CSC 230: Elementary Data Structures and Algorithms

Assignment Six

Objective:

To work with collections
To work with ArrayList
To work with Collections methods
To work in teams

This is a team assignment.

You can go to **People > Groups** and search for your name to see your group. If you click on the down arrow to the left of the group name, you will be shown all of the group members. You do not need to use the Canvas groups. You can get together any way you wish to – using Zoom, Slack, Teletype, Codeshare, Miro, Google Docs, Microsoft Teams, etc. You can, of course, chose to meet in a computer lab at UNCG or at someone's house. It is up to you how you collaborate.

Part I

Assignment:

Create a package called **homeworkSix.java**.

The program will generate a lottery of a three-digit integer. The program will prompt the user to enter a three-digit integer and determine whether the user wins according to the following rules:

1. If the user input matches the lottery number in the exact order, the award is $10,000.
2. If all digits in the user input match all digits in the lottery number, the award is $3,000.
3. If two digits in the user input matches two digits in the lottery number, the award is $2,000
4. If one digit in the user input matches one digit in the lottery number, the award is $1,000

The program *must* have these methods: generateNumber(), getGuess(),checkNumbers(), and collectWinners. You may choose to have other methods for the rest of the program, or you may do the rest in main.

- generateNumber() will generate a random whole number that is between 100-999.
- getGuess() will get a 3-digit whole number from the user.
- checkNumbers() will compare the numbers in the sorted lists to print out the correct information based on the criteria shown above.
- collectWinners() will be called in the checkNumbers() method if the user has won $10,000. After showing the "Exact match: You win $10,000!" message, the method will ask the user to enter his or her first and last name (into one variable). This name will be sent to the file called Winners.txt.

Notes:

- You do not need to add the Integrity Policy statement to this program.
- The program should be in a loop that allows the user to keep playing.
- The lottery number is one generated integer in between 100 -999. After it is generated, it will need to be placed into 3 separate variables.
- The user will be told to enter one (integer) 3-digit number. After it is entered, it will need to be placed into 3 separate variables.
- You will add the three digits in the lottery number and three digits in the user input into two lists.
- The team leader, and *only* the team leader, will be the person who ultimately will put together the pieces of the program and who will submit the files.
- You may choose to divide the work any way you and your teammates wish to do so. Do not, however, choose one person to write the program and another to do the summary. All of you must work on the program. In the "real" world, it is common to have one or two people to work on an individual function.
- I have added a grading rubric on the bottom of this assignment. This will allow you to check your work to make sure you have met all the objectives.

Hints:

- You can use the modulus logic to extract the digits from an integer.
- You can use any of the Collection's methods (contains() and containsAll() are quite helpful)

Above the class header, write a javadoc comment containing the integrity policy statement. Under the integrity statement, add comments on your program that show the title of the program (Homework Six), your first and last name, and the course name (CSC 230, Sec #).
As the first print lines for your program under the main method, make sure that the run screen also prints the title of the program (lastNameFirstInitialHW2), your first and last name, and the course name (CSC 230, Sec #), and the program information as shown in the sample run screen.

Part II

You will write a summary of the project & save it as **Homework Six**. The summary is a short analysis of the problem written in Word. If you use some other word processor, you will need to save the file as a Word file. Use the mini lecture called *The Summary* that is in the **Help Files** module for help in doing this. Your team's summary will include the:

1. problem definitions
2. how inputs are obtained
3. how output will be presented
4. the formulas
5. any problems you had
6. your successes
7. the time required for you to complete the project (this is the total time taken)
8. which team member worked on which part of the program
9. any unusual happenings

The summary will be shown in sections. You will use titles for each section.

How to Turn in the Assignment

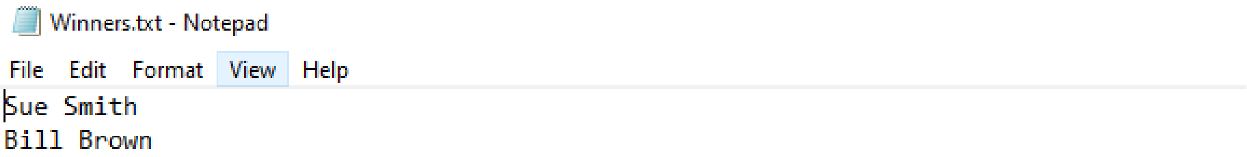
To submit this assignment, **only the team leader will submit**. He or she will upload the following files:

1. The package called **homeworkSix.zip**
2. The summary called **Homework Six**. This *must* be a Word document.

SAMPLE RUN SCREEN

```
Christine Kikuchi
Homework 6

This program will play a lottery with the user.

Enter your lottery pick (three digits, zeros are fine, as are duplicates): 630
The lottery number is 630
Exact match: You win $10,000!
Please enter your first name and last name, separated by a space: Sue Smith

Would you like to play again? Y
Enter your lottery pick (three digits, zeros are fine, as are duplicates): 555
The lottery number is 691
Sorry, you did not match the numbers: Try again.

Would you like to play again? y
Enter your lottery pick (three digits, zeros are fine, as are duplicates): 348
The lottery number is 453
You matched two of the numbers: You win $2,000!

Would you like to play again? Y
Enter your lottery pick (three digits, zeros are fine, as are duplicates): 467
The lottery number is 476
You matched all of the numbers: You win $3,000!

Would you like to play again? y
Enter your lottery pick (three digits, zeros are fine, as are duplicates): 499
The lottery number is 454
You matched one of the numbers: You win $1,000!

Would you like to play again? y
Enter your lottery pick (three digits, zeros are fine, as are duplicates): 174
The lottery number is 174
Exact match: You win $10,000!
Please enter your first name and last name, separated by a space: Bill Brown

Would you like to play again? n
```

SAMPLE OUTPUT FILE: Winners.txt



```
Sue Smith
Bill Brown
```

Grading Rubric

Task	Point Value
Summary	
Lists the team member's full names	.5
Has all 9 parts: (.5 points each) 1. problem definitions 2. how inputs are obtained (uses input variable names) 3. how output will be presented (uses output variable names) 4. the formulas 5. any problems you had 6. your successes 7. the time required for you to complete the project (total time includes summary and collaboration time) 8. which team member worked on which part of the program 9. any unusual happenings	4.5
Program	
The generateNumber() method generates a random whole number that is between 100-999.	5
The getGuess() method gets a 3-digit number from the user.	5
The checkNumbers()compares the numbers in the sorted lists to print out the correct information based on the criteria. The method uses the correct parameter(s).	7
The collectWinners() method is called by the checkNumbers() method if the user has won $10,000. After showing the "Exact match: You win $10,000!" message, the method asks the user to enter his or her first and last name (into one variable). The winner's names are sent to the file called Winners.txt. The method uses the correct parameter(s).	7
The program handles file checking	2

The main program uses the correct loop to allow the user to play again.	2
The main program prints the teammate's full names, the homework #, and the program information.	2
The generateNumber() method is called correctly	5
The getGuess() method is called correctly	5
The checkNumbers() method is called correctly	5
The collectWinners() method is called correctly	5
The random number is split into 3 variables	7
The number entered by the user is split into 3 variables	7
An ArrayList is created and used to add the random number variables	7
The list is sorted via the Collections method	6
An ArrayList is created and used to add the user's guess variables	7
The list is sorted via the Collections method	6
The programs output to screen and file are done like the sample.	5
Total for program	**100**

Your program must compile.
Programs that do not compile or run will not be accepted.