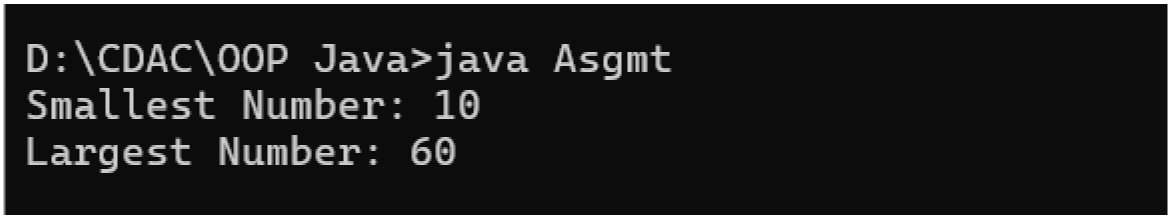
Name- Aarya Sanjay Dange

Assignment 03

1. Find the Largest and Smallest Element
   ○ Given an array, find the smallest and largest elements in it.

```
class Asgmt3{
	public static void main(String args[]){

		int[] arr={20,30,60,10,55};
		int max=arr[0];
		int min=arr[0];
		for(int i=0;i<arr.length;i++){
			if(arr[i]<min){
				min=arr[i];
			}
			if(arr[i]>max){
				max=arr[i];
			}
		}
		System.out.println("Smallest Number:
"+min);
```

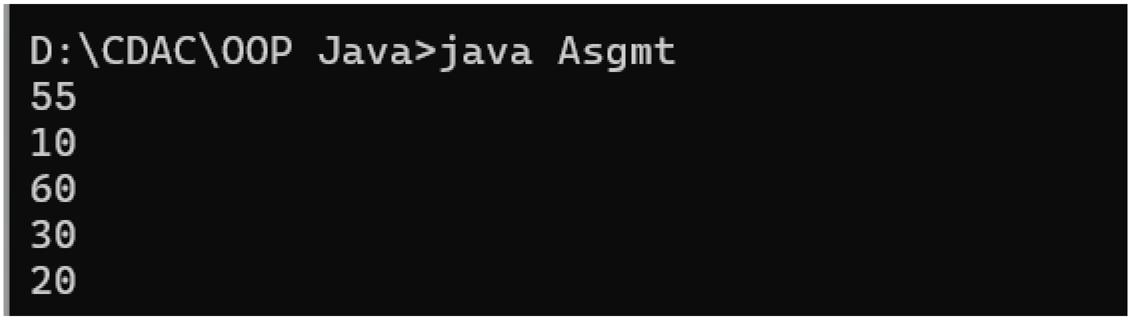
```
		System.out.println("Largest Number:
"+max);
	}
}
```



```
D:\CDAC\OOP Java>java Asgmt
Smallest Number: 10
Largest Number: 60
```

2. Reverse an Array
○ Reverse the given array in place.

```
class Asgmt{
	public static void main(String args[]){

		int[] arr={20,30,60,10,55};
		for(int i=arr.length-1;i>=0;i--){
			System.out.println(arr[i]);
		}
	}
}
```

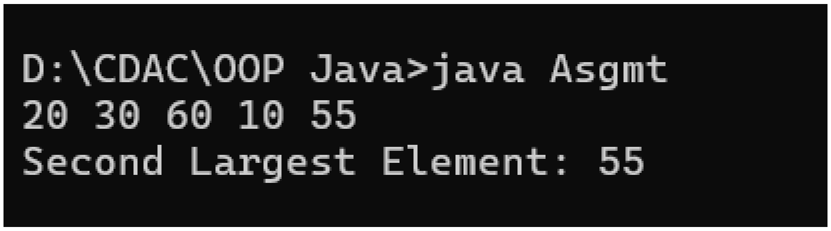


```
D:\CDAC\OOP Java>java Asgmt
55
10
60
30
20
```

3. Find the Second Largest Element

○ Find the second-largest element in the given array.

```
import java.util.*;
class Asgmt{
    public static void main(String args[]){

        int[] arr={20,30,60,10,55};

        for(int i=0;i<arr.length;i++){
            System.out.print(arr[i]+" ");
        }
        int num=arr.length-2;
        Arrays.sort(arr);
        System.out.println();
        System.out.println("Second Largest
Element: "+arr[num]);
    }
}
```



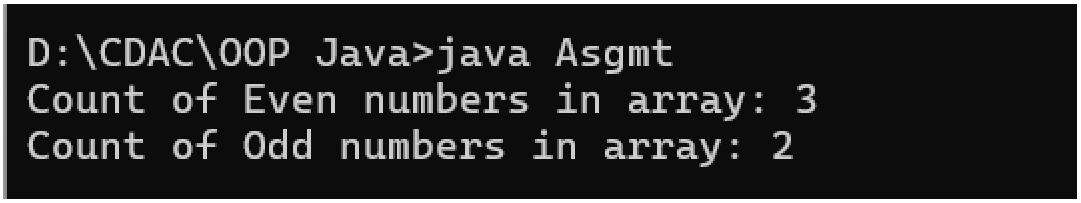
```
D:\CDAC\OOP Java>java Asgmt
20 30 60 10 55
Second Largest Element: 55
```

4. Count Even and Odd Numbers

○ Count the number of even and odd numbers in an array.

```
import java.util.*;
class Asgmt{
    public static void main(String args[]){

        int[] arr={20,30,63,10,55};
        int even=0;
        int odd=0;
        for(int i=0;i<arr.length;i++){
            if(arr[i]%2==0){
                even++;
            }
            else{
                odd++;
            }
        }
```

```
		System.out.println("Count of Even numbers in array: "+even);
		System.out.println("Count of Odd numbers in array: "+odd);
	}
}
```



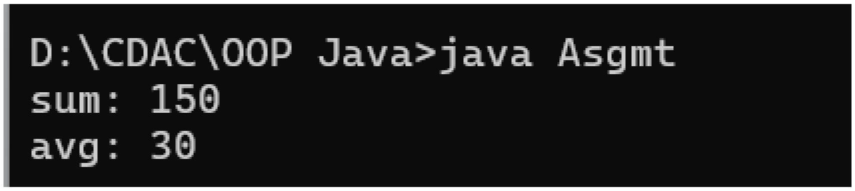
```
D:\CDAC\OOP Java>java Asgmt
Count of Even numbers in array: 3
Count of Odd numbers in array: 2
```

5. Find Sum and Average

○ Compute the sum and average of all elements in the array.

```
import java.util.*;
class Asgmt{
	public static void main(String args[]){

		int[] arr={20,30,40,10,50};
		int sum=0;
		for(int i=0;i<arr.length;i++){
			sum=sum+arr[i];
		}
		int avg=sum/arr.length;
```

```
        System.out.println("sum: "+sum);
        System.out.println("avg: "+avg);
    }
}
```

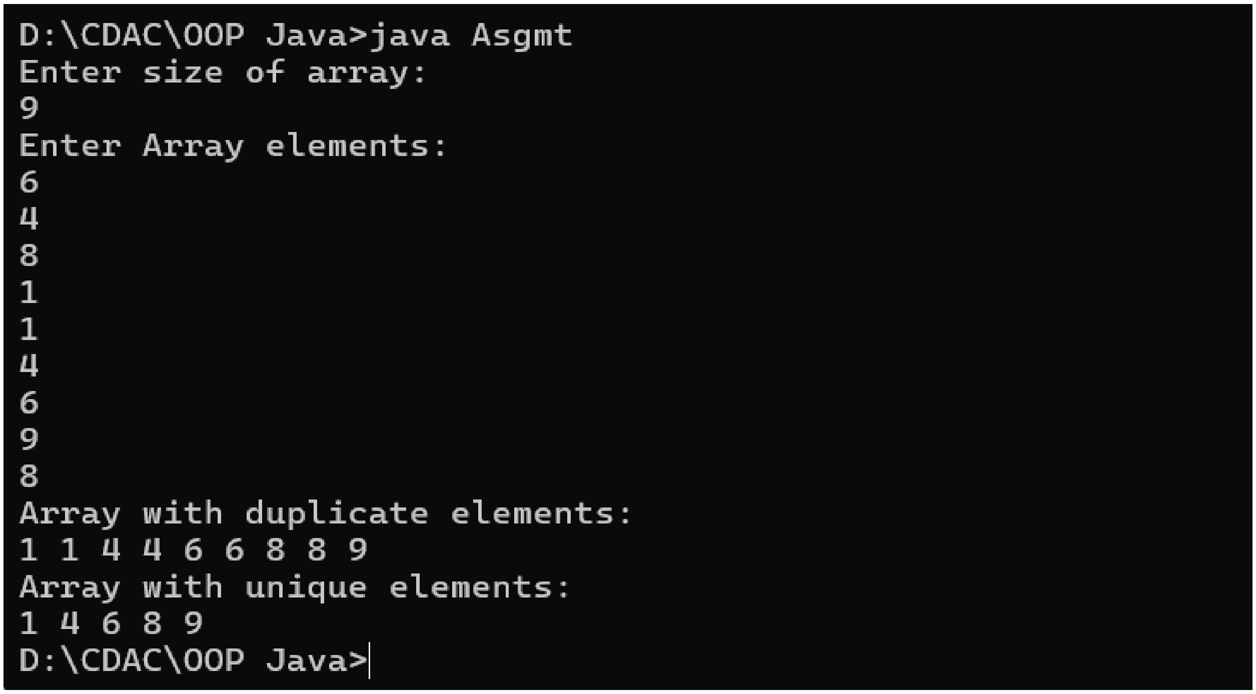


```
D:\CDAC\OOP Java>java Asgmt
sum: 150
avg: 30
```

6. Remove Duplicates from a Sorted Array

○ Remove duplicate elements from a sorted array without using extra space.

```
import java.util.*;
class Asgmt{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter size of array: ");
        int size=sc.nextInt();
        System.out.println("Enter Array elements: ");
        int arr[]=new int[size];

            for(int i=0;i<size;i++){
                arr[i]=sc.nextInt();
            }
            System.out.println("Array with duplicate
elements: ");
            Arrays.sort(arr);
            for(int i=0;i<arr.length;i++){
                System.out.print(arr[i]+" ");
            }
            System.out.println();
            System.out.println("Array with unique
elements: ");
            for(int i=0;i<arr.length-1;i++){
                if(arr[i]!=arr[i+1]){
                    System.out.print(arr[i]+" ");
                }
            }
            System.out.print(arr[size-1]);
    }
}
```

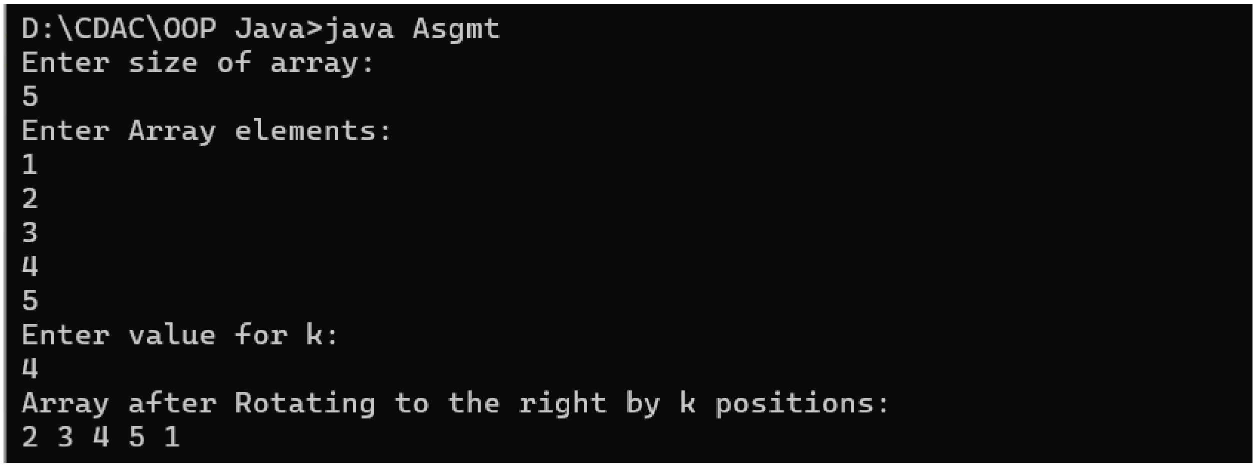


```
D:\CDAC\OOP Java>java Asgmt
Enter size of array:
9
Enter Array elements:
6
4
8
1
1
4
6
9
8
Array with duplicate elements:
1 1 4 4 6 6 8 8 9
Array with unique elements:
1 4 6 8 9
D:\CDAC\OOP Java>
```

