**SET A**

**Q1.**

1 import java.util.\*;  
 2 public class Stack\_Impl{  
 3 static int top=-1;  
 4 public static void main(String args[]){  
 5 Scanner sc = new Scanner(System.in);  
 6 System.out.println("Enter the size of stack: ");  
 7 int size= sc.nextInt();  
 8 int[] a = new int[size];  
 9 while(true){  
10 System.out.println("\nPress 1 to Push a stack element ");  
11 System.out.println("Press 2 to Pop a stack element ");  
12 System.out.println("Press 3 to topmost element ");  
13 System.out.println("Press 4 to Print all the stack elements ");  
14 System.out.println("Press 5 to quit");  
15 int choice = sc.nextInt();  
16 if(choice == 1){  
17 System.out.print("\nEnter the element to be pushed : ");  
18 int element = sc.nextInt();  
19 push(element,size,a);  
20 }  
21 else if(choice == 2){  
22 System.out.print("\nPopping the last entered digit ");  
23 pop(a);  
24 }  
25 else if(choice == 3){  
26 System.out.print("\nPeeking the topmost element :");  
27 peek(a);  
28 }  
29 else if(choice == 4){  
30 System.out.print("\nPrinting the stack");  
31 print(a);  
32   
33 }  
34 else if( choice == 5){  
35 break;  
36 }  
37 else  
38 System.out.print("Invalid Input");  
39 }   
40 }  
41   
42 public static void push(int e,int size,int[] array){  
43 top=top+1;  
44 if(top<=size-1){   
45 array[top] = e;  
46 }  
47 else  
48 System.out.print("STACK OVERFLOW");  
49 }  
50   
51 public static void pop(int[] array){  
52 if(top>0){  
53 array[top]=0;  
54 top=top-1;   
55 }  
56 else if(top==0){  
57 array[top]=0;  
58 top=-1;  
59 }  
60 else  
61 System.out.print("STACK UNDERFLOW");  
62 }  
63   
64 public static void peek(int[] array){  
65 System.out.print("\n"+array[top]);  
66 }  
67   
68 public static void print(int[] array){  
69 System.out.println("\n");  
70 if(top!=-1){  
71 for(int i =0; i<=top; i++){  
72 System.out.println(array[i]);  
73 }  
74 }   
75 if(top==-1){  
76 System.out.print("\nStack is Empty");  
77 }   
78 }  
79 }

**Q2.**

1 import java.util.\*;  
 2 class Node\_Top{  
 3 int data;  
 4 Node\_Top next ;  
 5 Node\_Top(){  
 6 data = 0;  
 7 next = null;  
 8 }  
 9 }  
10 public class StackArray\_Q2{  
11 static Node\_Top top;  
12 public static void main(String args[]){  
13 Scanner sc = new Scanner(System.in);  
14 int choice = 0;  
15 do{  
16 System.out.println(" Enter 1 to add element in stack.\n Enter 2 to pop elements.\n Enter 3 to print top element.\n Enter 4 to print stack.\n Enter 5 to exit.");  
17 choice = sc.nextInt();  
18 if(choice==1){  
19 System.out.print("Push element in stack: ");  
20 int a = sc.nextInt();  
21 push(a);  
22 }  
23 else if(choice==5) System.out.println("Program terminated.");  
24 else if(choice==4){  
25 print();  
26 }  
27 else if(choice==2){  
28 pop();  
29 }  
30 else if(choice==3){  
31 peek();  
32 }  
33 else System.out.println("Invalid Input, Enter again ");  
34 }while(choice!=5);  
35 }  
36 public static void push(int a){  
37 Node\_Top n = new Node\_Top();  
38 n.data = a;  
39 n.next=top;  
40 top = n;  
41 }  
42 public static void print(){  
43 Node\_Top temp = new Node\_Top();  
44 temp = top;  
45 if(top == null) System.out.println("Stack underflow");  
46 else{  
47 while(temp.next!=null){  
48 System.out.print(temp.data+" ");  
49 temp = temp.next;  
50 }  
51 System.out.print(temp.data);  
52 System.out.println();  
53 }  
54 }  
55 public static void pop(){  
56 Node\_Top temp = new Node\_Top();  
57 temp = top;  
58 if(top==null) System.out.println("Stack underflow");  
59 else{  
60 System.out.println("element : "+top.data);  
61 top = top.next;  
62 temp.next = null;  
63 }  
64 }  
65 public static void peek(){  
66 System.out.println("Node\_Top element : "+top.data);  
67 }  
68 }

