LAB SHEET -1

AIM 1: Understanding the concept of Array and its Applications (5 points)

An array is a collection of similar data elements. These data elements have the same data type. The elements of the array are stored in consecutive memory locations and are referenced by an index (also known as the subscript). The subscript is an ordinal number which is used to identify an element of the array.

Operations on Arrays
There are a number of operations that can be preformed on arrays. These operations include:

- Traversing an array Inserting an element in an array
- Searching an element in an array
- Deleting an element from an array
- Merging two arrays Sorting an array in ascending or descending order

1. **Implement a program for inserting a new element to the specified position of an array.**

InsertingArray.java

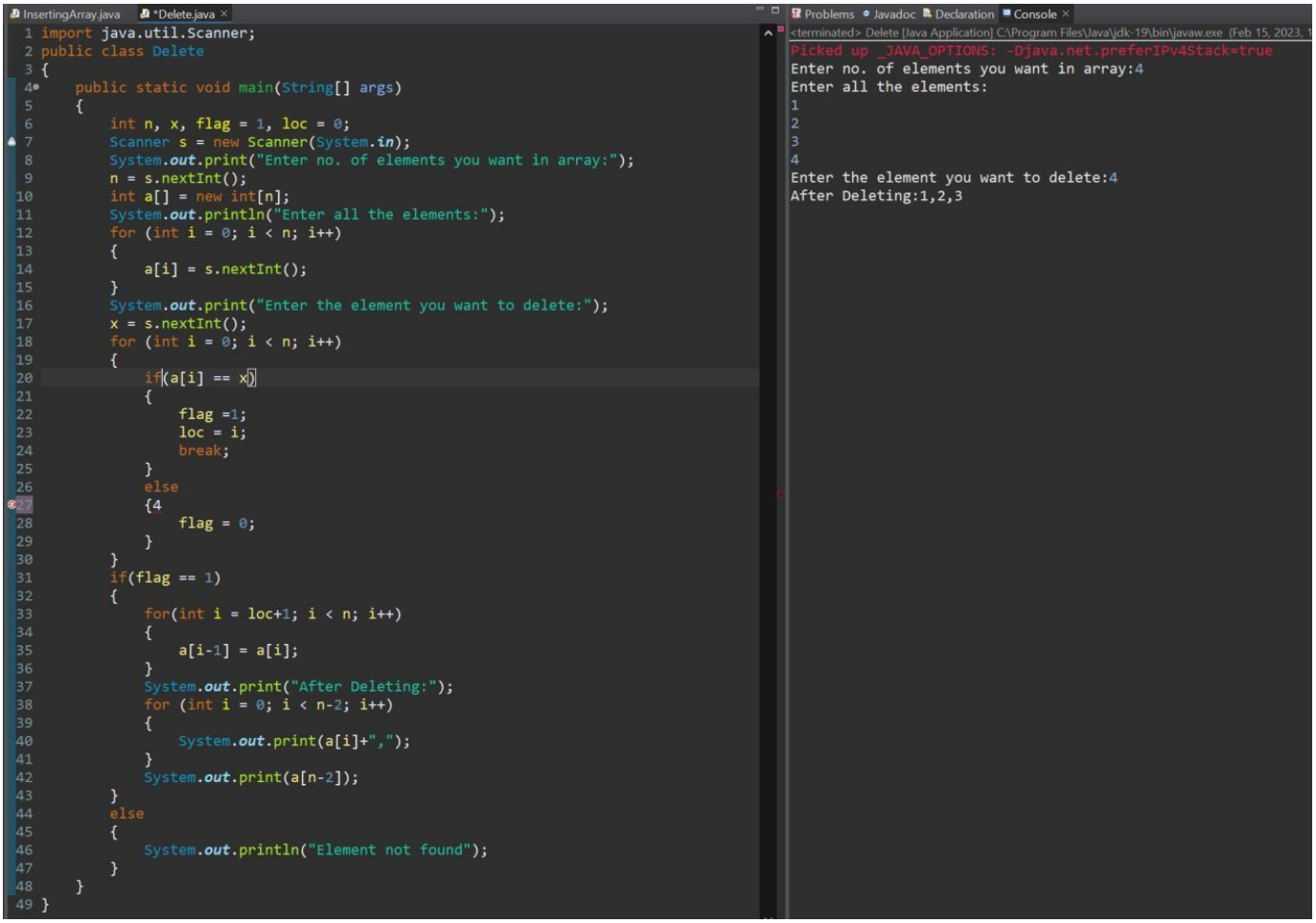
```
import java.util.Scanner;

public class InsertingArray
{
    public static void main(String[] args)
    {
        int n, pos, x;
        Scanner s = new Scanner(System.in); System.out.print("Enter no. of elements you want in array: ");
        n = s.nextInt();
        int a[] = new int[n+1];
        System.out.println("Enter all the elements:");
        for (int i = 0; i < n; i++)
        {
            a[i]= s.nextInt();
        }
        System.out.print("Enter the position where you want to insert element:");
        pos = s.nextInt();
        System.out.print("Enter the element you want to insert: ");
        x = s.nextInt();
        for (int i = (n-1); i >= (pos-1); i--)
        {
            a[i+1] = a[i];
        }
        a[pos-1] = x;
```