7. Rotate an Array

○ Rotate the array to the right by k positions.

```
import java.util.*;
class Asgmt{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter size of array: ");
        int size=sc.nextInt();

        System.out.println("Enter Array elements: ");
        int arr[]=new int[size];
        int arr2[]=new int[size];
            for(int i=0;i<size;i++){
                arr[i]=sc.nextInt();
            }
        System.out.println("Enter value for k: ");
        int k=sc.nextInt();
        k=k%size;

        for(int i=0;i<size;i++){
                arr2[(i+k)%size]=arr[i];
        }
            System.out.println("Array after Rotating to the right by k positions: ");
            for(int i=0;i<size;i++){
                System.out.print(arr2[i]+" ");
            }
            System.out.println();
    }
}
```



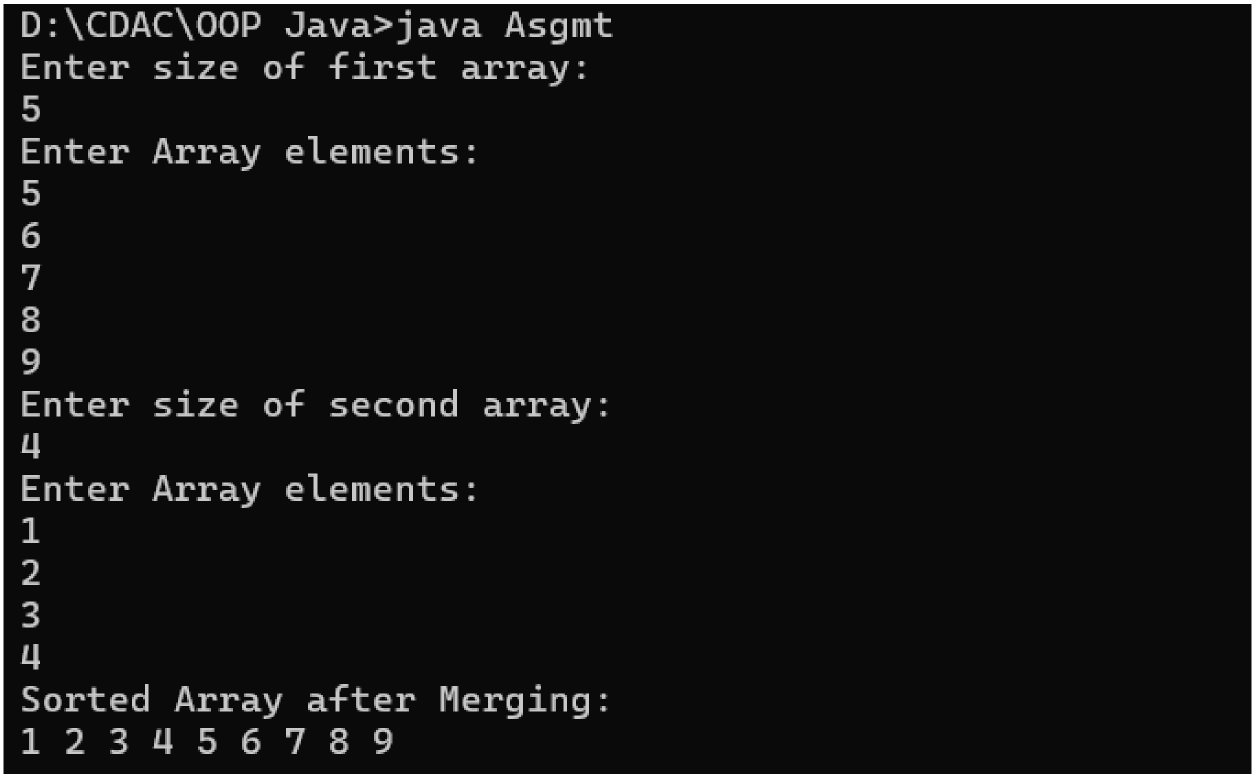
```
D:\CDAC\OOP Java>java Asgmt
Enter size of array:
5
Enter Array elements:
1
2
3
4
5
Enter value for k:
4
Array after Rotating to the right by k positions:
2 3 4 5 1
```

8. Merge Two Sorted Arrays

○ Merge two sorted arrays into a single sorted array without using extra space.

```
import java.util.*;
class Asgmt{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter size of first array:
");
        int size1=sc.nextInt();
        int arr1[]=new int[size1];
        System.out.println("Enter Array elements: ");

            for(int i=0;i<size1;i++){
                arr1[i]=sc.nextInt();
            }
            System.out.println("Enter size of second
array: ");
            int size2=sc.nextInt();
            int arr2[]=new int[size2];
            System.out.println("Enter Array
elements: ");
```

```
            for(int i=0;i<size2;i++){
                arr2[i]=sc.nextInt();
            }
            System.out.println("Sorted Array after
Merging: ");

            int res_size=size1+size2;
            int res[]=new int[res_size];
            for(int i=0;i<size1;i++){
                res[i]=arr1[i];
            }
            for(int
i=size1,j=0;i<(size1+size2);i++,j++){
                res[i]=arr2[j];
            }
            java.util.Arrays.sort(res);
            for(int i=0;i<res_size;i++){
                System.out.print(res[i]+" ");
            }
            System.out.println();
    }
}
```



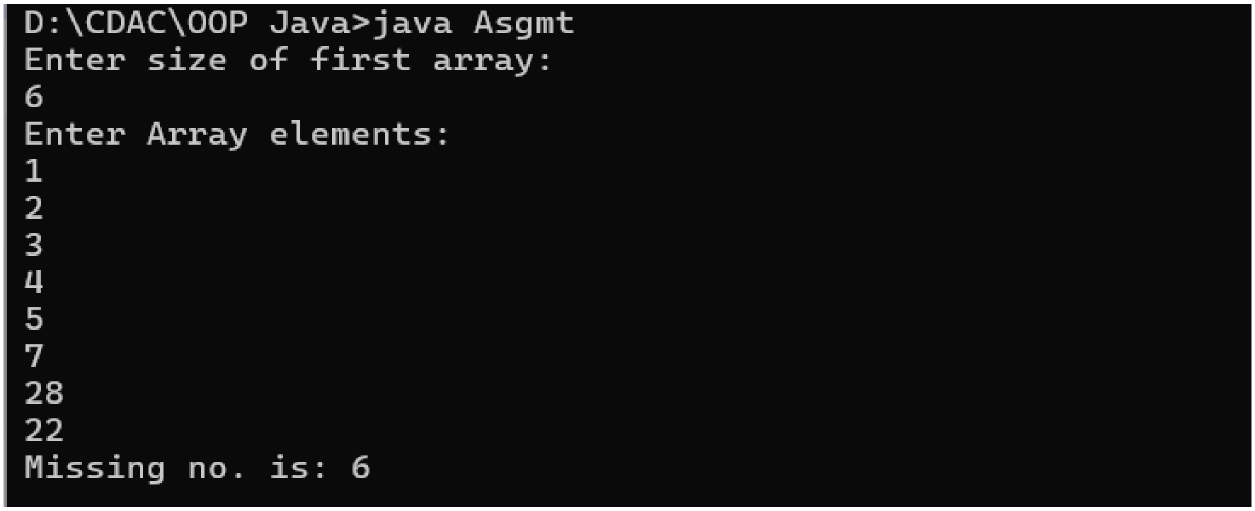
```
D:\CDAC\OOP Java>java Asgmt
Enter size of first array:
5
Enter Array elements:
5
6
7
8
9
Enter size of second array:
4
Enter Array elements:
1
2
3
4
Sorted Array after Merging:
1 2 3 4 5 6 7 8 9
```

9. Find Missing Number in an Array

○ Given an array of size n-1 containing numbers from 1 to n, find the missing number.

```
import java.util.*;
class Asgmt{
	public static void main(String args[]){
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter size of first array:
");
		int size=sc.nextInt();
		int n=size+1;
```

```
        int arr[]=new int[size];
        System.out.println("Enter Array elements: ");

            for(int i=0;i<size;i++){
                arr[i]=sc.nextInt();
            }
            int ideal_sum=(n*(n+1))/2;
            int sum=0;
            for(int i=0;i<size;i++){
                sum=sum+arr[i];
            }
            System.out.println(ideal_sum);
            System.out.println(sum);
            System.out.println("Missing no. is:
"+(ideal_sum-sum));
    }
}
```



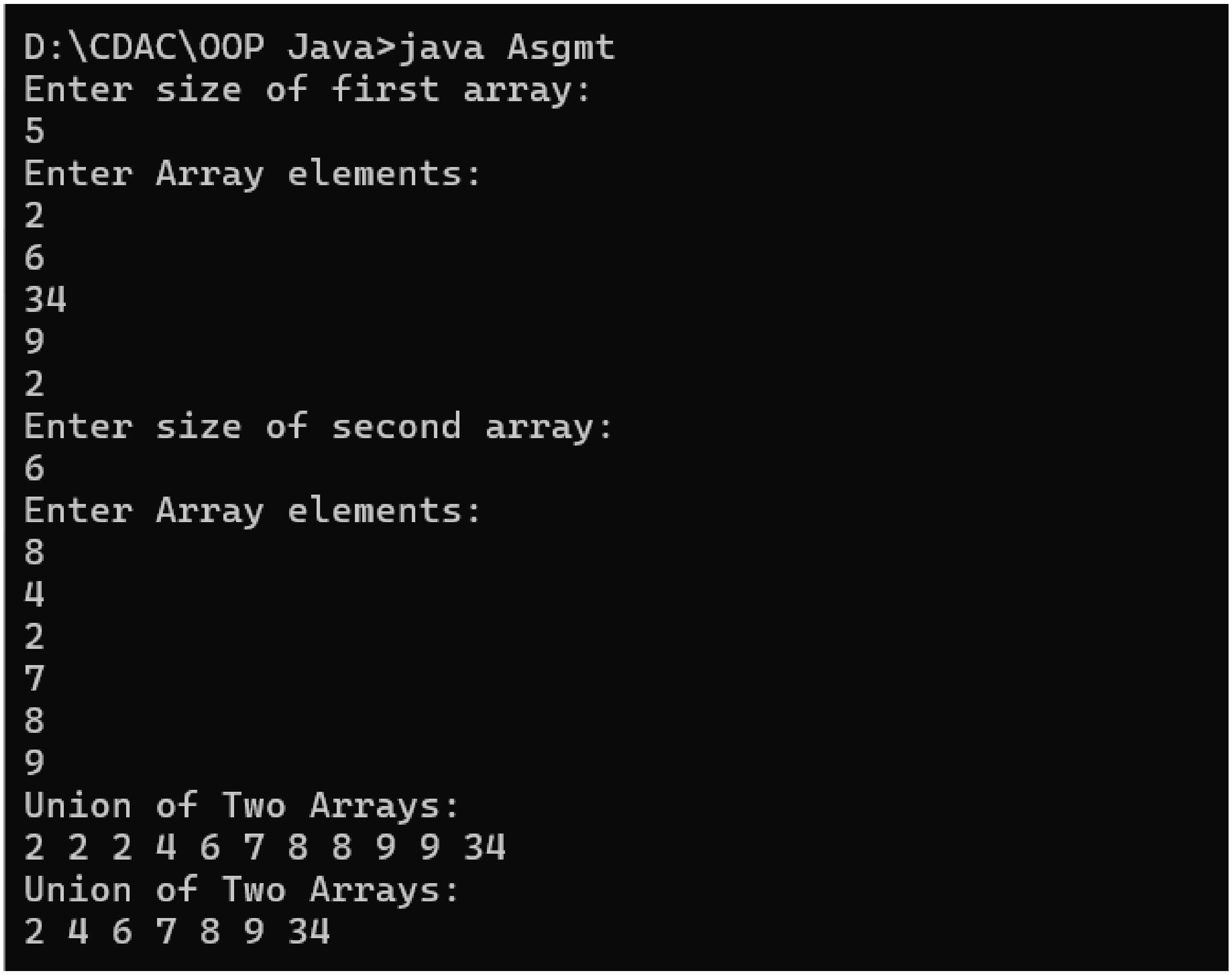
```
D:\CDAC\OOP Java>java Asgmt
Enter size of first array:
6
Enter Array elements:
1
2
3
4
5
7
28
22
Missing no. is: 6
```

