



## **Data Structures & Algorithms (CS09203)**

### **Lab Report**

Name: Jassim Bashir  
Registration #: SEU-F16-101  
Lab Report #: 03  
Dated: 16-04-2018  
Submitted To: Mr. Usman Ahmed

The University of Lahore, Islamabad Campus  
Department of Computer Science & Information Technology

## Experiment # 3

### Stack with Array implementation

#### Objective

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

#### Software Tool

1. Language: C++
2. Compiler: Dev C++
3. OS: Windows 10

## 1 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 inLast Outdata 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 Pushterminology 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, its just deleting the index of the TOP element which is added at last. In Stacks Popterminologyis used as deletion. Display of Stack:In displaying section, the elements of Stacks are beingdisplay by using loops and variables as a reverse order. Such that, last element isdisplay 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

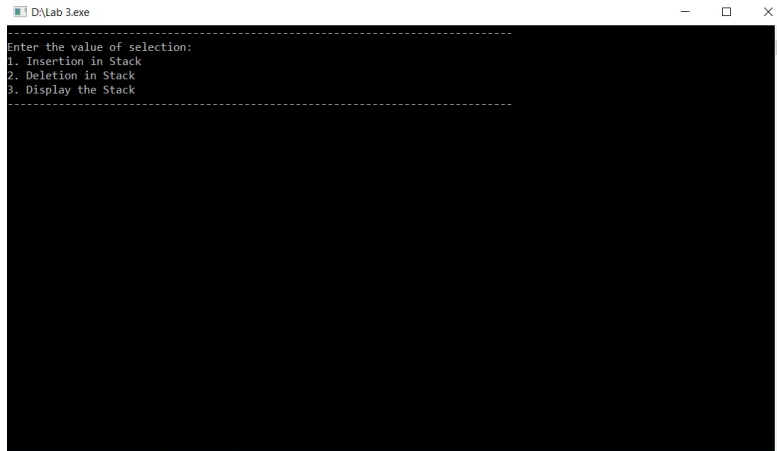


Figure 1: Stack

item = stack[top] top = top -1 Display item 5. For next pop operation, goto step 4. 6. Stop

## 2 Task

### 2.1 Procedure: Task 1

Write a C++ code to perform insertion and deletion in stack using arrays applying the algorithms given in the manual. 1. Insertion in stack 2. Deletion in stack 3. Display the stack

```
#include<iostream>
using namespace std;
int size=10;
int a[10];
int top=-1;
void push(int x){
    if(top==(size-1)){
        cout<<"Error: _Stack_Overflow\n";
        return;
    }
    a[++top]=x;
```

```

}
void pop(){
    if(top== -1){
        cout<<"Error: No element to pop\n";
    }
    top--;
}
void display(){
    cout<<"The values are:"<<endl;
    for(int i=0;i<top+1;i++){
        cout<<a[i]<<endl;
    }
}
int main(){
    int j;
    /*cout<<"Enter the values of the array:"<<endl;
    for(int i=0;i<=size;i++){
    cin>>a[i]<<endl;
    }/
    abc: cout<<"-----\n";
    cout<<"Enter the value of selection:\n";
    cout<<"1. Insertion in Stack\n";
    cout<<"2. Deletion in Stack\n";
    cout<<"3. Display the Stack\n";
    cout<<"-----\n";
    cin>>j;
    switch(j){
        case 1:
            int x;
            cout<<"Enter the value:"<<endl;
            cin>>x;
            push(x);
            goto abc;
            break;
        case 2:
            pop();
            goto abc;
            break;
        case 3:
            display();

```

```

                                break;
                        default :
                                cout<<" Please _try _again !" ;
                }
}

```

### 3 Conclusion

In this lab we learned how to create stack, its functioning and implementation. In this program we learned to add and delete an element from the stack.