CSC260 Homework 8 (CH11)

- 50 points in total
  - 10 points from MyProgrammingLab questions (2 points per each question you select)
    - You need to solve Ch10 and Ch11 questions.
  - 20 points from class activities
  - 20 points from Programming questions.
- Submit your homework at Canvas/Assignments/HW8
- If you have any questions about the homework, please send me an email (start with CSC260 with a section number) or open a Discussion on Canvas.

3. Programming (20 points)

1. You should make the program that returns correct answer.
2. You should print out the results and paste them as a comment.
3. For getting outputs, use the inputs from the sample run. Some programs don't need an input, then print out and copy the results.
4. Copy only Java files for submission; copy only the Java files in the `programming` directory.
5. Students earn 100% when they get correct answers and copied results, 60% when they get wrong answers, 0% when they can't compile the Java source or no answers copied.

3.1 MyStackTest.java (10 points)

Reimplment the stack using `ArrayList<>()`. This class should support the following methods.

1. public boolean isEmpty()
2. public int getSize()
3. public Object peek()
4. public Object pop()
5. public void push(Object o)
6. public String toString()

You should make the program to print all trues.

Hint

You need to implement each method as follows.

```
public boolean isEmpty() {
    return list.isEmpty();
}

public int getSize() {
    return list.size();
}

public Object peek() {
    return list.get(getSize() - 1);
}
...
```

3.2 RemoveDuplicate.java (10 points)

Reimplment the RemoveDuplicate class using `ArrayList`. You should make the program to print all trues.

Hint 1

You can use two-step process to make problem easy.

1. Find the values that are unique.
2. Put the unique values to a new ArrayList.

```
ublic static void removeDuplicate(ArrayList<Integer> list) {
   ArrayList<Integer> temp = new ArrayList<Integer>();

   // step1: temp doesn't have anything duplicate
   for (int i = 0; i < list.size(); i++)
       if (!temp.contains(list.get(i)))
           temp.add(list.get(i));

   // step2: list has the temp
   list.clear();
   for (int i = 0; i < temp.size(); i++)
       list.add(???)
```

Hint 2

- I added the programming answer (RemoveDuplicate-answer.java), but use this answer only when you spend too much time on this homework. I ask you to solve this question on your own or with your friends.