

Experiment No. 07
Experiment Name: Experiment with Java interfaces

Course title: Programming Language II(Java) Lab
Course code:
Spring 2025

Date of Submission:

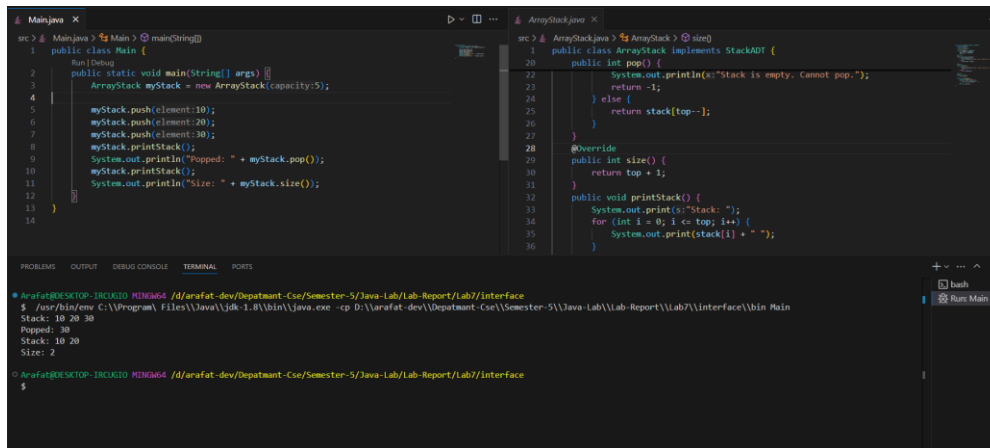


Submitted to-

Md. Rafsan Jani
Assistant Professor
Department of Computer Science and Engineering

Sl	Class Roll	Name
01	2023000010001	Md Samaul Islam

Screenshot:



```
src > Main.java
1 public class Main {
2     public static void main(String[] args) {
3         ArrayStack myStack = new ArrayStack(capacity:5);
4
5         myStack.push(element:10);
6         myStack.push(element:20);
7         myStack.push(element:30);
8         myStack.printStack();
9         System.out.println("Popped: " + myStack.pop());
10        myStack.printStack();
11        System.out.println("Size: " + myStack.size());
12    }
13 }
14
```

```
src > ArrayStack.java
1 public class ArrayStack implements StackADT {
2     public int pop() {
3         System.out.println("Stack is empty. Cannot pop.");
4         return -1;
5     } else {
6         return stack[top--];
7     }
8 }
9
10 @Override
11 public int size() {
12     return top + 1;
13 }
14
15 public void printStack() {
16     System.out.print("Stack: ");
17     for (int i = 0; i <= top; i++) {
18         System.out.print(stack[i] + " ");
19     }
20 }
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Arafat@DESKTOP-INC630 PID64 /d/arafat-dev/Depatmant-Cse/Semester-5/Java-Lab/Lab-Report/Lab7/interface
$ java/bin/java -cp D:\arafat-dev\Depatmant-Cse\Semester-5\Java-Lab\Lab-Report\Lab7\interface\bin Main
Stack: 10 20 30
Popped: 30
Stack: 10 20
Size: 2
```

Main.java

```
public class Main {
    public static void main(String[] args) {
        ArrayStack myStack = new ArrayStack(5);

        myStack.push(10);
        myStack.push(20);
        myStack.push(30);
        myStack.printStack();
        System.out.println("Popped: " + myStack.pop());
        myStack.printStack();
        System.out.println("Size: " + myStack.size());
    }
}
```

StackADT.java

```
public interface StackADT {
    void push(int element);
    int pop();
    int size();
}
```

```

public class ArrayStack implements StackADT {
    private int[] stack;
    private int top;
    private int capacity;
    // Constructor
    public ArrayStack(int capacity) {
        this.capacity = capacity;
        stack = new int[capacity];
        top = -1;
    }
    @Override
    public void push(int element) {
        if (top == capacity - 1) {
            System.out.println("Stack is full. Cannot push.");
        } else {
            stack[++top] = element;
        }
    }
    @Override
    public int pop() {
        if (top == -1) {
            System.out.println("Stack is empty. Cannot pop.");
            return -1;
        } else {
            return stack[top--];
        }
    }
    @Override
    public int size() {
        return top + 1;
    }
    public void printStack() {
        System.out.print("Stack: ");
        for (int i = 0; i <= top; i++) {
            System.out.print(stack[i] + " ");
        }
        System.out.println();
    }
}

```