**Q3.**

1 import java.util.\*;  
 2 public class Stack\_arr\_Q3{  
 3 static int top = -1;  
 4 public static void main(String args[]){  
 5 Scanner sc = new Scanner(System.in);  
 6 System.out.print("\nEnter the String : ");  
 7 String word = sc.next();  
 8 String[] a = word.split("\\s+");  
 9 int x = a.length;  
10 System.out.print(a[0]+a[1]+a[2]+a[3]);  
11 for(int i = 0; i <x; i++){  
12 push(a[i],x,a);  
13 }  
14 reverse(x,a);  
15   
16   
17 }  
18   
19 public static void push(String word,int size,String[] array){  
20 top=top+1;  
21 if(top<=size-1){   
22 array[top] = word ;  
23 }  
24 else  
25 System.out.print("STACK OVERFLOW");  
26   
27 }  
28   
29   
30 public static void reverse(int x, String[] words){  
31 for(int i = x-1; i>=0; i--){  
32 System.out.print("\n"+words[i]);  
33 }  
34 }  
35 }

**Q4.**

1 import java.util.\*;  
 2 class StackQ4{  
 3 static int top = -1;  
 4 public static void main(String args[]){  
 5 Scanner sc = new Scanner(System.in);  
 6 System.out.print("Enter a string of brackets : ");  
 7 String st = sc.next();  
 8 String[] str = st.split("");  
 9 String[] array = new String[str.length];  
10 check(str,array);  
11 }  
12 public static void check(String[] str, String[] array){  
13 for(int i = 0; i<str.length;i++){  
14 if(str[i].equals("(") || str[i].equals("[") || str[i].equals("{")){  
15 push(str[i],array);  
16 }  
17 else if(str[i].equals(")")){  
18 if(array[top].equals("(")) pop(array);  
19 else{  
20 System.out.println("Not Balanced");  
21 System.exit(0);  
22 }  
23 }  
24 else if(str[i].equals("]")){  
25 if(array[top].equals("[")) pop(array);  
26 else{  
27 System.out.println("Not Balanced");  
28 System.exit(0);  
29 }  
30 }  
31 else if(str[i].equals("}")){  
32 if(array[top].equals("{")) pop(array);  
33 else{  
34 System.out.println("Not Balanced");  
35 System.exit(0);  
36 }  
37 }  
38 }  
39 if(top==-1) System.out.println("Balanced");  
40 else System.out.println("Not balanced");  
41 }  
42 public static void pop(String array[]){  
43 if(top==-1){  
44 System.out.println("Stack underflow!!!");  
45 }  
46 else{  
47 top--;  
48 }  
49 }  
50 public static void push(String a,String array[]){  
51 top++;  
52 array[top] = a;  
53 }  
54 }

**Q5.**

1 import java.util.\*;  
 2 public class Stack\_Q5{  
 3 public static void main(String args[]){  
 4 Scanner sc = new Scanner(System.in);  
 5 System.out.print("\nEnter the number of disks: ");  
 6 int n = sc.nextInt();  
 7 t(n,'A','B','C');  
 8 }  
 9 public static void t(int n, char Begin, char Aux, char End){  
10 if(n==1){  
11 System.out.print("\n"+Begin+"-->"+End);  
12 }  
13 else{  
14 t(n-1,Begin,End,Aux);  
15 System.out.print("\n"+Begin+"-->"+End);  
16 t(n-1,Aux,Begin,End);  
17 }  
18 }  
19   
20 }

