CISS 160 Homework - Calculation Problems with Accums

The grade you earn will be based on a number of factors, including overall quality/correctness/validity of your application, following the file naming guidelines, following the compressed folder guidelines (see below and Syllabus), and following the other guidelines such as comments for your name, programming comments, test cases etc.

General guidelines for each program:

> Include three comment lines within each program's Form1.cs file with your name, student id number, date, and goal/purpose of the program:

//Author: Your NAME

//ID: Your Student ID Number (NOT YOUR SOCIAL SECURITY NUMBER)

//Date:

//Goal-Purpose of the Program: (…your description…)

> within any program file where you write source code, include comments throughout your code describing in your own words, what the various sections of your code are doing; single line comments can be preceded by two forward slashes //COMMENT…

> for each program create a Text file in the folder where your project files are located and call it TestPlan.txt – edit that file for each program below to include a description of the different 'Tests' that you performed to give yourself assurance that your program is valid/is working correctly; this can include validating data to be numeric, validating that required data was entered, exception handling, other tests to verify specific calculations, functionality, verifying User Interface events etc.

Write the C# programs described below:

> the requirements for each program will be described as a paragraph below OR the requirements will be represented as a compiled .exe of a program (note: this is not source code) that I include in the assignment folder so you can see a running version of how your program should work

> we will cover general aspects of various programming elements during lectures, but it is up to you to combine the lecture learning components along with the textbook reading to create a specific solution

> any images/graphics that are needed will be included in the assignment folder or I will provide in Angel

(see next page)

CISS 160 Homework - Calculation Problems with Accums

1. (30 Points) Auto Miles Per Gallon Calculator. Create a C# windows form that calculates a car's average miles per gallon. Be sure to refer to the general programming guidelines listed above. Name your project:

AutoMPGCalc - Your Name

Input/Data Entry

Assume you fill your tank completely at Gas Station #1, allow user entry of your car's BEGINNING Odometer Reading into your Form (no need to enter the gallons or price per gallon for this fill-up)

….assume you drive for some number of miles….

Assume you fill your tank completely again at Gas Station #2, allow user entry of your car's ENDING Odometer Reading into your Form

Allow user entry of the Gallons that were needed to fill your tank when you were at Gas Station #2 only (this tells you how many gallons you used for the MPG calculation)

Allow user entry of the Price you paid Per Gallon at Gas Station #2

Processing/Output

Add a Button called 'Calculate'

Calculate and Display the following as Label Controls:

- Miles Traveled between Stop #1 and Stop #2

- Miles Per Gallon between Stop #1 and Stop #2

- Fuel Cost Per Mile between Stop#1 and Stop #2

Validate the data entered:

Use a C# *try { … } catch { …. }* within your code to validate that the form text boxes have valid numeric data entered into them by the user

- in your code, the *try { }* will contain in it's braces, whatever your code is that takes the form's textbox fields and converts them to numeric variables

- in your code in the *catch { }* display some kind of message using *MessageBox.Show("Your error message here");* and then add a *return;* within the catch, to return control to the Form

(see next page)

CISS 160 Homework - Calculation Problems with Accums

2. (60 points) Mulch Delivery Truck Capacity Calculator with Accumulations - see the included .exe Note that you do not need to validate the input from the user. Be sure to refer to the general programming guidelines listed above.   Name your project:

MulchCalcWithAccums - Your Name

a) Create a C# Windows Application (see .exe for an example) that allows entry of height, width and depth (in feet) for one order of mulch. Calculate the cubic feet and cubic yards of mulch for one order given that 1 cubic yard = 27 cubic feet (note: make this a 'constant'). Assume a price of $30.00 per cubic yard of mulch (make this a 'constant') and calculate the cost of an order using the price multiplied by cubic yards ordered. Display the price and all other calculated fields for one order on your Form.

b) Accumulate all of the order quantity/cost calculations by adding each Order's calculations to a running total of all Orders since the window was opened, and display accumulation calculations on your Form (remember accumulations require 'field' variables that have to be declared below the 'class' and above/outside of the method/event handlers). Also keep a running count of the number of Orders so far that increments by 1 each time the Calculate button is clicked.

c) Calculate when the accumulated cubic yards of all orders fill a truck that can hold 4 cubic yards of mulch, displaying the Truck Base Capacity (assume a truck can hold 4 cubic yards of mulch and make this a 'constant'), Cubic Yards Ordered so far; then using those two fields calculate the Truck's Remaining Capacity in cubic yards.

d) Add a Clear button to clear all fields on the form and to reset all the accumulators on the Form and in memory variables

e) Make the background color of the label that displays the Truck's Remaining Capacity Green if the Truck's Remaining Capacity is greater than zero, or set the background color to red if it is less than or equal to zero

3. (20 Points **Optional Extra Credit**) Lorain County Sales Tax Calculator with Accumulation - see the included .exe. Note that you do not need to validate the input from the user. You may start with the earlier tax calculator homework assignment that you completed if you want to. Be sure to refer to the general programming guidelines listed above. Name your project:

SalesTaxCalcAccums - Your Name

Take all of your completed project folders above and copy them into a folder named:

CalcAccumProblems - Your Name

Zip up this folder and submit the zip file in this dropbox by the due date and time.