CISS 160 Homework – Decision Structure Problems A

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 details of your Testing approach and different specific 'Test Cases' 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, comparing to another source (eg. used a calculator, Excel to compare) 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 – Decision Structure Problems A

1. (30 Points) Circle Information – see the included .exe. Use the Math.PI constant for the value of PI. Also, use the Math.Pow method in the calculation for the area of the circle. Use the TryParse() method previously discussed to validate the data entered. This is meant to give you practice using the .NET Math library.   Be sure to refer to the general programming guidelines listed above. Name your project:
MathLibCircleInfo - Your Name

2. (80 Points) Tiered baseball player salaries. Create a program that will calculate the salary a player will earn based on the following business rules. Be sure to refer to the general programming guidelines listed above. Name your project:
TieredPlayerSalaryViaIF - Your Name

Tier 1: For 1 through 49 hits in a season, the player earns **$5,000** per hit for ALL hits.

Tier 2: For 50 through 99 hits, the player earns **$7,500** per hit for ALL hits.

Tier 3: For 100 through 149 hits, the player earns **$10,000** per hit for ALL hits.

Tier 4: For 150 or more hits, the player earns **$12,500** per hit for ALL hits.

Then consider All Star and MVP (most valuable player) awards:

- If a player is neither an All Star NOR a MVP designation, then add 0% to the tiered salary you calculated above

- If a player is both an All Star AND and a MVP, then add 15% to the tiered salary you calculated above

- If a player is only one of All Star OR a MVP, then add 10% to the tiered salary you calculated above

Note that this is a 'simple' calculation - in that once a player reaches the threshold for the next $ level per hit, the player earns that rate for ALL hits they had. For example, if Player1 had 149 hits in a season, he will earn
149 * 10,000 = $1,490,000. However, if that Player1 would have produced one more hit (150), he would have earned
150 * 12,500 = $1,875,000. Then, consider the impact of the All Star and MVP attributes on the salary calculation mentioned above.

INPUT: Have the user enter the player's name, number of hits, one checkbox to indicate if they are an All Star and a second checkbox to indicate if they are a MVP. Then use a Calculate button to process the data. Process the data only if the user enters a player's name (any non-space characters typed in constitutes entering a name). Process the data only if the user entered a valid hit count. A valid hit count is an integer that is greater than zero, but is less than or equal to 300. Validate the hit count programmatically – do not use a masked textbox. Use a validation method we covered other than *try..catch* to perform the validation. Add a Clear button to clear the data entry fields, checkboxes and calculations.

PROCESSING AND OUTPUT: Calculate the player's earnings based on the Tier they achieved and display the calculated salary and Tier number. Use an *if () …* decision structure to determine the Tier number and $ per hit amount (do not use a *switch (..case…)* decision structure). Determine any additional salary amount based on their All Star/MVP checkbox settings.

In addition to allowing the entry of the player name, hits details and calculating the current player's salary based on the above, also display a separate summary set of information on the window showing the **leading player's** name , number of hits, salary and Tier number for the player who has had the **most** hits of all players processed so far, since the Window was opened or since the Clear button was last clicked. If there are one or more ties, the *first player* entered who had the most hits will be the leader.

3. (20 Points Optional Extra Credit) Read this assignment at least 5 days ahead of when it is due.   If today's date is 5 or more days before this assignment's due date, login to Angel and send me an Angel email titled 'Homework – Decision Structures A - I read it!' saying that you read through this assignment at least 5 days in advance of it being due. If *for example*, the assignment is due on a Friday 1/15 and today is the Sunday 1/10 prior then that is considered 5 days prior to the due date. Monday 1/11 or after does not qualify since it is only 4 days prior to the due date.

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

DecisionStructureA - Your Name

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