10. Find Intersection and Union of Two Arrays

○ Find the intersection and union of two unsorted arrays.

```
import java.util.*;
class Asgmt{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter size of first array:
");
        int size1=sc.nextInt();
        int arr1[]=new int[size1];
        System.out.println("Enter Array elements: ");

            for(int i=0;i<size1;i++){
                arr1[i]=sc.nextInt();
            }
            System.out.println("Enter size of second
array: ");
            int size2=sc.nextInt();
            int arr2[]=new int[size2];
            System.out.println("Enter Array
elements: ");
```

```
            for(int i=0;i<size2;i++){
                arr2[i]=sc.nextInt();
            }
            System.out.println("Union of Two Arrays:
");

            int res_size=size1+size2;
            int res[]=new int[res_size];
            for(int i=0;i<size1;i++){
                res[i]=arr1[i];
            }
            for(int
i=size1,j=0;i<(size1+size2);i++,j++){
                res[i]=arr2[j];
            }
            java.util.Arrays.sort(res);
            for(int i=0;i<res_size;i++){
                System.out.print(res[i]+" ");
            }
            System.out.println();
        System.out.println("Union of Two Arrays: ");
            for(int i=0;i<res.length-1;i++){
                if(res[i]!=res[i+1]){
                    System.out.print(res[i]+" ");
                    }
                }
                System.out.print(res[(res.length)-1]);
                System.out.println();
        }
}
```



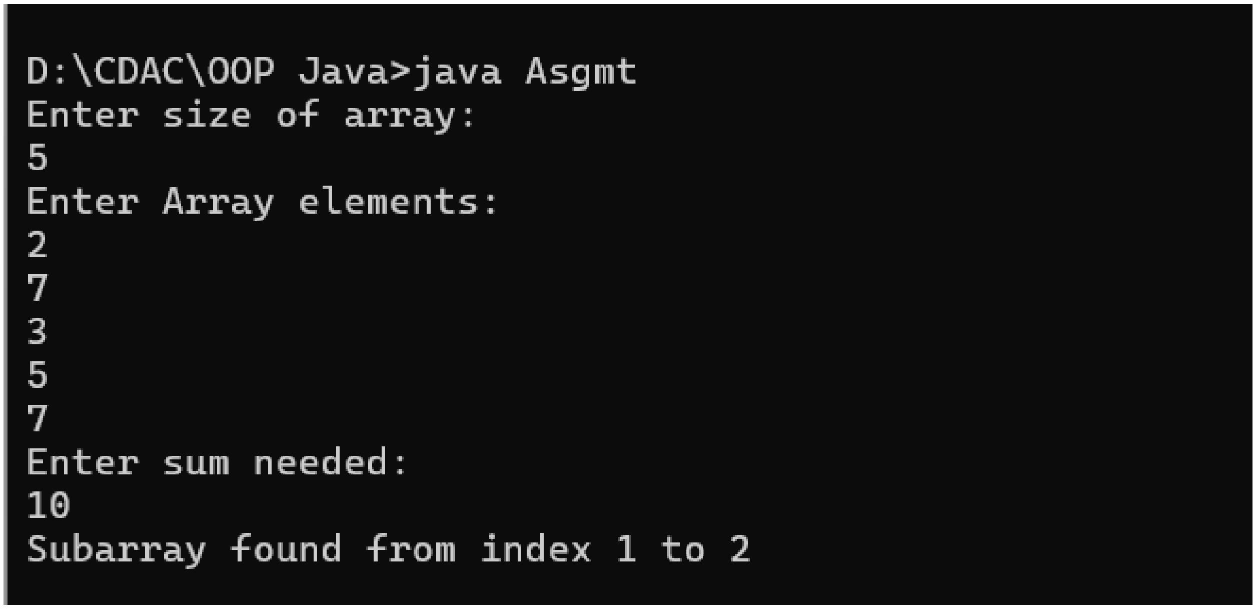
```
D:\CDAC\OOP Java>java Asgmt
Enter size of first array:
5
Enter Array elements:
2
6
34
9
2
Enter size of second array:
6
Enter Array elements:
8
4
2
7
8
9
Union of Two Arrays:
2 2 2 4 6 7 8 8 9 9 34
Union of Two Arrays:
2 4 6 7 8 9 34
```

11. Find a Subarray with Given Sum

○ Given an array of integers, find the subarray that sums to a given value S.

```
import java.util.*;
class Asgmt{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter size of array: ");
        int size=sc.nextInt();
        int arr[]=new int[size];
        System.out.println("Enter Array elements: ");

            for(int i=0;i<size;i++){
                arr[i]=sc.nextInt();
            }
            System.out.println("Enter sum needed:
");
        int s=sc.nextInt();
        for(int i=0;i<arr.length;i++){
            int sum=0;
            for(int j=i;j<arr.length;j++){
                sum=sum+arr[j];
                if(sum==s){
```

```
                              System.out.println("Subarray
found from index " + i + " to " + j );
                              return;
                        }
                  }
            }
System.out.println("no such substring found: ");
      }}
```

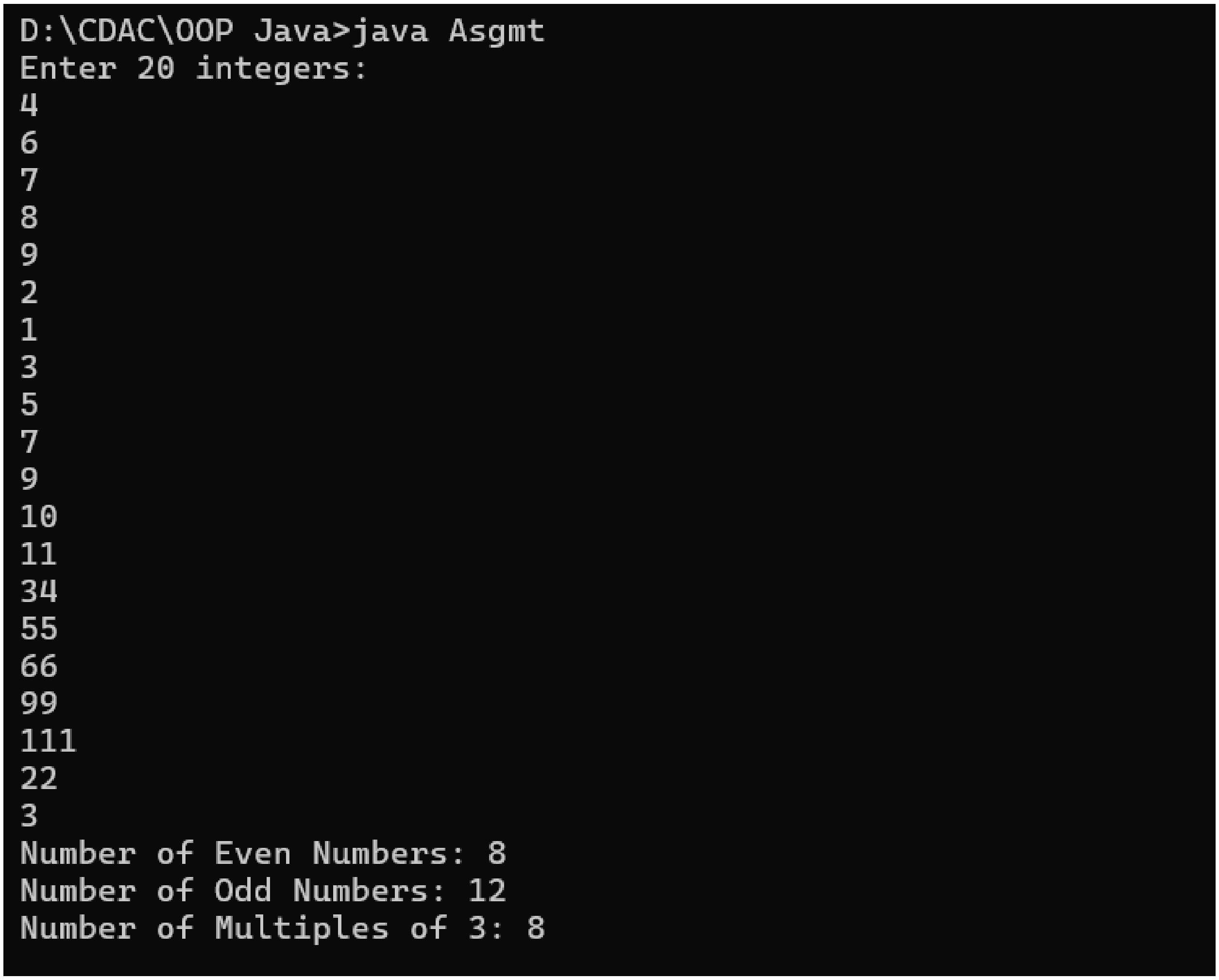


```
D:\CDAC\OOP Java>java Asgmt
Enter size of array:
5
Enter Array elements:
2
7
3
5
7
Enter sum needed:
10
Subarray found from index 1 to 2
```

12. Write a program to accept 20 integer numbers in a single Dimensional Array. Find and Display the following:

○ Number of even numbers.
○ Number of odd numbers.
○ Number of multiples of 3

```
import java.util.*;
class Asgmt{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int[] arr = new int[20];
        int evenCount = 0, oddCount = 0,
multipleOfThreeCount = 0;

        System.out.println("Enter 20 integers: ");
        for (int i = 0; i < 20; i++) {
            arr[i] = sc.nextInt();

            if (arr[i] % 2 == 0) {
                evenCount++;
            } else {
                oddCount++;
            }

            if (arr[i] % 3 == 0) {
                multipleOfThreeCount++;
            }
        }
        System.out.println("Number of Even Numbers: "
+ evenCount);
        System.out.println("Number of Odd Numbers: "
+ oddCount);
        System.out.println("Number of Multiples of 3: "
+ multipleOfThreeCount);
    }
}
```



