



**Jawahar Education Societys Annasaheb Chudaman Patil College of  
Engineering, Kharghar, Navi Mumbai**

**NAME: PRIYUSH BHIMRAO KHOBRADE**

**PRN NO: 211112018**

**SUBJECT: DATA STRUCTURES LAB**

## Practical NO. :- 01

Aim :- Implement Stack ADT using array.

Theory

### STACK

A stack is a linear data structure that follows the LIFO (Last-in first-out). It contains only one pointer top pointer pointing to the topmost element of the stack.

"Stack can be defined as container in which insertion & deletion can be done from one end known as top of the stack."

### Property / operation on the stack

push(): insert element in a stack.

pop(): Delete element from a stack.

isEmpty(): stack is empty or not determine

is full(): stack is full or not determine

peek(): it return the element at the given position.

count(): it return the total no. of element.

change(): it change the element at given position

display(): it print all the element.

### • PUSH operation

#### Algorithm

Start

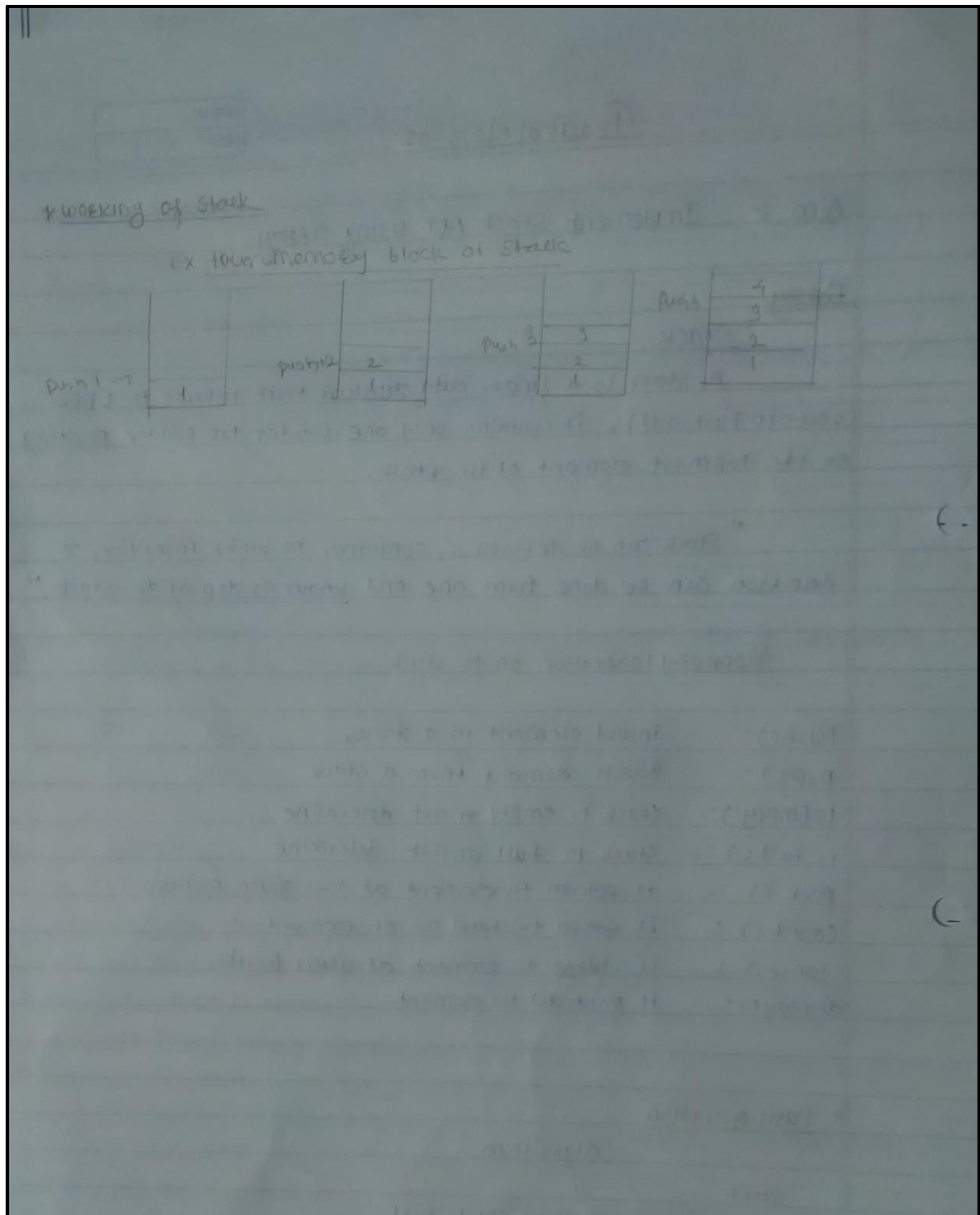
if top = n then stack full

top = top + 1

Stack[top] = item;

end

## AIM: Implement Stack ADT Using array



### Pop operation()

"Deletion of an element from the top of the stack is called pop operation."

### Algorithm

Start

if  $top = 0$  then stack empty;

item: = stack(top);

top = top - 1;

end;

Conclusion:-

Hence, we can understand about stack & its operation.

## AIM: Implement Stack ADT Using array

### Input:-

```
1 #include <stdio.h>
2 #include <conio.h>
3 #include <stdlib.h>
4 #define MAX 100
5
6 int top=-1,stack [MAX];
7 void push();
8 void pop();
9 void display();
10
11 void main()
12 {
13     int ch ;
14     while(1)
15     {
16
17         printf("\n\ STACK OPERATIONS USING ARRAY");
18         printf("\n-----");
19         printf("\nMenu:\n 1.PUSH\n 2.POP \n 3.show\n 4.EXIT\n");
20         printf("Enter Your Choice: \n");
21         scanf("%d", &ch);
