FINAL EXAM – TEST 02

Due Date: Mid-night (11.59 pm) Monday 13th April, 2020 ***Marks/Weightage: 50/25%***

Student Name/ID: _____

Instructions: Be sure to read the following general instructions carefully:

This take home Test 02 should be completed individually by all the students. You will have to demonstrate your solution in a scheduled lab session and submitting the Test02 through drop box link on eCentennial on or before the due date.

At the start, you must name your Visual Studio 2019 solution name according to the following rule:

FirstName-LastName_COMP212_SectionNumber_Test02

For Example: John-Smith_COMP212_Sec003_Test02 (say if your section number is 003)

>> And your **project name** should be as follows: FirstName-LastName_SectionNumber _ExerciseNumber

For Example: John-Smith_Sec003_Exercise01

>> If your take home Test 02 assignment has more than one exercise, then each subsequent exercise should be added to the same solution created above and named as firstname-last-name_exercise2, firstname_last-name_exercise3 etc.

>> After you complete, exit Visual Studio and go to solution folder, zip it up and you will get the following zip file. FirstName_LastName_COMP212_SectionNumber_Test02.zip

For example: John_Smith_COMP212_Sec003_Test02.zip (if your section is 003..)

>> Apply the naming conventions for variables, methods, classes, and namespaces:

- variable names start with a lowercase character for the first word and uppercase for every other word

- classes start with an uppercase character of every word

- namespace use only lowercase characters

- methods start with an uppercase character for the first word and uppercase for every other word

Note: Late submissions are not accepted. You must implement exception handling in all the exercises.

Exercise 01: Databases which you need for this take home test02 are on e-centennial. ***[10 marks]***

Create a **Windows Form App/WPF** to manipulate the data within **Books** database and displays the results of the following queries: *(Refer exercise 01 of Lab Assignment 04)*

a. Get a count of all the authors grouped by title, sorted by title. It should display a title and number of authors who have written that title.
b. Get a list of all the titles grouped by author name, sorted by author; for a given author name sort the titles alphabetically.

Exercise 02: Following is enhancement of exercise 02 of Lab Assignment 04. ***[15 marks]***

*[Build **Win Form/WPF app** to execute a query against the **Players** table of the **Baseball** database. Display the result in a DataGrid, and add a TextBox and Button to allow the user to search for a specific player by last name. Use a Label to identify the TextBox. Clicking the Button should execute the appropriate query. Also, provide a Button that enables the user to return to browsing the complete set of players.]*

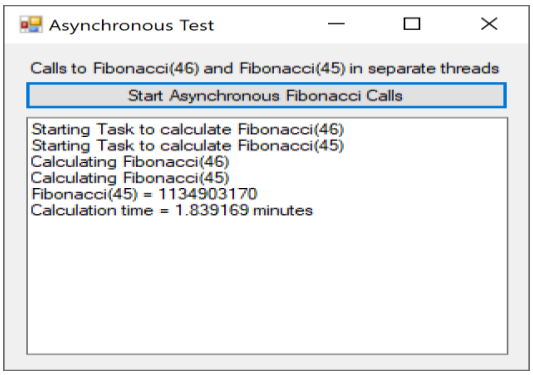
Now in addition to above which is exercise 02 of Lab 04, you need to add the following:

a. Search a record by the Player ID. (add corresponding label and textbox to the existing GUI)
b. Display the batting average of all the players. Add appropriate control e.g. a button to the GUI
c. Display a name of player who has got the highest batting score.

Note: Your Baseball database should have minimum 5 records/players

Exercise 03: This is based on Asynchronous programming. ***[25 marks]***

Build a **Windows Form App/WPF** (refer the screen shot below. This code example is covered in the class where we are calling **asynchronously** the Fibonacci method twice with input value of 46 and 45. Also we set up the start time and end time so that we can calculate the time taken by each.



So in this exercise 03, instead of call to Fibonacci(46), you need to define and call Factorial of the number 46, e.g. **Factorial(46).**Factorial function you have implemented in the Lab assignment 05.

And you need to define and call one more calculation intensive method –void **RollDie**(int number) which displays that face of a die which appeared highest number of times. For example if you roll a die (die is six faced cube) 10 times, then your function RollDie will display that which of (1, 2,3,4,5,6) have appeared maximum number of times. Input to RollDie should be 60,000,000. *[Hint: You need to use random function. Also you can use an array where index value is face and value is frequency.]*

So you will be calling the these three methods asynchronously ..

Calculating Factorial(46)

Calculating Fibonacci(45)

Calculating RollDie(60000000).