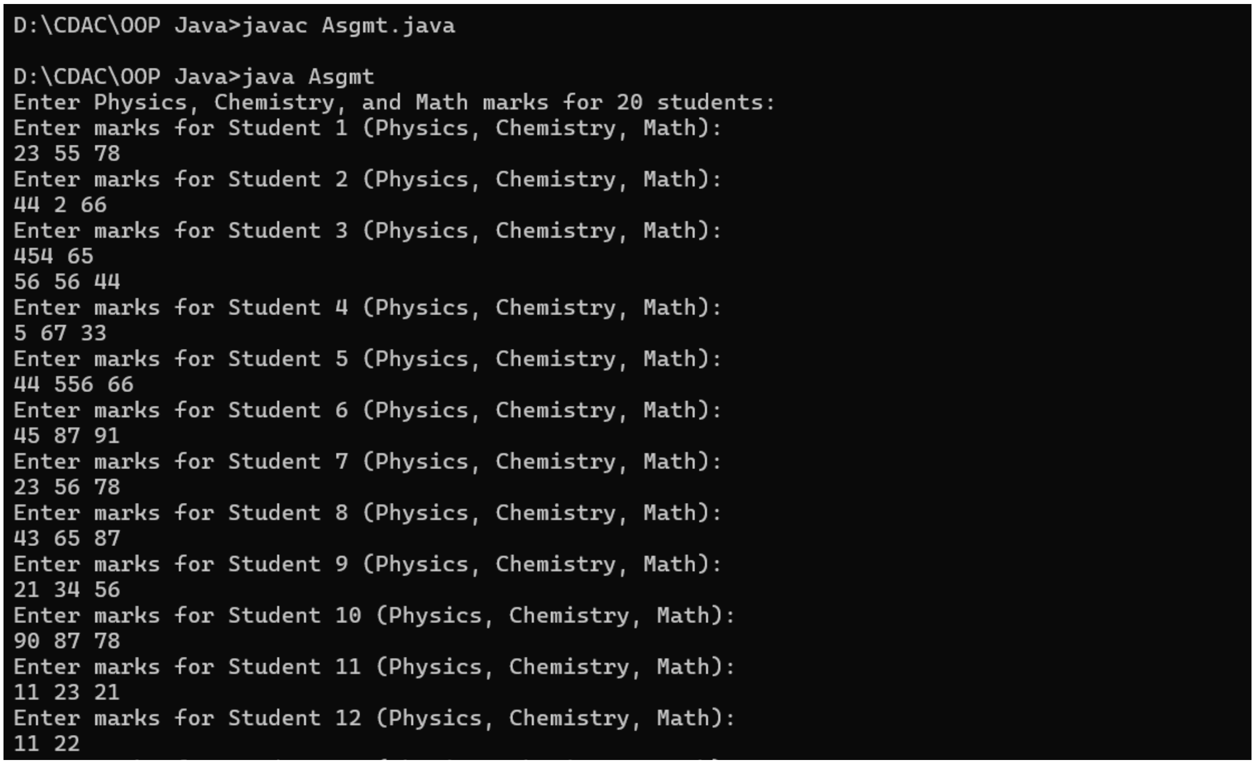
```
D:\CDAC\OOP Java>java Asgmt
Enter 20 integers:
4
6
7
8
9
2
1
3
5
7
9
10
11
34
55
66
99
111
22
3
Number of Even Numbers: 8
Number of Odd Numbers: 12
Number of Multiples of 3: 8
```

13. Write a program to accept the marks in Physics, Chemistry and Maths secured by 20 class students in a single Dimensional Array. Find and display the following:

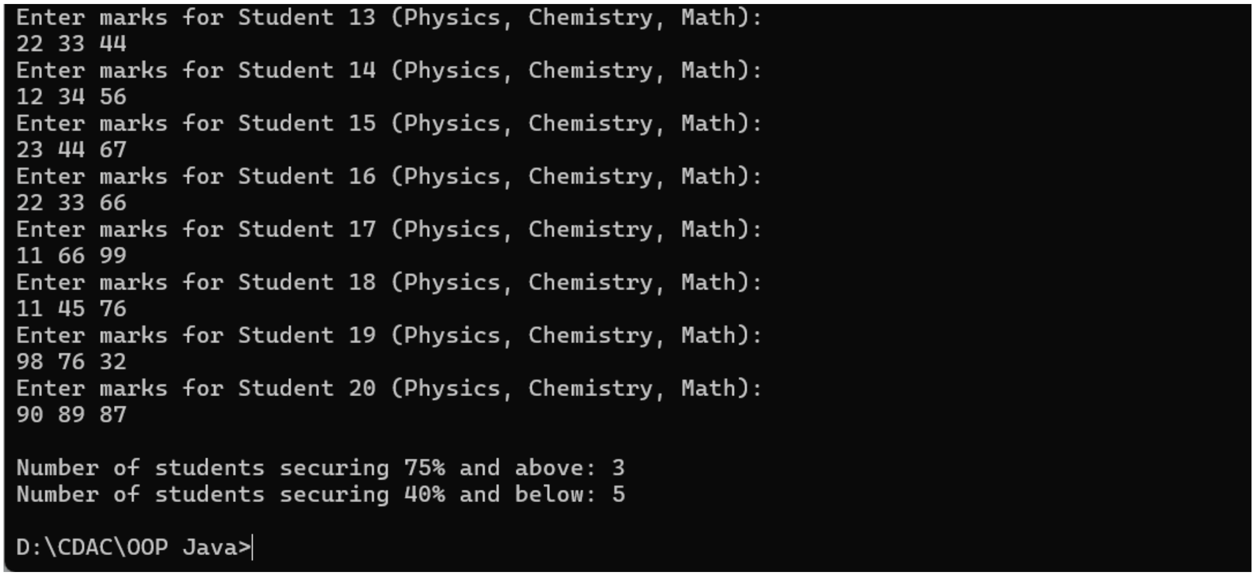
○ Number of students securing 75% and above in aggregate.

○ Number of students securing 40% and below in aggregate.

```
import java.util.*;
class Asgmt{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int totalStudents = 20;
        int[] marks = new int[totalStudents * 3];
        int highScorers = 0, lowScorers = 0;

        System.out.println("Enter Physics, Chemistry,
and Math marks for " + totalStudents + " students:");
        for (int i = 0; i < totalStudents; i++) {
            System.out.println("Enter marks for Student "
+ (i + 1) + " (Physics, Chemistry, Math):");
            int physics = sc.nextInt();
            int chemistry = sc.nextInt();
            int math = sc.nextInt();

            marks[i * 3] = physics;
            marks[i * 3 + 1] = chemistry;
            marks[i * 3 + 2] = math;

            int totalMarks = physics + chemistry + math;
            double percentage = (totalMarks / 3.0);
            if (percentage >= 75) {
                highScorers++;
            } else if (percentage <= 40) {
                lowScorers++;
            }
        }
        System.out.println("\nNumber of students
securing 75% and above: " + highScorers);
        System.out.println("Number of students
securing 40% and below: " + lowScorers);
    }
}
```



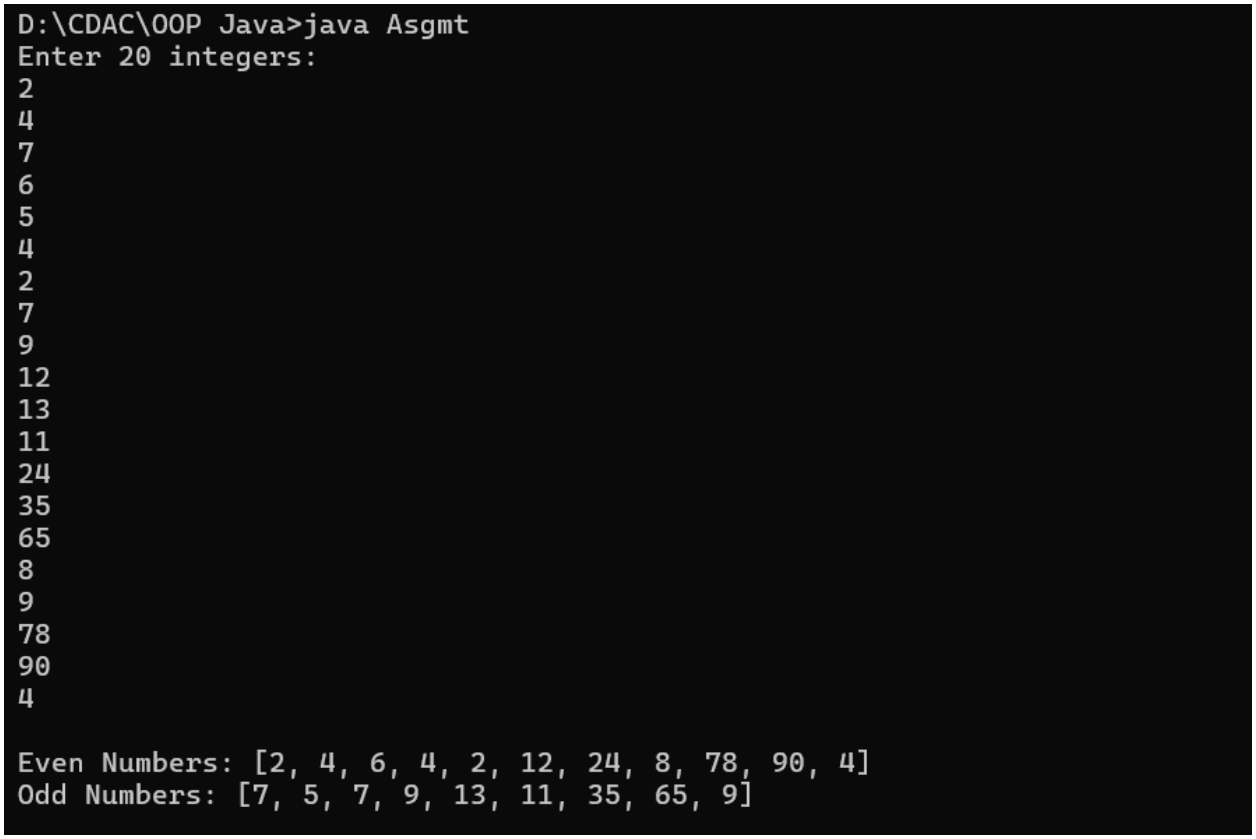
```
D:\CDAC\OOP Java>javac Asgmt.java

D:\CDAC\OOP Java>java Asgmt
Enter Physics, Chemistry, and Math marks for 20 students:
Enter marks for Student 1 (Physics, Chemistry, Math):
23 55 78
Enter marks for Student 2 (Physics, Chemistry, Math):
44 2 66
Enter marks for Student 3 (Physics, Chemistry, Math):
454 65
56 56 44
Enter marks for Student 4 (Physics, Chemistry, Math):
5 67 33
Enter marks for Student 5 (Physics, Chemistry, Math):
44 556 66
Enter marks for Student 6 (Physics, Chemistry, Math):
45 87 91
Enter marks for Student 7 (Physics, Chemistry, Math):
23 56 78
Enter marks for Student 8 (Physics, Chemistry, Math):
43 65 87
Enter marks for Student 9 (Physics, Chemistry, Math):
21 34 56
Enter marks for Student 10 (Physics, Chemistry, Math):
90 87 78
Enter marks for Student 11 (Physics, Chemistry, Math):
11 23 21
Enter marks for Student 12 (Physics, Chemistry, Math):
11 22
```



```
Enter marks for Student 13 (Physics, Chemistry, Math):
22 33 44
Enter marks for Student 14 (Physics, Chemistry, Math):
12 34 56
Enter marks for Student 15 (Physics, Chemistry, Math):
23 44 67
Enter marks for Student 16 (Physics, Chemistry, Math):
22 33 66
Enter marks for Student 17 (Physics, Chemistry, Math):
11 66 99
Enter marks for Student 18 (Physics, Chemistry, Math):
11 45 76
Enter marks for Student 19 (Physics, Chemistry, Math):
98 76 32
Enter marks for Student 20 (Physics, Chemistry, Math):
90 89 87

Number of students securing 75% and above: 3
Number of students securing 40% and below: 5

D:\CDAC\OOP Java>
```

14. Write a program in Java to accept 20 numbers in a single dimensional array arr[20]. Transfer and store all the even numbers in an array even[ ] and all the odd numbers in another array odd[ ]. Finally, print the elements of the even & the odd array.

```
import java.util.*;
class Asgmt{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int[] arr = new int[20];

        ArrayList<Integer> even = new ArrayList<>();
        ArrayList<Integer> odd = new ArrayList<>();

        System.out.println("Enter 20 integers: ");
        for (int i = 0; i < 20; i++) {
            arr[i] = sc.nextInt();

            if (arr[i] % 2 == 0) {
                even.add(arr[i]);
            } else {
                odd.add(arr[i]);
            }
        }

        System.out.println("\nEven Numbers: " + even);
        System.out.println("Odd Numbers: " + odd);
    }
}
```



```
D:\CDAC\OOP Java>java Asgmt
Enter 20 integers:
2
4
7
6
5
4
2
7
9
12
13
11
24
35
65
8
9
78
90
4

Even Numbers: [2, 4, 6, 4, 2, 12, 24, 8, 78, 90, 4]
Odd Numbers: [7, 5, 7, 9, 13, 11, 35, 65, 9]
```

