



THE UNIVERSITY
OF LAHORE
**ISLAMABAD
CAMPUS**

**Data Structures and Algorithms
(CS09203)**

Lab Report

Name: Rehman ullah baig

Reg No: CSU-F16-118

Lab Report: 3

Submitted To: Sir, Usman Ahmed

LAB: 03

Stack with Array implementation

Objective: -

The objective of this session is to understand the various operations on stack using arrays structure in C++.

Software Tools: -

1. Dev C++.

Theory: -

Stacks are the most important in data structures. The notation of a stack in computer science is the same as the notion of the Stack to which you are accustomed in everyday life. For example, a recursion program on which function call itself, but what happen when a function which is calling itself call another function. Such as a function 'A' call function 'B' as a recursion. So, the firstly function 'B' is call in 'A' and then function 'A' is work. So, this is a **Stack**. This is a Stack is **First in Last Out** data structure.

Insertions in Stack:

In Stacks, we know the array work, sometimes we need to modify it or add some element in it. For that purpose, we use insertion scheme. By the use of this scheme we insert any element in Stacks using array. In Stack, we maintain only one node which is called **TOP**. And **Push** terminology is used as insertions.

Deletion in Stack:

In the deletion process, the element of the Stack is deleted on the same node which is called **TOP**. In stacks, it's just deleting the index of the TOP element which is added at last. In Stacks **Pop** terminology is used as deletion.

Display of Stack:

In displaying section, the elements of Stacks are being display by using loops and variables as a reverse order. Such that, last element is display at on first and first element enters display at on last.

Algorithm for top of stack varying method

1. Declare and initialize necessary variables, eg top = -1,

MAXSIZE etc.

2. For push operation, if $top = MAXSIZE - 1$
print "stack overflow"
else

```
top = top + 1;
    Read item from user
stack[top] = item
3. For next push operation, goto step 2.
4. For pop operation,
If top = -1
    print "Stack underflow"
Else
    item = stack[top]
    top = top - 1
    Display item
5. For next pop operation, goto step 4.
6. Stop
```

Lab Task: -

1. Insertion in stack
2. Deletion in stack
3. Display the stack

Conclusion

**(Concerned Teacher/Lab
Engineer)**

Code:

```
#include<iostream>
#include<conio.h>
using namespace std;

#define SIZE 10

void push(int);
void pop();
void display();

int stack[SIZE], top = -1;

int main()
{
    int value, choice;
    system("cls");
    while(1)
    {
        cout<<"\n\n\t\tMENU\n";
        cout<<"PRESSS"<<endl;
        cout<<"1 For Insertion\n2 For Deletion\n3 For Display\n4 To Exit";
        cout<<"\nEnter your choice:";
        cin>>choice;
        switch(choice){
        case 1:
            cout<<"Enter the value to be insert:";
            cin>>value;
            push(value);
            break;
        case 2: pop();
            break;
        case 3: display();
            break;
        case 4: exit(0);
        default:
            cout<<"\nWrong selection!!! Try again!!!";
        }
    }
}

void push(int value){
    if(top == SIZE-1)
        cout<<"\nStack is Full!!! Insertion is not possible!!!";
    else{
        top++;
        stack[top] = value;
        cout<<"\nInsertion success!!!";
    }
}

void pop(){
    if(top == -1)
        cout<<"\nStack is Empty!!! Deletion is not possible!!!";
    else{
```

```

        cout<<"\nDeleted :"<<stack[top];
        top--;
    }
}
void display(){
    if(top == -1)
        cout<<"\nStack is Empty!!!";
    else{
        int i;
        cout<<"\nStack elements are:\n";
        for(i=top; i>=0; i--){
            cin>>stack[i];
        }
    }
}
}

```

Output:

```

E:\UOL\Data structure\DSA Lab\lab 3\lab 3.exe
MENU
PRESSSS
1 For Insertion
2 For Deletion
3 For Display
4 To Exit
Enter your choice:

```

```

E:\UOL\Data structure\DSA Lab\lab 3\lab 3.exe
MENU
PRESSSS
1 For Insertion
2 For Deletion
3 For Display
4 To Exit
Enter your choice:1
Enter the value to be insert:5
Insertion success!!!

```

```

E:\UOL\Data structure\DSA Lab\lab 3\lab 3.exe
1 For Insertion
2 For Deletion
3 For Display
4 To Exit
Enter your choice:2
Deleted :5

```