Programming Assignment 1/2

Scenario:

You have secured your first IT internship at Signs 'R Us, a company that creates neon advertising signs. You have been asked to create a program to help sign designers estimate the total cost to create a sign, based on the following gathered information:

- The minimum pre-tax total cost for all signs, regardless of options selected is $106.06
- All sign orders are subject to a 6% sales tax
- When the sign contains 5 words or less, the base sign cost is $93.24
- When the sign contains more than 5 words, but less than 10 words, the base sign cost is $102.72
- When the sign contains 10 or more words, the base sign cost is $107.16
- When the sign should be printed using an Arial font, there is no surcharge, but using a Comic Sans font requires a $10.60 surcharge

Create a program that will first ask the user for the number of words the sign will contain. [If the user inputs a value of 0 or a negative number, the program should end with a message stating the user has input an invalid number of words and they should restart the program.] Next, the program will ask the user for the font to be used on the sign. [If the user inputs a value other than Arial or Comic Sans, the program should end with a message stating the user has input an invalid font and they should restart the program.] After asking for the number of words on the sign and the font, the program will calculate the total cost, including any surcharges and tax. Display a well-formatted message that includes the number of words the sign contains, the selected font, the pre-tax amount, the amount of the tax, and the grand total (including tax).

To Do (Check Blackboard for Due Dates):

Programming Assignment 1: Solution Design

1) Create a defining diagram that shows the input, processing, and output
2) Create a solution algorithm using pseudocode
3) Show testing using the desk checking table method, to include test data, expected results, and a desk checking table. Make sure your desk checking considers multiple cases including both valid and invalid test data to prove your algorithm will work

Upload a Word document containing only items above to Blackboard.

Grading Criteria	
Requirement	**Points**
Defining Diagram with input, processing, and output	40
Efficient Solution Algorithm	40
Thorough Desk Checking Table including test data, and expected results	20

Full points will be awarded for an accurate, efficient, complete defining diagram, solution algorithm, and desk checking table. Partial credit is available.

Programming Assignment 2: Solution Implementation

Write a well-documented, efficient Java program that implements the algorithm you identified. Include appropriate documentation as identified in the documentation expectations document.

Note: You may not use the Scanner or System.out classes. For input/output, you must use the JOptionPane class.

Upload the .java file of the final program to Blackboard.

Full points will be awarded for an accurate, efficient, complete Java program. Partial credit is available. Any final program that does not compile will receive an automatic zero.

Grading Criteria	
Requirement	**Points**
Implementation of Java Program, using efficient practices where appropriate, such as the use of constants, good variable names, no redundant code, etc.	70
Appropriate objective-style documentation	10
Appropriate intermediate comments	20