CISS 160 Homework – Loops, Listboxes, Division via Mod

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 – Loops, Listboxes, Division via Mod

NOTE: Modulo division (quotient and remainder division) is not covered in the textbook but it is an important aspect of any programming language. I describe below the basics of how it works.

Segment of C# code

….

int myQuotient = 0, myRemainder = 0;

myQuotientInt = 100 / 3; //this will return a value of 33 for the quotient

myRemainderInt = 100 % 3; //this will return a value of 1 for the remainder

….

The result of the above is the Quotient is 33 and the Remainder is 1,

meaning that the answer to 100 divided by 3 can be stated as "33 Remainder 1"

1. (50 Points) Decimal (Real Number) Division vs. Modulo (Quotient-Remainder) Division in C# Create a C# windows form that demonstrates the differences between division operations using decimal division vs. modulo division. ***Use the enclosed .exe as an example and create a program that looks and functions the same.*** Be sure to refer to the general programming guidelines listed above. Name your project:

DivisionDecVsMod - Your Name

Decimal division results in a real number (eg. 100.0 divided by 3.0 = 33.333333). Modulo division results in a quotient, remainder (eg. 100 divided by 3 = Quotient:33 Remainder:1. Computer programming languages typically provide the ability to perform both types of division. This Form will show the result of dividing one number by another as a 1) Quotient/Remainder result and as a 2) Real number/decimal result.

a) Setup two text boxes to hold the two numbers to be divided (the Dividend by the Divisor)

b) Add a 'Calculate' Button

c) Upon clicking the button, validate the two text boxes using int.TryParse… to verify that a number was entered into each text box (the need for validation should be becoming 'second nature' to each of you!)

d) Also calculate the result of the division using decimal division (think: use the / operator) and display in decimal format with 4 decimals (eg. 33.3333) in one label on the form; you may also need to explicitly cast the integers as decimal via code like myResultDecimal = (decimal)myInt1 / (decimal)myInt2;

e) Also calculate the result of the division using modulo division [think: this is a combination of two statements, one using the / division operator, and the other using the % division-remainder operator) and display 1) the resulting Quotient value in a label on the Form and also 2) the resulting Remainder value in different label on the form (eg. Quotient:33 Remainder:1]

f) Include a Clear All button that clears all the textboxes and labels

(see next page)

CISS 160 Homework – Loops, Listboxes, Division via Mod

2. (75 Points) Looping and ListBoxes. The C# language provides various aspects of the language to handle 'looping' through a defined set of C# statements within braces {…} over and over until some end point is reached. Also, there is a ListBox control that can be added to any Form allowing for multiple 'items' (or lines) of text to be displayed in one ListBox control on a Form**. *Use the enclosed .exe as an example and create a program that looks and functions the same*.** Be sure to refer to the general programming guidelines listed above. Name your project:

LoopListBoxMod - Your Name

This Form will allow the User to type in a Starting Year and an Ending Year value. Also the User can check a Checkbox to indicate if they want to know of any year in which a Presidential Election will occur, and a Checkbox to indicate if they want to know of any year in which a Federal Census will occur. When the 'Loop Through the Years' button is clicked, your program should loop, starting with the Start Year value and going to the End Year value, and display each Year that is looped through in the listbox, along with text describing if that a given year is a Presidential Election year and/or a Census year, if applicable.

a) Setup two text boxes to hold the Starting Year and Ending Year; setup two Checkboxes for the user to indicate if they want to know if a given year is 1) a Presidential Election year and/or 2) a Census year

b) Add a 'Loop Through the Years' Button

c) Upon clicking the button, validate the two text boxes using int.TryParse… to verify that a number was entered into each text box (the need for validation should be becoming 'second nature' to each of you!)

d) Loop through all years starting with the Starting Year and ending with the Ending Year using any C# looping structure [think: use a *for (…)* OR use a *while(…)* looping structure]

e) As each year is processed in the loop, display a line (add an Item) to the ListBox control showing:

1) The current Year being processed, along with

2) If the Presidential Election Checkbox is checked, and if the Year being processed is a Presidential Election Year (think: evenly divisible by 4… think Remainder % operator) then also display ", is an Election Year", then

3) If the Census Year Checkbox is checked, and it the Year being processed is a Census Year (think: evenly divisible by 10… think Remainder % operator) then also display ", is a Census Year"

4) For Example: "Year 2000, is an Election Year, is a Census Year"

e) NOTE: the determination of the Election and/or Census status of a given Year requires NO HARD CODING of any specific year numbers in your if() statements. These determinations should be handled using modulo division (think: remainder division using the % operator)

– DO NOT HAVE 'LONG' if() CONDITIONS LIKE THIS IN YOUR CODE: if (currentYear == 2000 || currentYear == 2010 || currentYear = 2020 etc etc) – think: 'how can I use a Remainder calculation/variable to figure out the Election Year, Census Year?'

f) Include a Clear All button that clears all the textboxes and labels and/or listboxes

3. (20 Points Optional Extra Credit) Read this assignment at least 4 days ahead of when it is due. If today's date is 4 or more days before this assignment's due date, login to Angel and send me an Angel email titled 'Homework – Loops Listboxes Mod - I read it!' saying that you read through this assignment at least 4 days in advance of it being due. If *for example*, the assignment is due on a Wednesday 11/6 and today is the Saturday 11/2 prior then that is considered 4 days prior to the due date. Sunday 11/3 or after does not qualify since it is only 3 days prior to the due date.

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

DivLoopList - Your Name

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