Week 5 sample program

1.

```
Question 1
Correct
Mark 1.00 out of 1.00
 Given an array A of positive integers, let 5 be the sum of the digits of the minimal element of A.
 Return 0 if 5 is odd, otherwise return 1.
 Example 1:
 Input:
 34 23 1 24 75 33 54 8
 Output:
 Explanation:
 The minimal element is 1, and the sum of those digits is S = 1 which is odd, so the answer is 0.
 Example 2:
 Input:
 99 77 33 66 55
 Output:
 Explanation:
 The minimal element is 33, and the sum of those digits is 5 = 3 + 3 = 6 which is even, so the answer is 1.
 Constraints:

    1 <= Alength <= 100</li>

 1 <= A[i] <= 100</li>
```

Program:

```
import java.util.Scanner; public class
MinElementDigitSum {
   public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of elements in the array: ");
   int size = scanner.nextInt();        if (size <= 0) {
            System.out.println("Array size must be greater than zero.");
            return;
        }
        int[] array = new int[size];</pre>
```

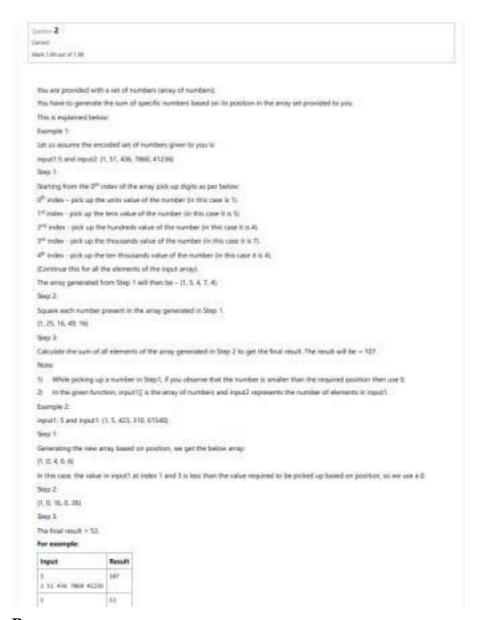
```
System.out.println("Enter the elements of the array:");
    for (int i = 0; i < size; i++) {
array[i] = scanner.nextInt();
    }
    int minElement = array[o];
    for (int i = 1; i < size; i++) {
                                      if
(array[i] < minElement) {</pre>
minElement = array[i];
      }
    }
    int sumOfDigits = 0;
                            int
number = minElement;
                            while
(number > 0) {
                     sumOfDigits
+= number % 10;
      number /= 10;
    }
    System.out.println(sumOfDigits % 2 == 0 ? 1 : 0);
 }
}
```

Output:

```
Enter the number of elements in the array: 8
Enter the elements of the array:
34
23
1
24
75
33
54
8
0
Enter the number of elements in the array: 5
Enter the elements of the array:
99
77
33
```

2.

66 55



Program:

```
import java.util.Scanner; public class
DigitSumCalculator {      public static void
main(String[] args) {
            Scanner scanner = new Scanner(System.in);
            System.out.println("Enter the number of elements in the array:");
int n = scanner.nextInt();      int[] input = new int[n];
System.out.println("Enter the elements of the array:");
      for (int i = 0; i < n; i++) {
input[i] = scanner.nextInt();
    }</pre>
```

```
int finalSum = calculateFinalSum(input);
    System.out.println("Final result = " + finalSum);
  }
  public static int calculateFinalSum(int[] input) {
                      for (int i = 0; i < input.length;
int finalSum = 0;
i++) {
             int currentNumber =
               int digitPosition = i + 1;
input[i];
                                               int digit =
getDigitAtPosition(currentNumber, digitPosition);
                                                          finalSum
+= digit * digit;
    }
    return finalSum;
  }
  public static int getDigitAtPosition(int number, int position) {
                                                                     String
                                             int length =
numberStr = Integer.toString(number);
numberStr.length();
                         if (length < position) {
      return o;
    }
    char digitChar = numberStr.charAt(length - position);
return Character.getNumericValue(digitChar);
  }
}
```

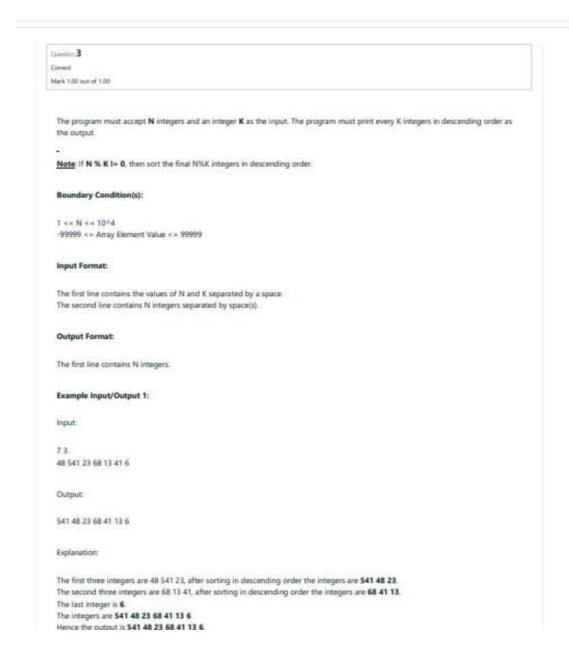
Output:

```
Enter the number of elements in the array:

5
Enter the elements of the array:

1
5
423
310
61540
Final result = 53
```

3.



Program:

import java.util.Arrays; import
java.util.Scanner; public class SegmentSorter
{ public static void main(String[] args) {

```
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the values of N and K:");
int N = scanner.nextInt(); int K = scanner.nextInt();
    System.out.println("Enter the " + N + " elements:");
                                                              int[]
                      for (int i = 0; i < N; i++) {
arr = new int[N];
arr[i] = scanner.nextInt();
    }
    for (int i = 0; i < N; i += K) {
int end = Math.min(i + K, N);
Arrays.sort(arr, i, end); reverse(arr,
i, end - 1);
    }
    for (int num : arr) {
      System.out.print(num + " ");
    }
  }
  public static void reverse(int[] arr, int start, int end) {
while (start < end) {
                           int temp = arr[start];
                                                        arr[start]
= arr[end];
                  arr[end] = temp;
      start++;
end---;
    }
  }
}
```

Output:

```
Enter the values of N and K:
7 3
Enter the 7 elements:
48
541
23
68
13
41
6
541 48 23 68 41 13 6
```