15. Write a Java program to print all sub-arrays with 0 sum present in a given array of integers.

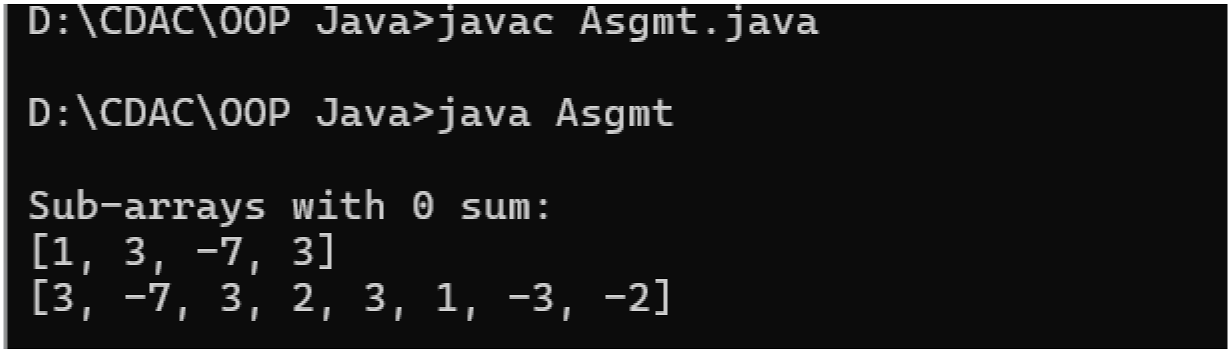
Example:
Input :

nums1 = { 1, 3, -7, 3, 2, 3, 1, -3, -2, -2 }
nums2 = { 1, 2, -3, 4, 5, 6 }
nums3= { 1, 2, -2, 3, 4, 5, 6 }
Output:

```
Sub-arrays with 0 sum : [1, 3, -7, 3]
Sub-arrays with 0 sum : [3, -7, 3, 2, 3, 1, -3, -2]

Sub-arrays with 0 sum : [1, 2, -3]
Sub-arrays with 0 sum : [2, -2]
```

```
import java.util.*;
class Asgmt{
     public static void main(String args[]){
          Scanner sc=new Scanner(System.in);
           int[] nums = {1, 3, -7, 3, 2, 3, 1, -3, -2, -2};
           int n = nums.length;
      boolean found = false;

      System.out.println("\nSub-arrays with 0 sum:");
      for (int i = 0; i < n; i++) {
         int sum = 0;
         for (int j = i; j < n; j++) {
            sum += nums[j];
```

```
                if (sum == 0) {
System.out.println(Arrays.toString(Arrays.copyOfRange(nums, i, j + 1)));
                    found = true;
                }
            }
        }
        if (!found) {
            System.out.println("No subarray with sum 0 found.");
        }
      }}
```



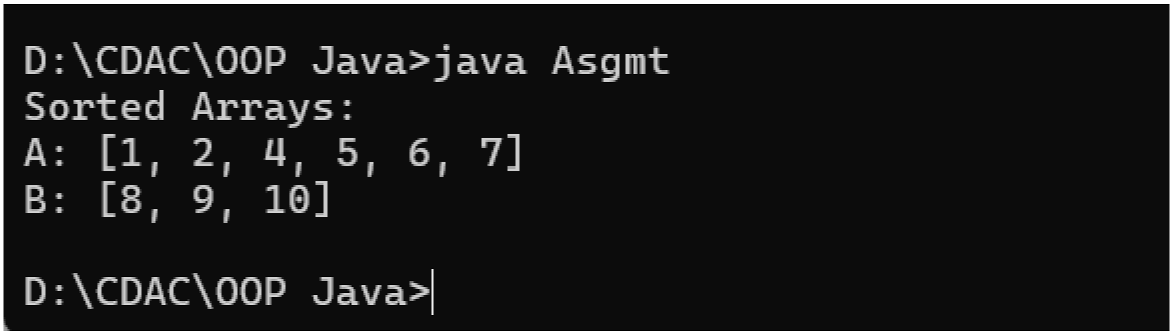
```
D:\CDAC\OOP Java>javac Asgmt.java

D:\CDAC\OOP Java>java Asgmt

Sub-arrays with 0 sum:
[1, 3, -7, 3]
[3, -7, 3, 2, 3, 1, -3, -2]
```

16. Given two sorted arrays A and B of size p and q, write a Java program to merge elements of A with B by maintaining the sorted order i.e. fill A with first p smallest elements and fill B with remaining elements.

```
 Example:
 Input :
 int[] A = { 1, 5, 6, 7, 8, 10 }
int[] B = { 2, 4, 9 }
Output:
Sorted Arrays:
A: [1, 2, 4, 5, 6, 7]
B: [8, 9, 10]
```

```
import java.util.Arrays;

class Asgmt {
   public static void mergeSortedArrays(int[] A, int[]
B) {
      int p = A.length, q = B.length;
      int[] merged = new int[p + q];

      System.arraycopy(A, 0, merged, 0, p);
```

```
        System.arraycopy(B, 0, merged, p, q);

        Arrays.sort(merged);

        System.arraycopy(merged, 0, A, 0, p);
        System.arraycopy(merged, p, B, 0, q);

        System.out.println("Sorted Arrays:");
        System.out.println("A: " + Arrays.toString(A));
        System.out.println("B: " + Arrays.toString(B));
    }
    public static void main(String[] args) {
        int[] A = {1, 5, 6, 7, 8, 10};
        int[] B = {2, 4, 9};

        mergeSortedArrays(A, B);
    }
}
```

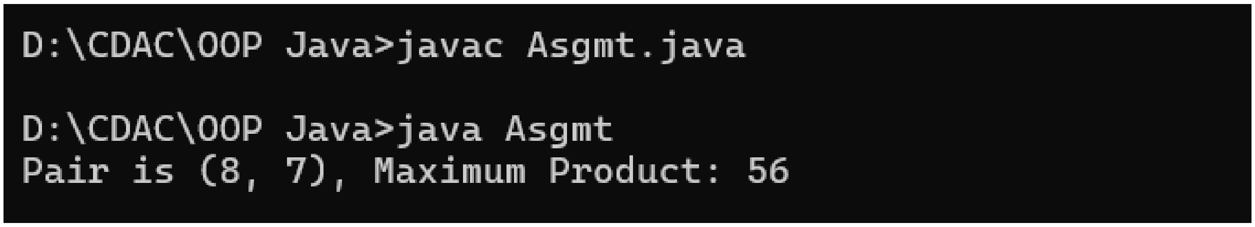


```
D:\CDAC\OOP Java>java Asgmt
Sorted Arrays:
A: [1, 2, 4, 5, 6, 7]
B: [8, 9, 10]

D:\CDAC\OOP Java>
```

17. Write a Java program to find the maximum product of two integers in a given array of integers.

Example:
Input : nums = { 2, 3, 5, 7, -7, 5, 8, -5 }
Output: Pair is (7, 8), Maximum Product: 56

```
import java.util.Arrays;

class Asgmt {
    public static void main(String[] args) {
        int[] nums = {2, 3, 5, 7, -7, 5, 8, -5};

        int max1 = Integer.MIN_VALUE, max2 =
Integer.MIN_VALUE;
        int min1 = Integer.MAX_VALUE, min2 =
Integer.MAX_VALUE;

        for (int num : nums) {
            if (num > max1) {
                max2 = max1;
                max1 = num;
            } else if (num > max2) {
                max2 = num;
            }

            if (num < min1) {
                min2 = min1;
                min1 = num;
            } else if (num < min2) {
                min2 = num;
            }
        }
        int product1 = max1 * max2;
        int product2 = min1 * min2;

        if (product1 > product2) {
            System.out.println("Pair is (" + max1 + ", " +
max2 + "), Maximum Product: " + product1);
        } else {
            System.out.println("Pair is (" + min1 + ", " +
min2 + "), Maximum Product: " + product2);
        }
    }
}
```

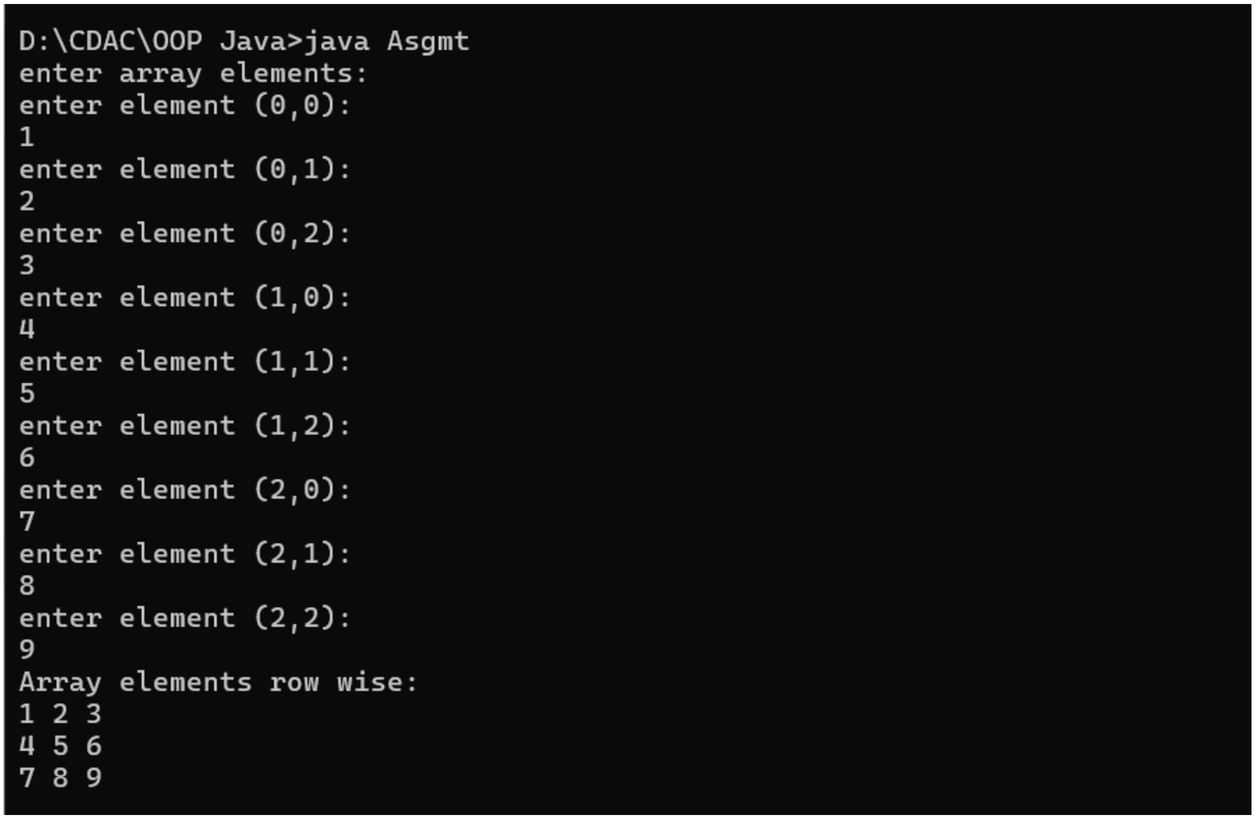


```
D:\CDAC\OOP Java>javac Asgmt.java

D:\CDAC\OOP Java>java Asgmt
Pair is (8, 7), Maximum Product: 56
```

18. Print a Matrix

○ Given an m x n matrix, print all its elements row-wise.

```
import java.util.Arrays;
import java.util.Scanner;

class Asgmt {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
            int arr[][]=new int[3][3];
            System.out.println("enter array elements: ");
            for(int i=0;i<3;i++){
                for(int j=0;j<3;j++){
                    System.out.println("enter element
("+i+","+j +"):");
                    arr[i][j]=sc.nextInt();
                }
            }
            System.out.println("Array elements row
wise: ");
            for(int i=0;i<3;i++){
                for(int j=0;j<3;j++){
                    System.out.print(arr[i][j]+" ");
                }
                System.out.println();
            }
        }
    }
```



