COMP228 – Java Programming

Mid-Term Exam
Console Lottery Number Picker

Due Week #7 (October 27, 2015) by end of class

Value 15%

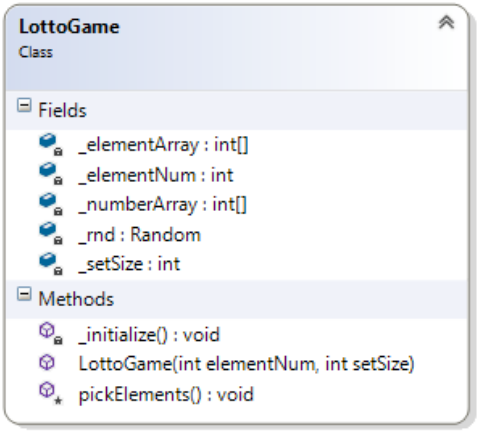
Console Lottery Number Picker **Maximum Mark: 30**

Overview: Create a console app that **randomly picks 6 lottery numbers from 1 to 49 (e.g. 6/49)** and unique **bonus number**. The program will randomly generate 5 sets of lottery numbers and display them on the console.

Instructions:

(30 Marks: Functionality)

1. Write an abstract class called **LottoGame** that has the following shape (12 Marks Functionality):



   a. The Instance Variable **`_numberArray`** should hold an array of integers from 1 to the size of the lotto game **set size** (up to 50 should cover it). Assume that you may have other lotto games in future iterations of your program (2 Marks: functionality).
   b. You will initialize the **`_numberArray`** with numbers from 1 to **`_setSize`** (i.e. 1 to 49) using a loop in the **`_initialize`** method (2 Marks: Functionality).

c. The Instance Variable **`_elementArray`** will contain a number of elements equal to the **`_elementNum`** Instance Variable and will contain your final numbers (from 1 to **`_setSize`**) (2 Marks: Functionality)
d. These numbers in the **`_elementArray`** will be randomly generated in the **`public pickElements`** method. These numbers must not repeat (i.e. you can't have two numbers that are the same in your list of lotto numbers) (4 Marks: Functionality).
e. The **`constructor`** method of the abstract class will take two parameters: **`elementNum`** (which will tell the program how many elements are in the lotto game) and **`setSize`** (which indicates how large the set is – e.g. 1 to 49) (2 Marks: Functionality).

2. Create an interface **`IBonusNumber`** that outlines a **`public abstract bonusNumber`** method. The method should return an **`Integer`** (4 Marks: Functionality).
3. Create a subclass that extends the **`LottoGame`** abstract super class and implements the **`IBonusNumber`** interface. This class should be named appropriately (i.e. **`Lotto649`**) (8 Marks: Functionality).
    a. This constructor method of the derived class will include a call to the superclass that will generate 5 sets of lotto numbers with the appropriate **element number** (i.e. **6** numbers) of the appropriate **set size** (i.e. from 1 to 49) (2 Marks: Functionality).
    b. Ensure that you create a **bonusNumber** method that picks a random number from 1 to **`setSize`** and that is not a duplicate of the lotto numbers that have already been generated from the inherited **`pickElements`** method (4 Marks: Functionality).
    c. Override the built in **`public toString`** method so that it returns a string of properly formatted lottery numbers that is generated from the inherited **pickElements** method and the bonus number from the **`bonusNumber`** method. (2 Marks: Functionality).
4. In your main method of your driver class – instantiate an object of the above subclass whenever the user runs the program (i.e. create a **Lotto649** object which outputs 5 sets of 6 numbers from 1 to 49) (4 Marks: Functionality).
5. Ensure your output is properly formatted so that each set of numbers are output to separate lines with a comma and a space between each number in the set (e.g. 3, 15, 26, 38, 45, 51) (1 Marks: Functionality). (**hint**: use the **`toString`** method of the **`Lotto649`** object) (2 Marks: Functionality).

SUBMITTING YOUR WORK

Your submission should include:

1. A zip archive of your Project files uploaded to e-centennial.

Please zip all files in to a single project archive.

This assignment is weighted **15%** of your total mark for this course.