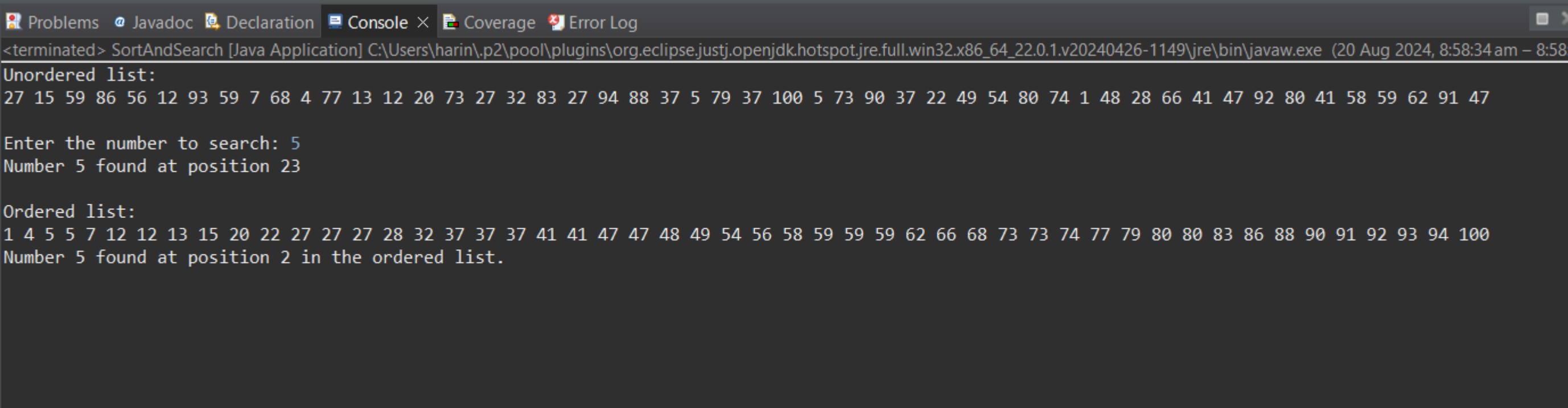```java
// Harini. R 192324108
//JP section-3 question-1
package test;

import java.util.Random;
import java.util.Scanner;

public class SortAndSearch {
    public static void main(String[] args) {
        int[] numbers = new int[50];
        Random rand = new Random();
        for (int i = 0; i < numbers.length; i++) {
            numbers[i] = rand.nextInt(101);
        }
        System.out.println("Unordered list:");
        displayArray(numbers);
        Scanner scanner = new Scanner(System.in);
        System.out.print("\nEnter the number to search: ");
        int searchValue = scanner.nextInt();
        int position = linearSearch(numbers, searchValue);
        if (position != -1) {
            System.out.println("Number " + searchValue + " found at position " + position);
        } else {
            System.out.println("Number " + searchValue + " not found.");
        }
        bubbleSort(numbers);
        System.out.println("\nOrdered list:");
        displayArray(numbers);
        position = linearSearch(numbers, searchValue);
        if (position != -1) {
            System.out.println("Number " + searchValue + " found at position " + position + " in the ordered list.");
        } else {
            System.out.println("Number " + searchValue + " not found in the ordered list.");
        }

        scanner.close();
    }
    private static void displayArray(int[] array) {
```

```java
        }
        System.out.println();
    }
    private static int linearSearch(int[] array, int value) {
        for (int i = 0; i < array.length; i++) {
            if (array[i] == value) {
                return i;
            }
        }
        return -1;
    }
    private static void bubbleSort(int[] array) {
        int n = array.length;
        for (int i = 0; i < n - 1; i++) {
            for (int j = 0; j < n - 1 - i; j++) {
                if (array[j] > array[j + 1]) {
                    int temp = array[j];
                    array[j] = array[j + 1];
                    array[j + 1] = temp;
                }
            }
        }
    }
}
```

```
Unordered list:
27 15 59 86 56 12 93 59 7 68 4 77 13 12 20 73 27 32 83 27 94 88 37 5 79 37 100 5 73 90 37 22 49 54 80 74 1 48 28 66 41 47 92 80 41 58 59 62 91 47

Enter the number to search: 5
Number 5 found at position 23

Ordered list:
1 4 5 5 7 12 12 13 15 20 22 27 27 27 28 32 37 37 37 41 41 47 47 48 49 54 56 58 59 59 59 62 66 68 73 73 74 77 79 80 80 83 86 88 90 91 92 93 94 100
Number 5 found at position 2 in the ordered list.
```