Problems Javadoc Declaration Console

```
<terminated> InsertingArray [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe (Feb 15, 2023, 11:30:07 AM – 11:30:14 AM) [pid: 14016]
Picked up _JAVA_OPTIONS: -Djava.net.preferIPv4Stack=true
Enter no. of elements you want in array: 2
Enter all the elements:
22
11
Enter the position where you want to insert element:1
Enter the element you want to insert: 2
After inserting: 2,22,11
```

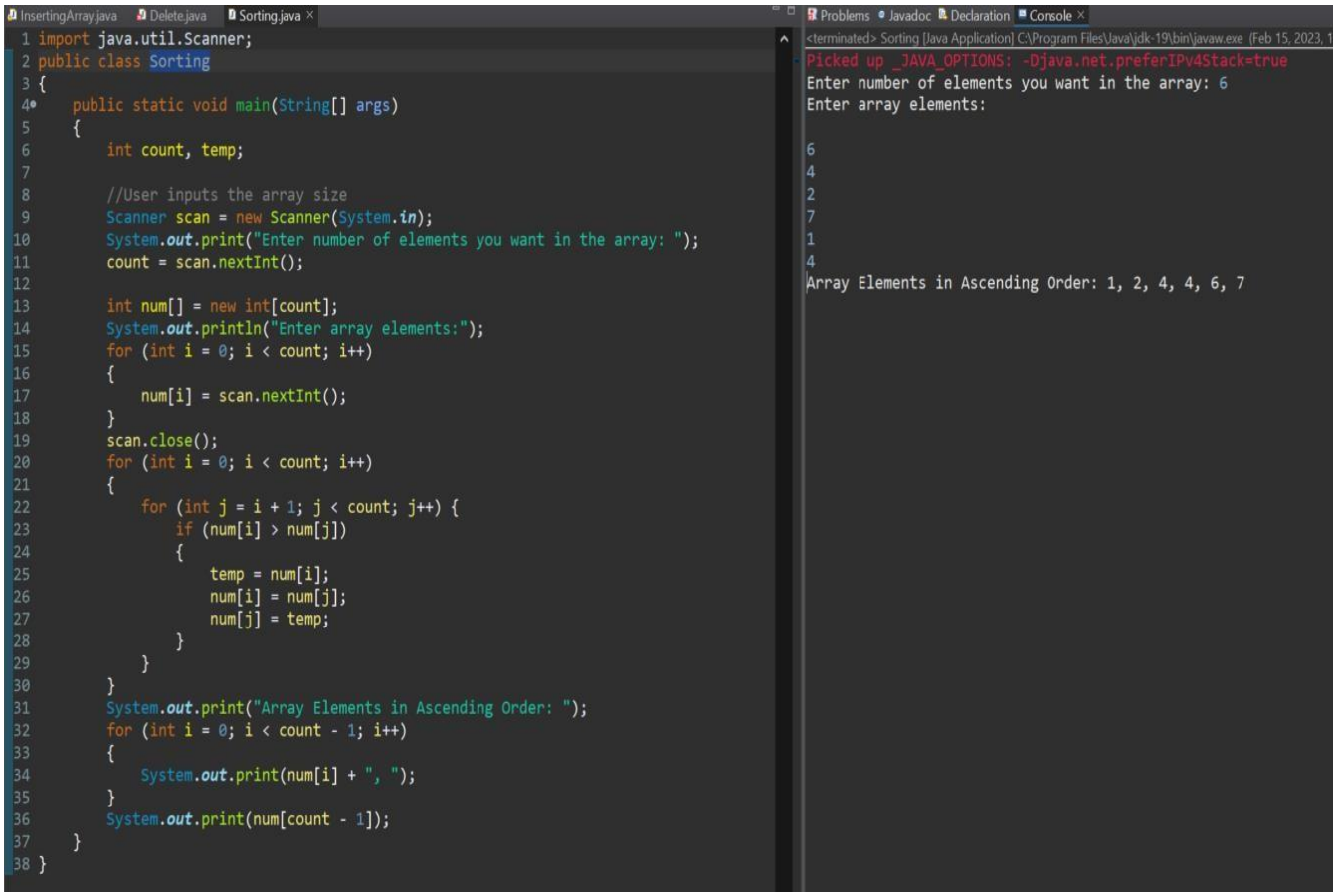
2. Implement a program for deleting an element from the specified position of an array.



```
import java.util.Scanner;
public class Delete
{
    public static void main(String[] args)
    {
        int n, x, flag = 1, loc = 0;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter no. of elements you want in array:");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.println("Enter all the elements:");
        for (int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
        System.out.print("Enter the element you want to delete:");
        x = s.nextInt();
        for (int i = 0; i < n; i++)
        {
            if(a[i] == x)
            {
                flag =1;
                loc = i;
                break;
            }
            else
            {4
                flag = 0;
            }
        }
        if(flag == 1)
        {
            for(int i = loc+1; i < n; i++)
            {
                a[i-1] = a[i];
            }
            System.out.print("After Deleting:");
            for (int i = 0; i < n-2; i++)
            {
                System.out.print(a[i]+",");
            }
            System.out.print(a[n-2]);
        }
        else
        {
            System.out.println("Element not found");
        }
    }
}
```

```
Picked up _JAVA_OPTIONS: -Djava.net.preferIPv4Stack=true
Enter no. of elements you want in array:4
Enter all the elements:
1
2
3
4
Enter the element you want to delete:4
After Deleting:1,2,3
```

