Computer Science I 1370

Programming Assignment 1: Equation Evaluator

I. Learner Objectives:

At the conclusion of this programming assignment, participants should be able to:

- Analyze a basic set of requirements for a problem and derive logical solutions to them
- Declare variables
- Apply C++ data types and associated mathematical operators
- Comment a program according to class standards
- Logically order sequential C++ statements to solve small problems
- Compose a small C++ language program
- Compile a C++ program using Microsoft Visual Studio
- Execute a program
- Create basic test cases for a program

II. Prerequisites:

Before starting this programming assignment, participants should be able to:

- Access Microsoft Visual Studio Integrated Development Environment (IDE)
- Summarize topics from the textbook on ZyBooks
  - C++ language elements
  - The standard C++ data types

III. Overview & Requirements:

Write a C++ program that evaluates the equations provided below. The program must prompt the user for inputs to the equations and evaluate them based on the inputs. All variables on the right hand sides of the equations must be inputted by the user. All variables, except for the *plaintext*_character, *encoded_character*, variable *a*, *shift*, *R1*, *R2*, and *R3* are floating-point values. The *plaintext_character* and *encoded_character* variables are characters, and the *a*, *shift*, *R1*, *R2*, and *R3* variables are integers. The constant PI must be defined as a constant macro (#defined constants). Error checking is not required for your program. You do not need to check for faulty user input or dividing by zero. However, please consider inputs that could cause your program to work incorrectly.

1. Newton's Second Law of Motion: force = mass * acceleration
2. Volume of a cylinder: volume_cylinder = PI * radius^2 * height
3. Character encoding: encoded_character = (plaintext_character - 'A') + 'a' - shift; *shift* is an integer (note: what happens if plaintext_character is uppercase? What happens with various shift keys?)
4. Distance between two points: distance = square root of $((x_1 - x_2)^2 + (y_1 - y_2)^2)$
5. Tangent: tan_theta = sin (theta) / cos (theta) (recall: find the appropriate functions in <math.h>)
6. Equivalent parallel resistance: parallel_resistance = 1 / (1 / R1 + 1 / R2 + 1 / R3), for 3 resistors. R1, R2, and R3 are integers.
7. General equation: y = (2 / 3) - y + z * x / (a % 2) + PI (recall: *a* is an integer; the 2 and 3 constants in the equation should be left as integers initially, but explicitly type-casted as floating-point values)

Note: you will need to display the results for all of the equations!

IV. Submitting Assignments:

1. Please submit your solution on Blackboard.
2. Your project should contain your C++ source file (which must be a .C++ file).

3. **Your project must build properly. The most points an assignment can receive if it does not build properly is 65 out of 100.**

VI. Grading Guidelines:

This assignment is worth 100 points. Your assignment will be evaluated based on a successful compilation and adherence to the program requirements. We will grade according to the following criteria:

- **5 pts for correct declaration of constant macros**
- **35 pts for proper prompts and handling of input (5 pts/equation)**
- **49 pts for correct calculation of results based on given inputs (7 pts/equation)**
- **11 pts for adherence to proper programming style established for the class and comments**