Data Structure and Algorithms

Lab Report

Name:	M Ahmed Akram
Registration #:	CSU-F16-111
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. Dev C++

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 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, its 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

2 Lab Task

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

2.1 Program:

```cpp
#include<iostream>
#include<conio.h>
using namespace std;
int top = -1, z[5];
void push(int value)
{
        if(top==4)
        {
                cout<<"\n Stack is Full or Overflow!!\n";
        }
        else
        {
                top++;
                z[top]=value;
        }
```

```cpp
}

void pop()

{
        if(top==-1)
        {
                cout<<"\nStack is Empty or Underflow!!\n";
        }
        else
        {
                top--;
        }
}
void display()
{
        if(top==-1)
        {
                cout<<"\n Nothing to Display!!\n";
        }
        else
        {
                cout<<"\n Array is:";
                for(int i=0;i<=top;i++)
                {
                        cout<<"\t"<<z[i];
                }
        }
}
int main()
{
        int value, choice;
        do
        {
                cout<<"\n 1.PUSH \n 2.POP \n 3.Display \n 4.Exist \n Input Choice:";
                cin>>choice;
                system("cls");
                if(choice==1)
                {
```
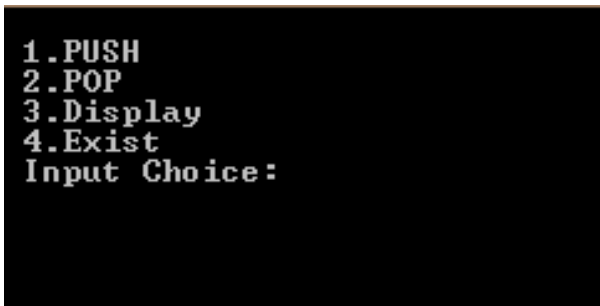
```
                              cout<<"\n Enter Value : " ;
                              cin>>value ;
                              push ( value );
                   }
                   if ( choice ==2)
                   {
                              pop ();
                   }
                   if ( choice ==3)
                   {
                              display ();
                   }
         } while ( choice !=4);
         cout<<"\n\n\n Existing .....!! \ n" ;
         return  0;
}
```


Figure : 1 Output

3   Conclusion

A stack is a container of objects that are inserted and removed according
to the last-in first-out (LIFO) principle. In the pushdown stacks only two
operations are allowed: push the item into the stack, and pop the item out
of the stack. A stack is a limited access data structure - elements can be
added and removed from the stack only at the top. push adds an item to
the top of the stack, pop removes the item from the top. A helpful analogy
is to think of a stack of books; you can remove only the top book, also you
can add a new book on the top.

4