3. Implement a program for sorting a given set of numbers.



```
import java.util.Scanner;
public class Sorting
{
    public static void main(String[] args)
    {
        int count, temp;

        //User inputs the array size
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter number of elements you want in the array: ");
        count = scan.nextInt();

        int num[] = new int[count];
        System.out.println("Enter array elements:");
        for (int i = 0; i < count; i++)
        {
            num[i] = scan.nextInt();
        }
        scan.close();
        for (int i = 0; i < count; i++)
        {
            for (int j = i + 1; j < count; j++) {
                if (num[i] > num[j])
                {
                    temp = num[i];
                    num[i] = num[j];
                    num[j] = temp;
                }
            }
        }
        System.out.print("Array Elements in Ascending Order: ");
        for (int i = 0; i < count - 1; i++)
        {
            System.out.print(num[i] + ", ");
        }
        System.out.print(num[count - 1]);
    }
}
```

```
Picked up _JAVA_OPTIONS: -Djava.net.preferIPv4Stack=true
Enter number of elements you want in the array: 6
Enter array elements:

6
4
2
7
1
4
Array Elements in Ascending Order: 1, 2, 4, 4, 6, 7
```

AIM 2: Understanding the concepts of stack, its implementations and applications. (5 points)

- Stack is linear data structure in which addition or deletion takes place at the same end. This end is called the top of stack. Examples of stack are: Stack of plates, Stack of Books etc. Stack is a sequence of items, which can be added and removed from one end only.
- Stack is known as LIFO (last in first out).
- Insert Operation (PUSH) Stacks can be implemented using arrays by defining a structure containing an array and variable to indicate the position of top of stack. PUSH – add data x to stack Increment top and then set data[top]= x.
- Delete Operation (POP) POP-remove and return data from stack Return data[top] and decrement top.