**SET B**

**Q6.**

1 import java.util.\*;  
 2 class StackkkkkkkQ6{  
 3 static int top = -1;  
 4 public static void main(String args[]){  
 5 Scanner sc = new Scanner(System.in);  
 6 int[] array;  
 7 int size = 0;  
 8 System.out.print("Enter no.of elements: ");  
 9 size = sc.nextInt();  
10 array = new int[size];  
11 System.out.println("Push elements in stack");  
12 for(int i =0; i<size ; i++){  
13 System.out.print("Enter "+(i+1)+" element: ");  
14 int a = sc.nextInt();  
15 push(a,array);  
16 }  
17 RemoveOdd(array);  
18 System.out.println("Stack after removing odd elements.");  
19 print(array);  
20 }  
21 public static void push(int a,int array[]){  
22 if(top==array.length-1){  
23 System.out.println("Stack overflow!!!");  
24 }  
25 else{  
26 top++;  
27 array[top] = a;  
28 }  
29 }  
30 public static void pop(int array[]){  
31 if(top==-1){  
32 System.out.println("Stack underflow!!!");  
33 }  
34 else{  
35 top--;  
36 }  
37 }  
38 public static void print(int array[]){  
39 if(top==-1) System.out.println("Empty Stack.");  
40 else{  
41 for(int i = top; i>=0; i--){  
42 System.out.print(array[i]+" ");  
43 }  
44 System.out.println();  
45 }  
46 }  
47 public static void RemoveOdd(int array[]){  
48 if(top==-1) System.out.println("Empty Stack.");  
49 else{  
50 int top1 = -1;  
51 int temp[] = new int[array.length];  
52 while(top>=0){  
53 if(array[top]%2 == 0){  
54 top1++;  
55 temp[top1] = array[top];  
56 pop(array);  
57 }  
58 else pop(array);  
59 }  
60 while(top1>=0){  
61 push(temp[top1],array);  
62 top1--;  
63 }  
64 }  
65 }   
66 }

**Q7.**

1 import java.util.\*;  
 2 class Postfix{  
 3 static int top = -1;  
 4 public static void main(String args[]){  
 5 Scanner sc = new Scanner(System.in);  
 6 System.out.print("Enter expression: ");  
 7 String exp = sc.next();  
 8 exp = exp + ")";  
 9 String[] array = new String[exp.length()];  
 10 push("(",array);  
 11 expre(exp,array);  
 12 }  
 13 public static void push(String a,String array[]){  
 14 top++;  
 15 array[top] = a;  
 16 }  
 17 public static void pop(String array[]){  
 18 if(top==-1){  
 19 System.out.println("Stack underflow!!!");  
 20 }  
 21 else{  
 22 top--;  
 23 }  
 24 }  
 25 public static void expre(String exp, String[] array){  
 26 String post = "";  
 27 String[] exp1 = exp.split("");  
 28 boolean g = true;  
 29 for(String i : exp1){  
 30 try {   
 31 Double.parseDouble(i);   
 32 g = true;  
 33 } catch(NumberFormatException e){   
 34 g = false;   
 35 }  
 36 if(g){  
 37 post = post + i;  
 38 }  
 39 else if(i.equals("(")){  
 40 // if(!array[top].equals("(")){  
 41 // String k = array[top];  
 42 // pop(array);  
 43 // while(!array[top].equals("(")){  
 44 // post = post + array[top];  
 45 // pop(array);  
 46 // }  
 47 // push(k,array);  
 48 // }  
 49 push(i,array);  
 50 }  
 51 else if(i.equals(")") && g == false){  
 52 while(!array[top].equals("(")){  
 53 post = post + array[top];  
 54 pop(array);  
 55 }  
 56 pop(array);  
 57 }  
 58 else if(!(i.equals(")")) && g==false){  
 59 if((i.equals("+") || i.equals("-"))){  
 60 if(array[top].equals("(")) push(i,array);  
 61 else if(top != -1){  
 62 while(!array[top].equals("(")){  
 63 post = post + array[top];  
 64 pop(array);  
 65 }  
 66 push(i,array);  
 67 }  
 68 }  
 69   
 70 else if((i.equals("\*") || i.equals("/")) && top != -1){  
 71 if((array[top].equals("\*") || array[top].equals("/") || array[top].equals("^"))){  
 72 if(top != -1 && array[top] != "("){  
 73 while(!array[top].equals("(") && !array[top].equals("^")){  
 74 post = post + array[top];  
 75 pop(array);  
 76 }  
 77 push(i,array);  
 78 }  
 79 else push(i,array);  
 80 }  
 81 else{  
 82 push(i,array);  
 83 }  
 84 }  
 85 else{  
 86 push(i,array);  
 87 }  
 88 }  
 89 }  
 90 System.out.println(post);  
 91 evaluate(post,array);  
 92 }  
 93 public static void evaluate(String post, String[] array){  
 94 String[] str1 = post.split("");  
 95 for(int i = 0; i<str1.length;i++){  
 96 boolean g = true;  
 97 try {   
 98 Double.parseDouble(str1[i]);   
 99 g = true;  
100 } catch(NumberFormatException e){   
101 g = false;   
102 }  
103 if(g){  
104 push(str1[i],array);  
105 }  
106 else{  
107 if(str1[i].equals("\*")){  
108 String b = array[top];  
109 pop(array);  
110 String a = array[top];  
111 pop(array);  
112 String c =Double.toString(Double.parseDouble(a)\*Double.parseDouble(b));  
113 push(c,array);  
114 }  
115 else if(str1[i].equals("/")){  
116 String b = array[top];  
117 pop(array);  
118 String a = array[top];  
119 pop(array);  
120 String c =Double.toString(Double.parseDouble(a)/Double.parseDouble(b));  
121 push(c,array);  
122 }  
123 else if(str1[i].equals("+")){  
124 String b = array[top];  
125 pop(array);  
126 String a = array[top];  
127 pop(array);  
128 String c =Double.toString(Double.parseDouble(a)+Double.parseDouble(b));  
129 push(c,array);  
130 }  
131 else if(str1[i].equals("-")){  
132 String b = array[top];  
133 pop(array);  
134 String a = array[top];  
135 pop(array);  
136 String c =Double.toString(Double.parseDouble(a)-Double.parseDouble(b));  
137 push(c,array);  
138 }  
139 else if(str1[i].equals("^")){  
140 String b = array[top];  
141 pop(array);  
142 String a = array[top];  
143 pop(array);  
144 double l = 1;  
145 for(int j = 0;j<Integer.parseInt(b);j++){  
146 l = l\*Integer.parseInt(a);  
147 }  
148 String c =Double.toString(l);  
149 push(c,array);  
150 }  
151 }  
152 }  
153 System.out.println(array[top]);  
154 }  
155 }

