**Name: Sreelakshmi Madhusoodhanan**

**Roll No: 39**

**Batch: RMCA**

**Date:01-06-2022**

**Object Oriented Programming LAB**

**Experiment No.: 23**

**Aim**

Program to create a generic stack and do the Push and Pop operations***.***

**Procedures**

**Source Code**

import java.util.\*;

class operations{

public void operation()

{

int top =-1,ch,n,e;

Scanner inp = new Scanner(System.in);

System.out.println("Enter Size of Stack");

n = inp.nextInt();

int size=n-1;

int[] arr = new int[n];

do {

System.out.println("\n=============\n MENU : \n1.push \n2.pop \n3.Display \n4.Exit \n=============");

System.out.println("Enter your choice");

ch = inp.nextInt();

switch(ch)

{

case 1 :

if(top == size)

{

System.out.println(" \*\*\* Stack is Full \*\*\* ");

}

else

{

System.out.println("Enter Element : ");

e = inp.nextInt();

top++;

arr[top] =e;

}

break;

case 2 :

if(top == -1)

{

System.out.println("\n\*\*\* Stack is empty \*\*\* ");

}

else

{

System.out.println("\n"+ arr[top] + " is removed ");

top--;

}

break;

case 3 :

if(top == -1)

{

System.out.println(" \*\*\* Stack is empty \*\*\*");

}

else

{

System.out.println("\n\*\*\* Stack : \*\*\*\n");

for(int i=top;i>=0;i--)

{

System.out.println(" " +arr[i]);

System.out.println("-----");

}

}

break;

case 4 :

System.exit(0);

default : System.out.println("Invalid Choice");

}

}while(ch !=4);

}

}

public class Stackopertaion{

public static void main(String[] args) {

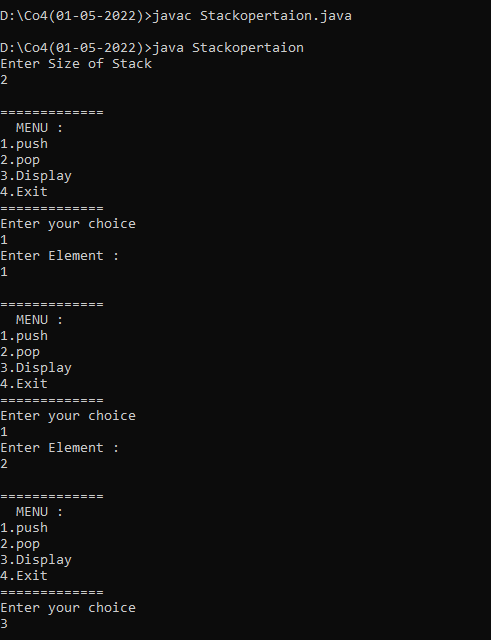
operations obj = new operations();

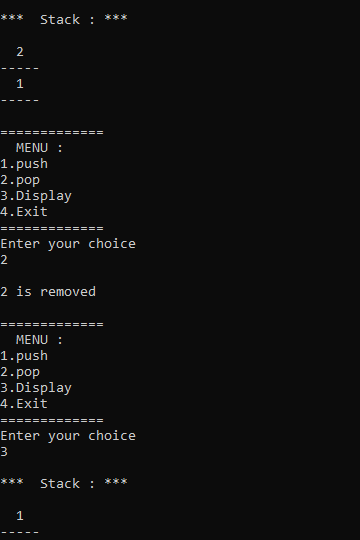
obj.operation();

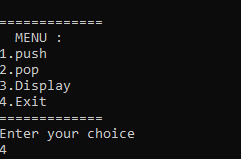
}

}

**Output**







**Experiment No: 24**

**Name: Sreelakshmi Madhusoodhanan**

**Roll No:39**

**Batch: RMCA B**

**Date:01/06/2022**

**Aim**

Define 2 classes; one for generating Fibonacci numbers and other for displaying even numbers in a given range. Implement using threads. (Runnable Interface).

**Procedure**

iimport java.util.\*;

class fibonacci implements Runnable {

int l;

fibonacci(int n) {

l = n;

}

public void run() {

int c;

int a = 0, b = 1;

System.out.print("Fibonacci:");

System.out.print(a + " " + b);

for (int i = 0; i <= l; i++) {

c = a + b;

System.out.print(" " + c);

a = b;

b = c;

}

}

}

class even implements Runnable {

int l;

even(int n) {

l = n;

}

public void run() {

System.out.print("Even Number:");

for (int i = 0; i <= l; i++) {

if (i % 2 == 0)

System.out.print(i + " ");

}

System.out.println("");

}

}

class Numbers{

public static void main(String args[]) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter Limit :");

int l = sc.nextInt();

even e = new even(l);

Thread T2 = new Thread(e);

T2.start();

fibonacci f = new fibonacci(l);

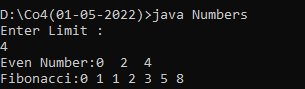
Thread T1 = new Thread(f);

T1.start();

}

}

**Output Screenshot**



**Experiment No: 25**

**Name: Sreelakshmi Madhusoodhanan**

**Roll No:39**

**Batch: RMCA B**

**Date:01/06/2022**

**Aim**

Find the average of N positive integers, raising a user defined exception for each negative input.

**Procedure**

import java.util.Scanner;

public class IntegerNum{

public static class InvalidNumberException extends Exception {

public InvalidNumberException() {

super("Please provide a valid number!");

}

}

public static void main(String [] args){

Scanner sc=new Scanner(System.in);

int c,num,sum=0;

double avg;

System.out.println("enter the count:");

c=sc.nextInt();

System.out.println("enter the Numbers:");

for(int i=0;i<c;i++){

try{

num=sc.nextInt();

if(num>0){

sum+=num;

}else{

i--;

throw new InvalidNumberException();

}

}

catch(InvalidNumberException e){

System.out.println(e.getMessage());

}

}

System.out.println("sum is"+sum);

System.out.println("avg is"+sum/c);

}

}

**Output Screenshot**