1. Implement a program for creating a new stack, adding element to the stack, removing elements from stack.

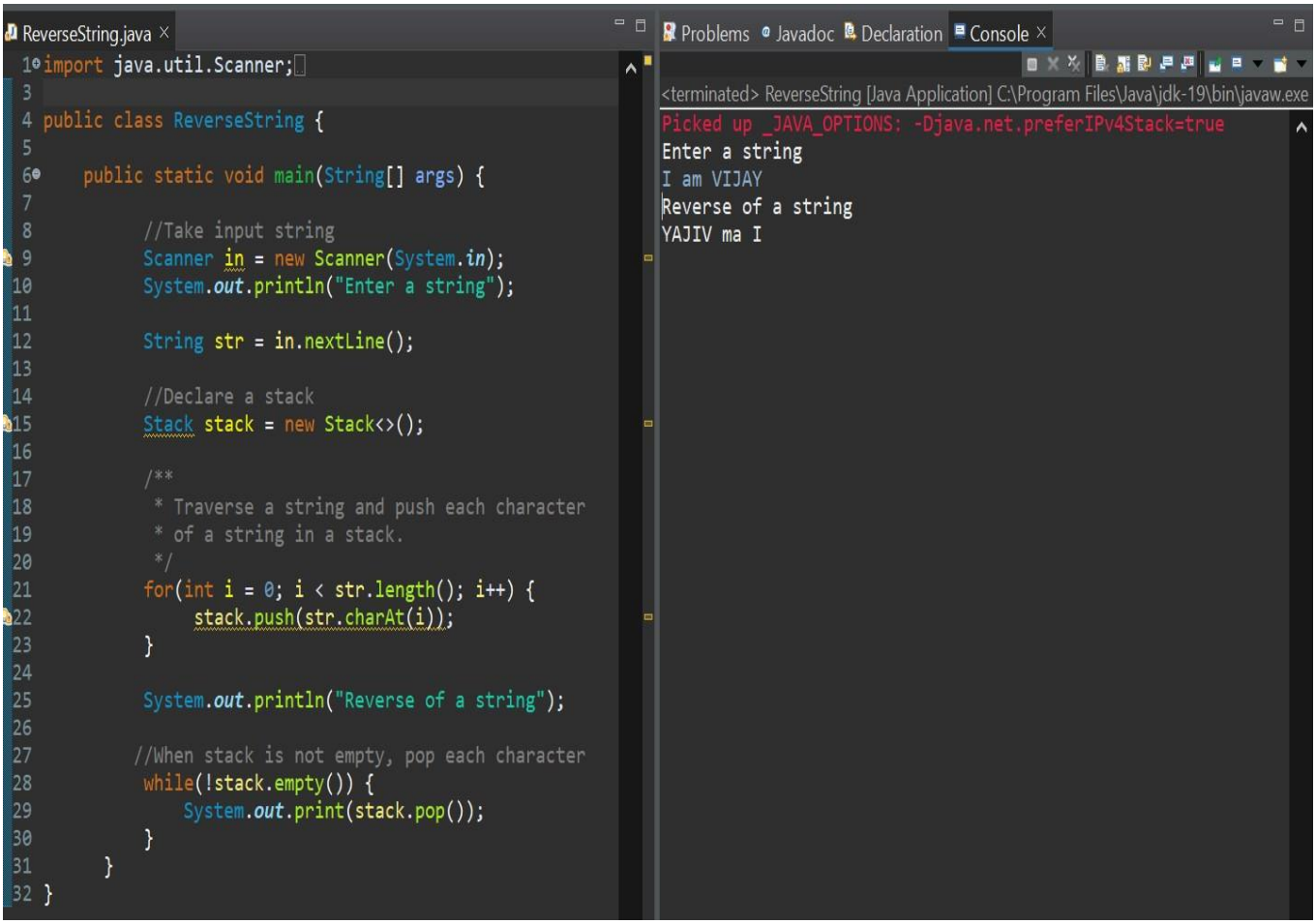
stack.java

```
1  import java.util.Stack;
2
3  public class stack {
4      public static void main(String[] args) {
5          Stack<Integer> stack = new Stack<>();
6
7          // Add elements to the top of the stack
8          stack.push(1);
9          stack.push(2);
10         stack.push(3);
11
12         // Print the elements of the stack
13         System.out.println("Elements of stack after adding 1, 2, and 3: " + stack);
14
15         // Remove an element from the top of the stack
16         stack.pop();
17
18         // Print the elements of the stack after removal
19         System.out.println("Elements of stack after removing one element: " + stack);
20
21         // Remove the remaining elements from the top of the stack
22         stack.pop();
23         stack.pop();
24
25         // Print the elements of the stack after removing all elements
26         System.out.println("Elements of stack after removing all elements: " + stack);
27     }
28 }
```

Problems Javadoc Declaration Console

```
<terminated> stack [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe (Feb 21, 2023, 11:59:39 AM – 11:59:39 AM) [pid: 2486
Picked up _JAVA_OPTIONS: -Djava.net.preferIPv4Stack=true
Elements of stack after adding 1, 2, and 3: [1, 2, 3]
Elements of stack after removing one element: [1, 2]
Elements of stack after removing all elements: []
```

2. Implement a program to reverse a given string using stack



ReverseString.java

```
1 import java.util.Scanner;
3
4 public class ReverseString {
5
6     public static void main(String[] args) {
7
8         //Take input string
9         Scanner in = new Scanner(System.in);
10        System.out.println("Enter a string");
11
12        String str = in.nextLine();
13
14        //Declare a stack
15        Stack stack = new Stack<>();
16
17        /**
18         * Traverse a string and push each character
19         * of a string in a stack.
20         */
21        for(int i = 0; i < str.length(); i++) {
22            stack.push(str.charAt(i));
23        }
24
25        System.out.println("Reverse of a string");
26
27       //When stack is not empty, pop each character
28        while(!stack.empty()) {
29            System.out.print(stack.pop());
30        }
31    }
32 }
```

Problems Javadoc Declaration Console

```
<terminated> ReverseString [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe
Picked up _JAVA_OPTIONS: -Djava.net.preferIPv4Stack=true
Enter a string
I am VIJAY
Reverse of a string
YAJIV ma I
```