**Q8.**

1 import java.util.\*;  
 2 class ReverseIndi{  
 3 static int top =-1;  
 4 public static void main(String args[]){  
 5 String result = "";  
 6 Scanner sc = new Scanner(System.in);  
 7 System.out.print("Enter String: ");  
 8 String str = sc.nextLine();  
 9 String[] array = new String[str.length()];  
10 String[] str1= str.split(" ");  
11 for(int i = 0; i<str1.length; i++){  
12 String[] str2 = str1[i].split("");  
13 for(int j = 0; j<str2.length; j++){  
14 push(str2[j],array);  
15 }  
16 while(top>=0){  
17 result = result+array[top];  
18 pop(array);  
19 }  
20 result = result+" ";  
21 }  
22 System.out.println(result);  
23 }  
24 public static void push(String a,String array[]){  
25 if(top==array.length-1){  
26 System.out.println("Stack overflow!!!");  
27 }  
28 else{  
29 top++;  
30 array[top] = a;  
31 }  
32 }  
33 public static void pop(String array[]){  
34 if(top==-1){  
35 System.out.println("Stack underflow!!!");  
36 }  
37 else{  
38 top--;  
39 }  
40 }  
41 }

**SET C**

**Q10.**

1 import java.util.\*;  
 2 public class Stack\_SetC\_Q2{  
 3 static int top1=-1;  
 4 static int top2=-1;  
 5 static int top3=-1;  
 6 public static void main(String args[]){  
 7 Scanner sc = new Scanner(System.in);  
 8 System.out.print("\nEnter the size of STACK 1 :");  
 9 int size1 = sc.nextInt();  
 10 System.out.print("\nEnter the size of STACK 2 :");  
 11 int size2 = sc.nextInt();  
 12 System.out.print("\nEnter the size of STACK 3 :");  
 13 int size3 = sc.nextInt();  
 14 int[] s1 = new int[size1];  
 15 int[] s2 = new int[size2];  
 16 int[] s3 = new int[size3];  
 17 int sum1=0,sum2=0,sum3=0;  
 18 for(int i = 0; i<size1; i++){  
 19 System.out.print("\nEnter the STACK 1 element "+(i+1)+" :");  
 20 s1[i]= sc.nextInt();  
 21 sum1=sum1+s1[i];  
 22 push1(s1[i],size1,s1);  
 23   
 24 }  
 25   
 26 for(int i = 0; i<size2; i++){  
 27 System.out.print("\nEnter the STACK 2 element "+(i+1)+" :");  
 28 s2[i]= sc.nextInt();  
 29 sum2=sum2+s2[i];  
 30 push2(s2[i],size2,s2);  
 31   
 32 }  
 33   
 34 for(int i = 0; i<size3; i++){  
 35 System.out.print("\nEnter the STACK 3 element "+(i+1)+" :");  
 36 s3[i]= sc.nextInt();  
 37 sum3=sum3+s3[i];  
 38 push3(s3[i],size3,s3);  
 39   
 40 }  
 41 while(true){  
 42 if(sum1==sum2 && sum2==sum3){  
 43 System.out.print("Maximum sum is : "+sum1);  
 44 }  
 45 else{  
 46 if(sum1>=sum2 && sum1>=sum3){  
 47 while(true){  
 48 sum1=sum1-s1[top1];  
 49 if(top1>=0)  
 50 pop1(s1);  
 51 if(sum1<=sum2 && sum1<=sum3){  
 52 break;  
 53 }  
 54 }  
 55 }  
 56 else if(sum2>=sum1 && sum2>=sum3){  
 57 while(true){  
 58 sum2=sum2-s2[top2];  
 59 if(top2>=0)  
 60 pop2(s2);  
 61 if(sum2<=sum1 && sum2<=sum3){  