22         switch (ch)
23         {
24             case 1: push();
25             break;
26
27             case 2: pop();
28             break;
29
30             case 3: display();
31             break;
32
33             case 4:
34                 exit(0);
35
36             default: printf("Wrong Choice\n");
37
38         }
39     }
40 }
41
42 void push()
43 {
44     int c;
45     if (top==MAX-1)
46     {
47         printf("\nStack is full\n");
48     }
49     else
50     {
51         printf("Enter the element \n");
52         scanf("%d",&c);
53         top=top+1;
54         stack[top]=c;
55     }
56 }
57
58
59
60 void pop()
61 {
62     if (top== -1)
63     {
64         printf("\nStack is empty!\n");
65     }
66     else
67     {
68         printf("\nDelete the element \n",stack[top]);
69         top=top-1;
70     }
71 }
72
73
74 void display()
75 {
76     int k;
77     if (top== -1)
78     {
79         printf("\nStack is empty!\n");
80     }
81     else {
82         printf("\nStack is:\n");
83         for(k=top; k>=0;--k)
84         {
85             printf("%d\n",stack[k]);
86         }
87     }
88     getch();
89 }
90
91
92 }
```

## AIM: Implement Stack ADT Using array

### Output:-

```
"C:\Users\Rupesh\Documents\DS 2ND\Implement Stack ADT using array _Prarticle 1.exe"

STACK OPERATIONS USING ARRAY
-----
Menu:
1.PUSH
2.POP
3.show
4.EXIT
Enter Your Choice:
1
Enter the element
10

STACK OPERATIONS USING ARRAY
-----
Menu:
1.PUSH
2.POP
3.show
4.EXIT
Enter Your Choice:
1
Enter the element
20

STACK OPERATIONS USING ARRAY
-----
Menu:
1.PUSH
2.POP
3.show
4.EXIT
Enter Your Choice:
3

Stack is:
20
10

STACK OPERATIONS USING ARRAY
-----
Menu:
1.PUSH
2.POP
3.show
4.EXIT
Enter Your Choice:
2

Delete the element

STACK OPERATIONS USING ARRAY
-----
Menu:
1.PUSH
2.POP
3.show
4.EXIT
Enter Your Choice:
3

Stack is:
10
```

**Conclusion:** - Hence, we can understanding about stack and its operations.