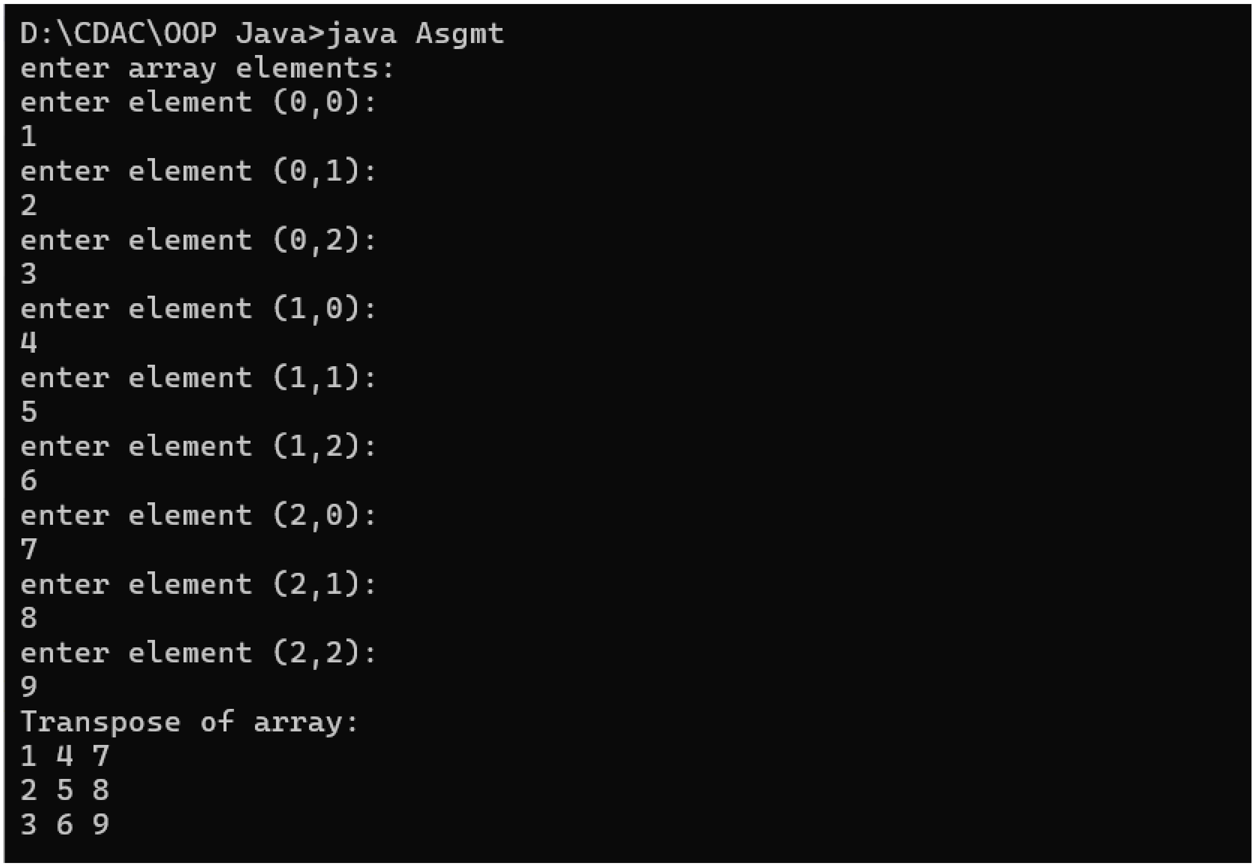
```
D:\CDAC\OOP Java>java Asgmt
enter array elements:
enter element (0,0):
1
enter element (0,1):
2
enter element (0,2):
3
enter element (1,0):
4
enter element (1,1):
5
enter element (1,2):
6
enter element (2,0):
7
enter element (2,1):
8
enter element (2,2):
9
Array elements row wise:
1 2 3
4 5 6
7 8 9
```

19. Transpose of a Matrix

○ Given a matrix, return its transpose (swap rows and columns).

```
import java.util.*;

class Asgmt {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
             int arr[][]=new int[3][3];
             int transpose[][]=new int[3][3];
             System.out.println("enter array elements: ");
             for(int i=0;i<3;i++){
                    for(int j=0;j<3;j++){
                           System.out.println("enter element
("+i+","+j +"):");
                           arr[i][j]=sc.nextInt();
                    }
             }
             System.out.print("Transpose of array: ");
             System.out.println();
             int temp;
             for(int i=0;i<3;i++){
                    for(int j=0;j<3;j++){
```

```
                    transpose[j][i]=arr[i][j];
            }
        }
        for(int i=0;i<3;i++){
            for(int j=0;j<3;j++){
                System.out.print(transpose[i][j]+" ");
            }
            System.out.println();
        }
    }
}
```



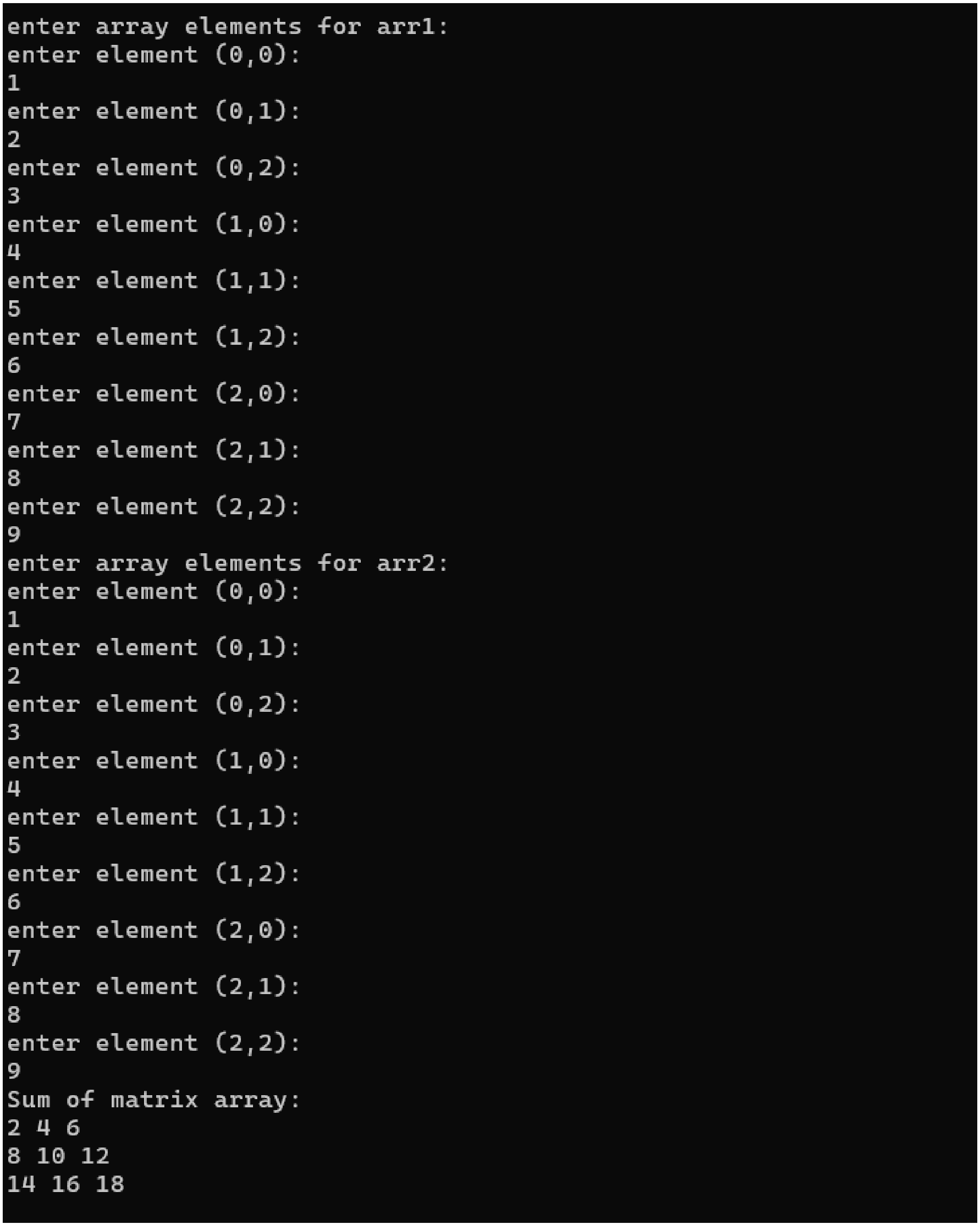
```
D:\CDAC\OOP Java>java Asgmt
enter array elements:
enter element (0,0):
1
enter element (0,1):
2
enter element (0,2):
3
enter element (1,0):
4
enter element (1,1):
5
enter element (1,2):
6
enter element (2,0):
7
enter element (2,1):
8
enter element (2,2):
9
Transpose of array:
1 4 7
2 5 8
3 6 9
```

20. Sum of Two Matrices

○ Given two matrices of the same size, compute their sum.

```
import java.util.Arrays;
import java.util.Scanner;

class Asgmt {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
             int arr1[][]=new int[3][3];
             int arr2[][]=new int[3][3];
             System.out.println("enter array elements for
arr1: ");
             for(int i=0;i<3;i++){
                    for(int j=0;j<3;j++){
                           System.out.println("enter element
("+i+","+j +"):");
                           arr1[i][j]=sc.nextInt();
                    }
             }
             System.out.println("enter array elements for
arr2: ");
             for(int i=0;i<3;i++){
```

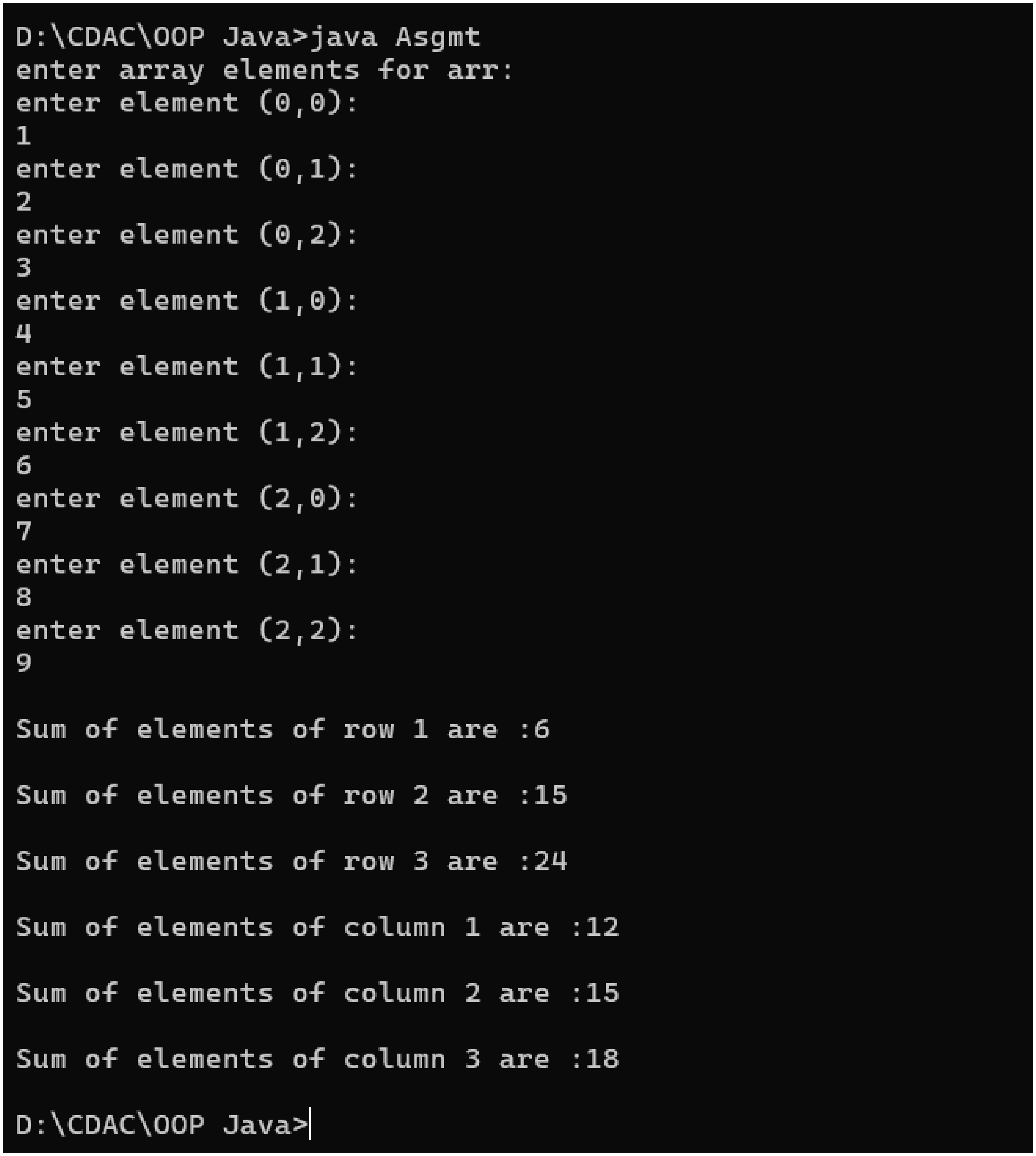
```
            for(int j=0;j<3;j++){
                System.out.println("enter element
("+i+","+j +"):");
                arr2[i][j]=sc.nextInt();
            }
        }

        System.out.print("Sum of matrix array: \n");
        for(int i=0;i<3;i++){
            for(int j=0;j<3;j++){

System.out.print(arr1[i][j]+arr2[i][j]+" ");
            }
            System.out.println();
        }
  }
}
```



```
enter array elements for arr1:
enter element (0,0):
1
enter element (0,1):
2
enter element (0,2):
3
enter element (1,0):
4
enter element (1,1):
5
enter element (1,2):
6
enter element (2,0):
7
enter element (2,1):
8
enter element (2,2):
9
enter array elements for arr2:
enter element (0,0):
1
enter element (0,1):
2
enter element (0,2):
3
enter element (1,0):
4
enter element (1,1):
5
enter element (1,2):
6
enter element (2,0):
7
enter element (2,1):
8
enter element (2,2):
9
Sum of matrix array:
2 4 6
8 10 12
14 16 18
```

