CSC260 Homework 4 (CH7)

- 50 points in total
  - 10 points from MyProgrammingLab questions (2 points per each question you select)
  - 20 points from class activities
  - 20 points from Programming questions.
- Submit your homework at Canvas/Assignments/HW4
- 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.

Programming

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 Reverse.java (5 points)

Don't forget that when a method returns an array, it actually returns the reference of the array. In this program, you should make a reverse method that returns the reverse of the original input. For example, this main method returns a new array list2 with elements of {6, 5, 4, 3, 2, 1}. And print out the values.

```
public static void main(String[] args) {
    int[] list1 = {1, 2, 3, 4, 5, 6};
    int[] list2 = reverse(list1);
    for (int result: list2) {
        System.out.println(result);
    }
}
```

- You can use the skeleton code, or you can make your program from scratch.
- You can use a hint from Liang's text book in the section "Returning an array from a method."
- Your program should print out the following: 6, 5, 4, 3, 2, 1

3.2 IndexOfSmallestElement.java (5 points)

Write a method that returns **the index** of the smallest element in an array of integers. Use the following signature as your method: `public static int indexOfSmallestElement(int[] array)`

- You can use the skeleton code, or you can make your program from scratch.
- Use a list of input `int[] list = {5,1,4,5,2};` The index of smallest of element is `1` in this example.
- Your program should print out the following:

```
The smallest index is 1
```

3.3 EliminateDuplicates.java (10 points)

Write a method that returns a new array by eliminating the duplicate values in the array using the following signature as your method: `public static int[] eliminateDuplicates(int[] list)` Write a test program that reads in ten integers, invokes the method, and displays the result. Here is the sample run of the program:

```
Enter ten numbers: 1 2 3 2 1 6 3 4 5 2
The distinct numbers are: 1 2 3 6 4 5
```

Use the following hints, but I highly recommend you to solve the problem by youself. Use the hints only when you absolutely need them.

Hint1 - your program needs a search method

```
public static int linearSearch(int[] list, int key) {
    for (int i = 0; i < list.length; i++) {
        if (key == list[i])
            return i;
    }
    return -1;
}
```

Hint2 - your program needs two arrays First array (temp) is used to find the non-duplicate numbers. The second array (result) is to return the non-duplicate numbers.

```
public static int[] eliminateDuplicates(int[] numbers) {
    int[] temp = new int[numbers.length];
    ...
    // Copy the actual number of elements from temp to result
    int[] result = new int[size];
    for (int i = 0; i < size; i++)
        result[i] = temp[i];

    return result;
}
```

Hint3 - this is the overall skeleton of your code

```java
public class EliminateDuplicates {
    public static int[] eliminateDuplicates(int[] numbers) {
        int[] temp = new int[numbers.length];
        int size = 0;
        for (int i = 0; i < numbers.length; i++) { // For each number in numbers array
            if (linearSearch(temp, numbers[i]) == -1) {
                ...
            }
        }

        // Copy the actual number of elements from temp to result
        int[] result = new int[size];
        for (int i = 0; i < size; i++)
            result[i] = temp[i];

        return result;
    }

    /**
     * The method for finding a key in the list
     */
    public static int linearSearch(int[] list, int key) {
        ...
    }

    public static void main(String[] args) {
        int[] numbers = new int[10];

        java.util.Scanner input = new java.util.Scanner(System.in);
        System.out.print("Enter ten integers: ");

        for (int i = 0; i < numbers.length; i++) {
            // Read and store numbers in an array if it is new
            numbers[i] = input.nextInt();
        }

        int[] result = eliminateDuplicates(numbers);

        System.out.println(
                "The number of distinct integers is " + result.length);
        System.out.print("The distinct integers are ");
        for (int e : result)
            System.out.print(e + " ");
    }
}
```