589793.
7. The program must display the result of the Leibniz series approximation of π to 15 decimal places as shown below.
8. The program must display the approximation of π using 22.0 / 7.0. Display this approximation to 15 decimal places.
9. The program must display the approximation of π using 355.0 / 113.0. Display this approximation to 15 decimal places.
10. The program must display the difference between the Leibniz series approximation of π and the known value of PI as an absolute value. Display this value to 10 decimal places.
11. The program must display the difference between the 22.0 / 7.0 approximation of π and the known value of PI as an absolute value. Display this value to 15 decimal places digits.

12. The program must display the difference between the 355.0 / 113.0 approximation of π and the known value of PI as an absolute value. Display this value to 10 significant digits.
13. The program will be tested with the value of n at 1,000 and 100,000.
14. The output of the program must look similar to the following.

Sample Output for values for n at 1,000,000:

The value of n in the Leibniz Series:	1,000,000
The Leibniz Series approximation of PI at n is:	3.141593653588775
The known value of PI to 15 decimal points:	3.141592653589793
The difference between PI and the Leibniz Series:	0.000000999998982
The approximation of PI using 22 / 7:	3.142857142857143
The difference between PI and the Leibniz Series:	0.001263489268368
The approximation of PI using 355 / 113:	3.141592920353983
The difference between PI and the Leibniz Series:	0.000000733234792

15. The output in the above should be aligned as shown. All differences should be calculated as absolute values.
16. You will have 90 minutes to complete your work.
17. Your name and/or school name should NOT appear on any work you submit for grading.

Save and submit a copy your entire solution/project on the flash drive provided. You must submit your entire solution/project so that the graders may open your project to review the source code and/or build and execute your solution/project. **Submissions that do not contain source code will *not* be graded**.

Development Standards

- Standard name prefixes must be utilized for variables.
- All subroutines, functions, and methods must be documented with comments explaining the purpose of the method, the input parameters (if any), and the output (if any).

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 is 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