 62 break;  
 63 }  
 64 }  
 65 }  
 66 else if(sum3>=sum1 && sum3>=sum2){  
 67 while(true){  
 68 sum3=sum3-s3[top3];  
 69 if(top3>=0)  
 70 pop3(s3);  
 71 if(sum3<=sum1 && sum3<=sum2){  
 72 break;  
 73 }  
 74 }  
 75 }  
 76 }  
 77 if(sum1==sum2 && sum2==sum3){  
 78 System.out.print("Maximum sum is : "+sum1);  
 79 break;  
 80 }  
 81 }  
 82   
 83   
 84 }  
 85   
 86   
 87 /\* print1(s1);  
 88 print2(s2);  
 89 print3(s3);  
 90 \*/  
 91   
 92   
 93 public static void push1(int e,int size,int[] array){  
 94 top1=top1+1;  
 95 if(top1<=size-1){   
 96 array[top1] = e;  
 97 }  
 98 else  
 99 System.out.print("STACK OVERFLOW");  
100 }  
101   
102 public static void push2(int e,int size,int[] array){  
103 top2=top2+1;  
104 if(top2<=size-1){   
105 array[top2] = e;  
106 }  
107 else  
108 System.out.print("STACK OVERFLOW");  
109 }  
110   
111 public static void push3(int e,int size,int[] array){  
112 top3=top3+1;  
113 if(top3<=size-1){   
114 array[top3] = e;  
115 }  
116 else  
117 System.out.print("STACK OVERFLOW");  
118 }  
119   
120 public static void pop1(int[] array){  
121 if(top1>0){  
122 array[top1]=0;  
123 top1=top1-1;   
124 }  
125 else if(top1==0){  
126 array[top1]=0;  
127 top1=-1;  
128 }  
129 else  
130 System.out.print("STACK UNDERFLOW");  
131 }  
132   
133 public static void pop2(int[] array){  
134 if(top2>0){  
135 array[top2]=0;  
136 top2=top2-1;   
137 }  
138 else if(top2==0){  
139 array[top2]=0;  
140 top2=-1;  
141 }  
142 else  
143 System.out.print("STACK UNDERFLOW");  
144 }  
145   
146 public static void pop3(int[] array){  
147 if(top3>0){  
148 array[top3]=0;  
149 top3=top3-1;   
150 }  
151 else if(top3==0){  
152 array[top3]=0;  
153 top3=-1;  
154 }  
155 else  
156 System.out.print("STACK UNDERFLOW");  
157 }  
158   
159 public static void print1(int[] array){  
160 System.out.println("\nSTACK 1");  
161 if(top1!=-1){  
162 for(int i =0; i<=top1; i++){  
163 System.out.println(array[i]);  
164 }  
165 }   
166 if(top1==-1){  
167 System.out.print("\nStack is Empty");  
168 }   
169 }  
170   
171 public static void print2(int[] array){  
172 System.out.println("\nSTACK 2");  
173 if(top2!=-1){  
174 for(int i =0; i<=top2; i++){  
175 System.out.println(array[i]);  
176 }  
177 }   
178 if(top2==-1){  
179 System.out.print("\nStack is Empty");  
180 }   
181 }  
182   
183 public static void print3(int[] array){  
184 System.out.println("\nSTACK 3");  
185 if(top3!=-1){  
186 for(int i =0; i<=top3; i++){  
187 System.out.println(array[i]);  
188 }  
189 }   
190 if(top3==-1){  
191 System.out.print("\nStack is Empty");  
192 }   
193 }  
194   
195 }  
196   
197   
198

**BINAY 19CSU370**

**IOT-A**