21. Row-wise and Column-wise Sum
○ Find the sum of each row and each column of a given matrix.

```
import java.util.Arrays;
import java.util.Scanner;

class Asgmt {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
            int arr1[][]=new int[3][3];
            //int sum=0;
            System.out.println("enter array elements for
arr: ");
            for(int i=0;i<3;i++){
                for(int j=0;j<3;j++){
                    System.out.println("enter element
("+i+","+j +"):");
                    arr1[i][j]=sc.nextInt();
                }
            }
            for(int i=0;i<3;i++){
                int sum=0;
                for(int j=0;j<3;j++){
                        sum=sum+(arr1[i][j]);
                }
                System.out.println("\nSum of elements
of row "+(i+1)+" are :"+sum);
            }
            for(int j=0;j<3;j++){
                int sum=0;
                for(int i=0;i<3;i++){
                        sum=sum+(arr1[i][j]);
                }
                System.out.println("\nSum of elements
of column "+(j+1)+" are :"+sum);
            }

    }
}
```



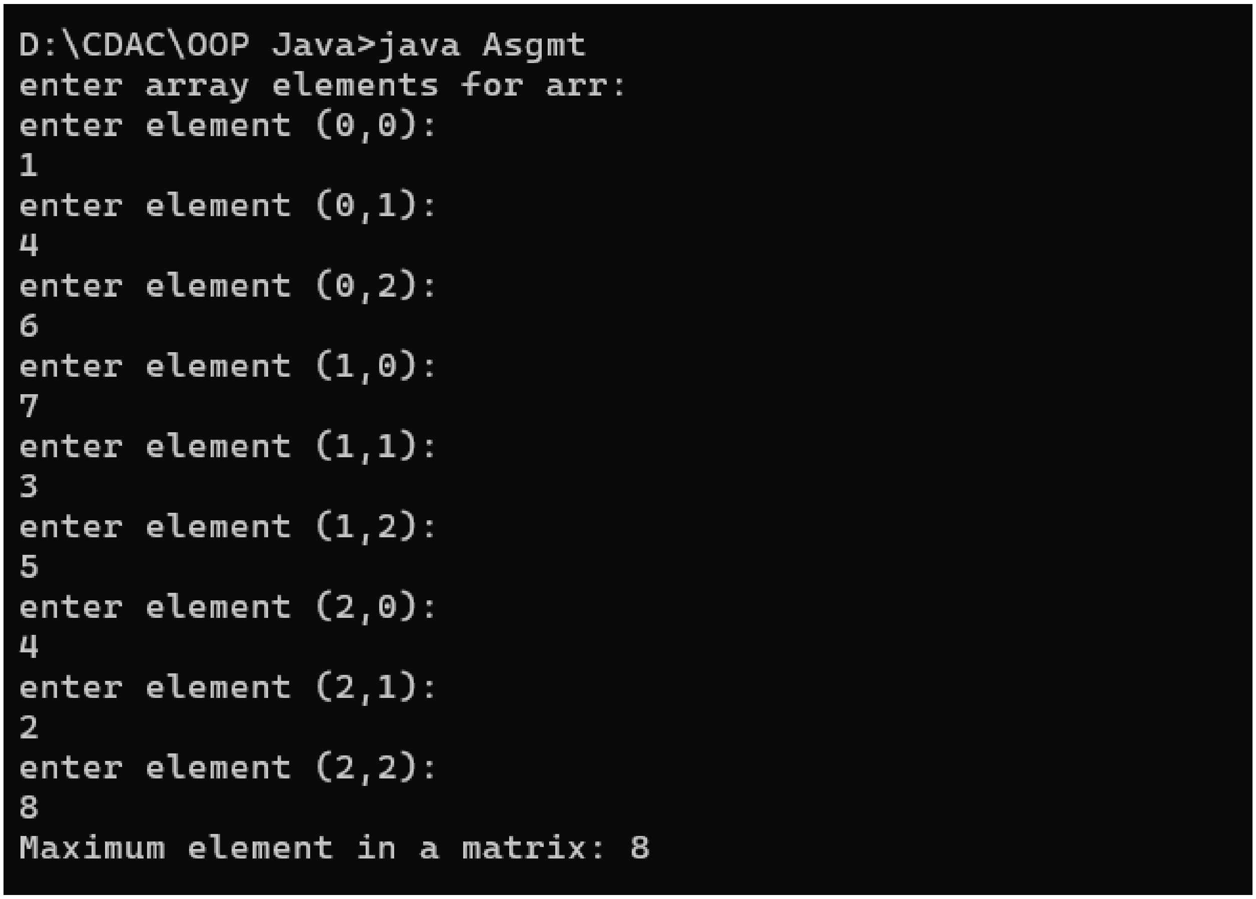
```
D:\CDAC\OOP Java>java Asgmt
enter array elements for arr:
enter element (0,0):
1
enter element (0,1):
2
enter element (0,2):
3
enter element (1,0):
4
enter element (1,1):
5
enter element (1,2):
6
enter element (2,0):
7
enter element (2,1):
8
enter element (2,2):
9

Sum of elements of row 1 are :6

Sum of elements of row 2 are :15

Sum of elements of row 3 are :24

Sum of elements of column 1 are :12

Sum of elements of column 2 are :15

Sum of elements of column 3 are :18

D:\CDAC\OOP Java>
```

22. Find the Maximum Element in a Matrix

○ Find the largest element in a given matrix.

```
import java.util.Arrays;
import java.util.Scanner;

class Asgmt {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
            int arr1[][]=new int[3][3];
            int max=0;
            System.out.println("enter array elements for
arr: ");
            for(int i=0;i<3;i++){
                  for(int j=0;j<3;j++){
                        System.out.println("enter element
("+i+","+j +"):");
                        arr1[i][j]=sc.nextInt();
                  }
            }
            for(int i=0;i<3;i++){
                  for(int j=0;j<3;j++){
                              if(arr1[i][j]>0){
                                    max=arr1[i][j];
```

```
                    }
                }
            }
            System.out.println("Maximum element in a matrix: "+max);
    }
}
```



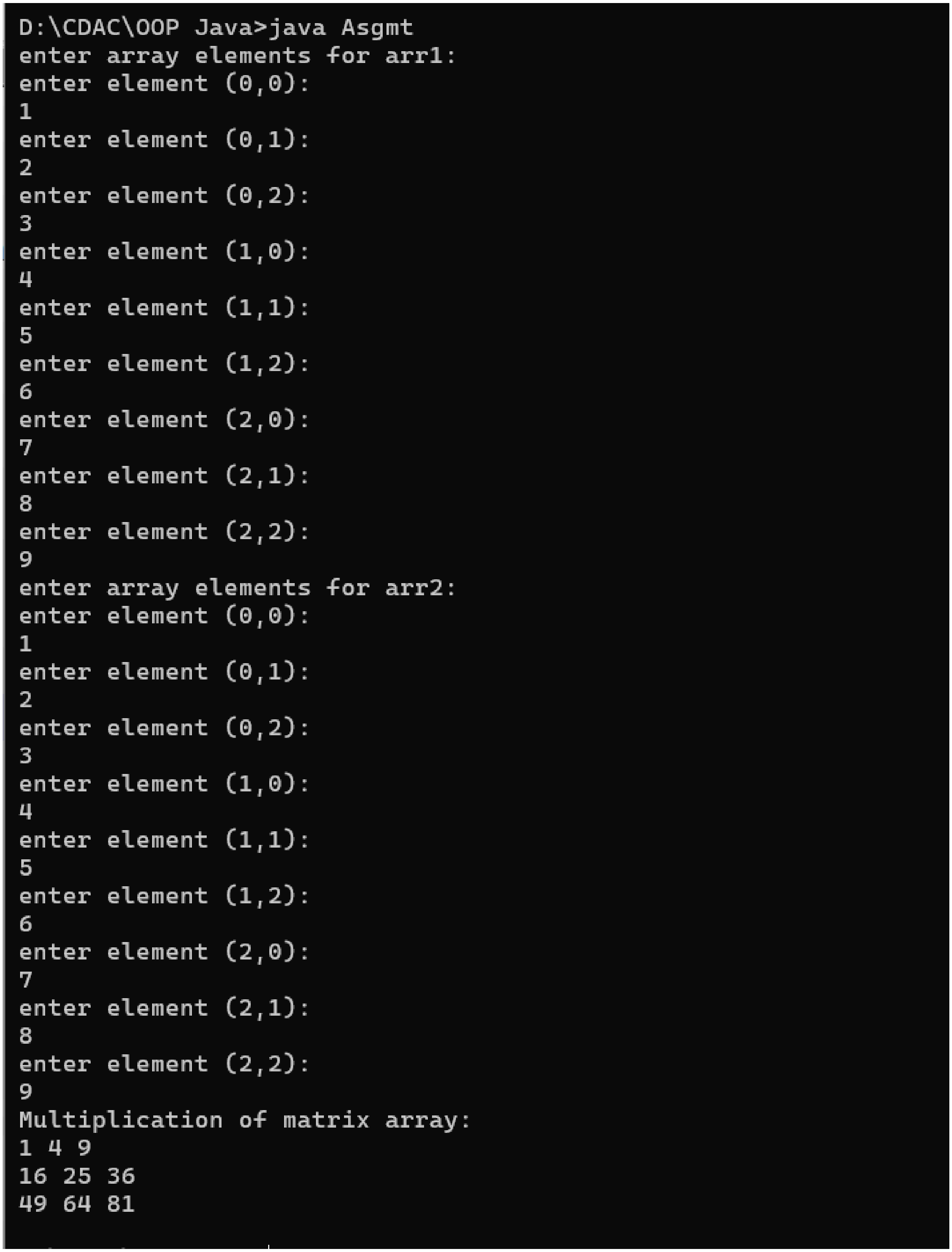
```
D:\CDAC\OOP Java>java Asgmt
enter array elements for arr:
enter element (0,0):
1
enter element (0,1):
4
enter element (0,2):
6
enter element (1,0):
7
enter element (1,1):
3
enter element (1,2):
5
enter element (2,0):
4
enter element (2,1):
2
enter element (2,2):
8
Maximum element in a matrix: 8
```

