Introduction to Data Structure and Algorithm in C++ (**STACKS*****)***

Activities for this lab:

- Explain the concepts of Stack
- ***Stack creation and implementation using Array and LL***
- ***lab exercise***

STACKS

OVERVIEW
A stack is a last in, first out (LIFO) data structure. Items are removed from a stack in the reverse order from the way they were inserted

AIM:

- To perform all stacks operation:
  - push(object): inserts an element
  - object pop(): removes and returns the last inserted element
  - object top(): returns the last inserted element without removing it
  - integer size(): returns the number of elements stored
  - boolean isEmpty(): indicates whether no elements are stored
  - Use both Array and Linked list for implementation

ALGORITHM Steps:

For Array Based, Declare and initialize necessary variables, eg top = -1, MAXSIZE etc.

Initialy **top = -1;**

1. **isEmpty** returns boolean **(top == -1).**
2. For push operation, operation increases top by one and writes pushed element to **stack[top];**

```
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, checks that top is not equal to -1 and decreases top variable by 1;

```
      If top = -1
        print "Stack underflow"
      Else
        item = stack[top]
        top = top - 1
        Display item
```

5. For next pop operation, goto step 4.

6. **peek/Top** operation checks that top is not equal to -1 and returns **storage[top];**

7. stop

Lab Exercise

Modify the code given in the lab session, split the code into stack.cpp, stack.h and main.cpp to create a Menu that enable user to enter option for stack operations

- Push()
- Pop()
- Top()
- isEmpty() and isFull()
- sizeofStack()
- display()
- exit

Note:
Modify Push() function so that it will enable you to push more items using (Y/N): E.g

Enter your Choice: 1
Push item to stack: 12
Enter more(y/n): y
Push item to stack: 15
Enter more(y/n): n

Modify Pop() function so that it will confirm user want to delete item or not,(give warning that item deleted cannot be recovered) using (Delete/Cancel): E.g (D/C)

Example of INPUT / OUTPUT

Enter the operation to be performed: 1) push 2) pop 3) display 4) size 5) exit

1
Enter the number to be pushed: 11
Enter the operation to be performed: 1) push 2) pop 3) display 4) size 5) exit
1
Enter the number to be pushed: 22
Enter the operation to be performed: 1) push 2) pop 3) display 4) size 5) exit
1
Enter the number to be pushed: 33
Enter the operation to be performed: 1) push 2) pop 3) display 4) size 5) exit
2
The number Popped is : 33
Enter the operation to be performed: 1) push 2) pop 3) display 4) size 5) exit
3
The stack is 11 22
Enter the operation to be performed: 1) push 2) pop 3) display 4) size 5) exit
4

The size of the stack is 2

RESULT:

All the stack operations are performed using the switch case (An Array based stack)