23. Matrix Multiplication

○ Multiply two matrices and return the resultant matrix.

```
import java.util.Arrays;
import java.util.Scanner;

class Asgmt {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
             int arr1[][]=new int[3][3];
             int arr2[][]=new int[3][3];
             System.out.println("enter array elements for
arr1: ");
             for(int i=0;i<3;i++){
                    for(int j=0;j<3;j++){
                          System.out.println("enter element
("+i+","+j +"):");
                          arr1[i][j]=sc.nextInt();
                    }
             }
             System.out.println("enter array elements for
arr2: ");
             for(int i=0;i<3;i++){
```

```
            for(int j=0;j<3;j++){
                System.out.println("enter element
("+i+","+j +"):");
                arr2[i][j]=sc.nextInt();
            }
        }
        System.out.print("Multiplication of matrix
array: \n");
        for(int i=0;i<3;i++){
            for(int j=0;j<3;j++){

System.out.print(arr1[i][j]*arr2[i][j]+" ");
            }
            System.out.println();
        }
  }
}
```



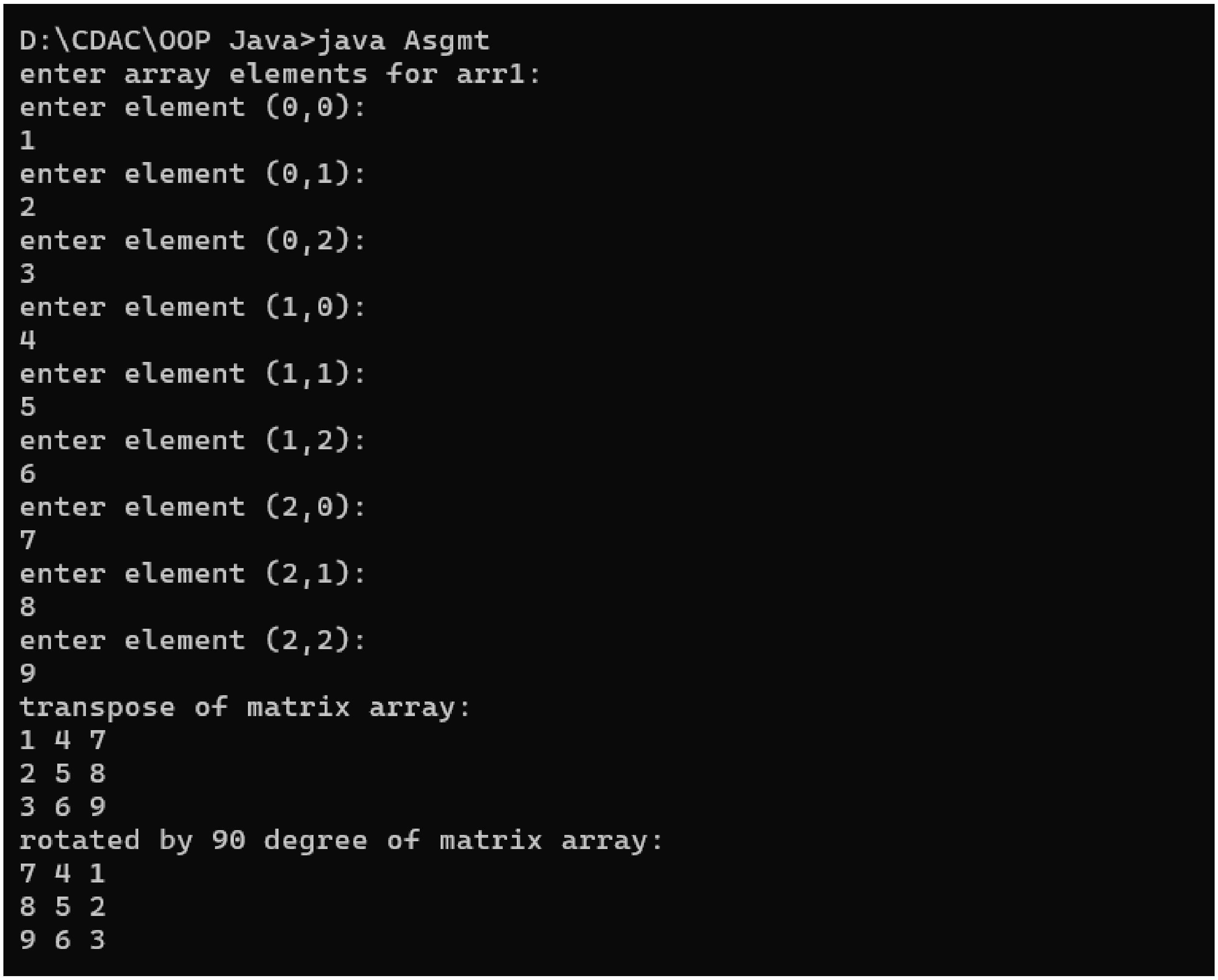
```
D:\CDAC\OOP Java>java Asgmt
enter array elements for arr1:
enter element (0,0):
1
enter element (0,1):
2
enter element (0,2):
3
enter element (1,0):
4
enter element (1,1):
5
enter element (1,2):
6
enter element (2,0):
7
enter element (2,1):
8
enter element (2,2):
9
enter array elements for arr2:
enter element (0,0):
1
enter element (0,1):
2
enter element (0,2):
3
enter element (1,0):
4
enter element (1,1):
5
enter element (1,2):
6
enter element (2,0):
7
enter element (2,1):
8
enter element (2,2):
9
Multiplication of matrix array:
1 4 9
16 25 36
49 64 81
```

24. Rotate a Matrix by 90 Degrees

○ Rotate a given N x N matrix by 90 degrees clockwise.

```
import java.util.Arrays;
import java.util.Scanner;

class Asgmt {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
            int arr1[][]=new int[3][3];
            int transpose[][]=new int[3][3];
            System.out.println("enter array elements for
arr1: ");
            for(int i=0;i<3;i++){
                for(int j=0;j<3;j++){
                    System.out.println("enter element
("+i+","+j +"):");
                    arr1[i][j]=sc.nextInt();
                }
            }

            System.out.print("transpose of matrix array:
\n");
```

```
        for(int i=0;i<3;i++){
            for(int j=0;j<3;j++){
                transpose[j][i]=arr1[i][j];
            }
        }
        for(int i=0;i<3;i++){
            for(int j=0;j<3;j++){
                System.out.print(transpose[i][j]+" ");
            }
            System.out.println();
        }
        System.out.print("rotated by 90 degree of
matrix array: \n");

        for(int i=0;i<3;i++){
            for(int j=2;j>=0;j--){
                System.out.print(transpose[i][j]+" ");
            }
            System.out.println();
        }


    }
}
```



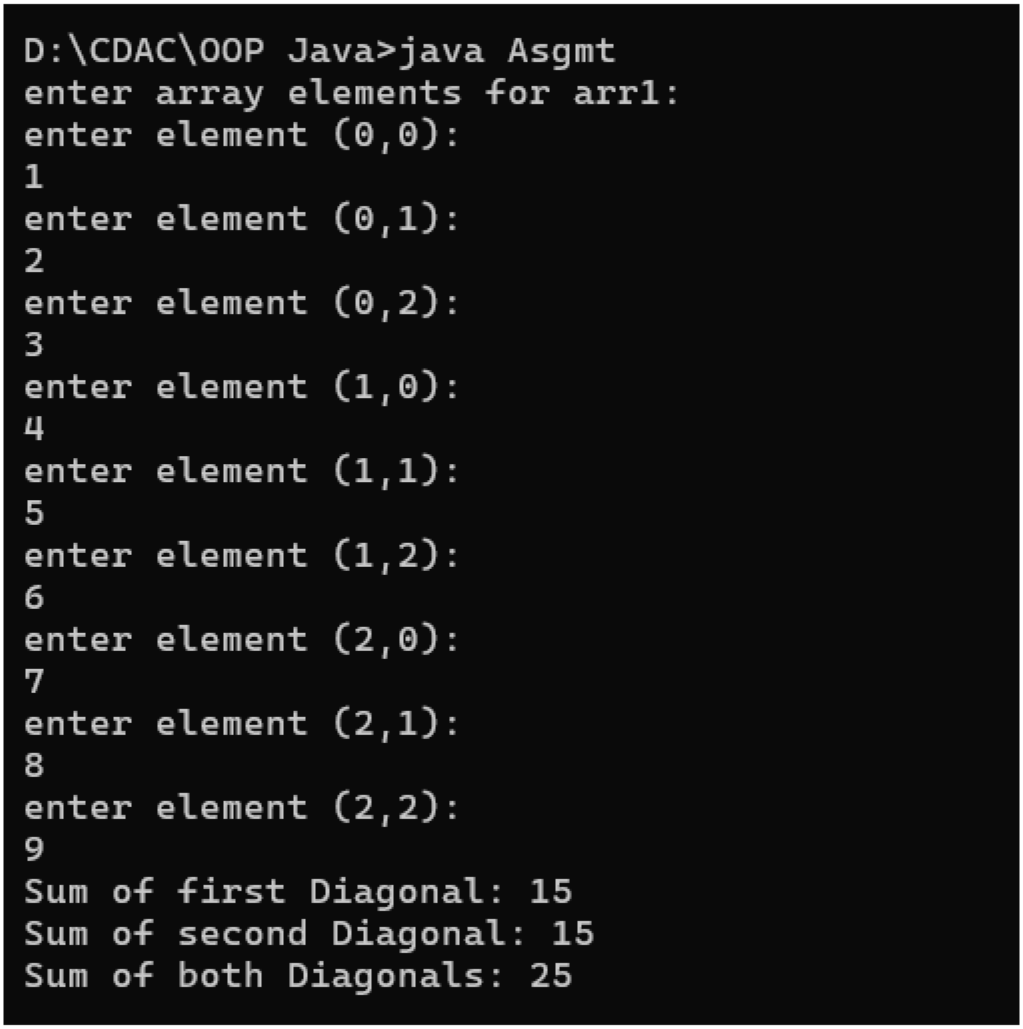
```
D:\CDAC\OOP Java>java Asgmt
enter array elements for arr1:
enter element (0,0):
1
enter element (0,1):
2
enter element (0,2):
3
enter element (1,0):
4
enter element (1,1):
5
enter element (1,2):
6
enter element (2,0):
7
enter element (2,1):
8
enter element (2,2):
9
transpose of matrix array:
1 4 7
2 5 8
3 6 9
rotated by 90 degree of matrix array:
7 4 1
8 5 2
9 6 3
```

25. Find the Diagonal Sum

○ Compute the sum of both diagonals in a square matrix.

```
import java.util.Arrays;
import java.util.Scanner;

class Asgmt {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
            int arr1[][]=new int[3][3];
            int primarySum=0,secSum=0;
            System.out.println("enter array elements for
arr1: ");
            for(int i=0;i<3;i++){
                  for(int j=0;j<3;j++){
                        System.out.println("enter element
("+i+","+j +"):");
                        arr1[i][j]=sc.nextInt();
                  }
            }
            for(int i=0;i<3;i++){
                  primarySum=primarySum+arr1[i][i];
                  secSum=secSum+(arr1[i][2-i]);
            }
            int total=primarySum+secSum;
            total=total-(arr1[3-2][3-2]);
            System.out.println("Sum of first Diagonal:
"+primarySum);
            System.out.println("Sum of second
Diagonal: "+secSum);
            System.out.println("Sum of both Diagonals:
"+total);
```

```
    }
}
```



```
D:\CDAC\OOP Java>java Asgmt
enter array elements for arr1:
enter element (0,0):
1
enter element (0,1):
2
enter element (0,2):
3
enter element (1,0):
4
enter element (1,1):
5
enter element (1,2):
6
enter element (2,0):
7
enter element (2,1):
8
enter element (2,2):
9
Sum of first Diagonal: 15
Sum of second Diagonal: 15
Sum of both Diagonals: 25
```