Introduction to stack

Stack is a non-primitive linear data structure that stores different data items temporarily and performs different operations on those data items on the LIFO (Last-In-first-Out) sequence.

Elements are added and remove from the same end i.e top of the stack. As all the insertions and deletions operation are done through the top of the stack, the last added items will be firstly deleted so called LIFO type of list.

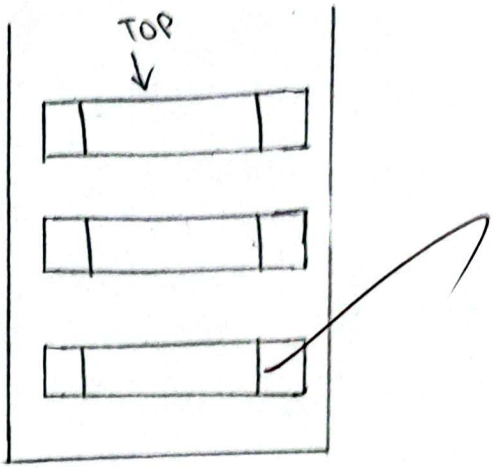

Fig:- Computer stack

Terminologies

① **MAXSIZE:-**

It is the total capacity of a stack.

② **TOP:-**

It is an external pointer of a stack that is used to point to the topmost element of stack.

③ **Push:-**

It is the process of insertion of the new data items into a stack.

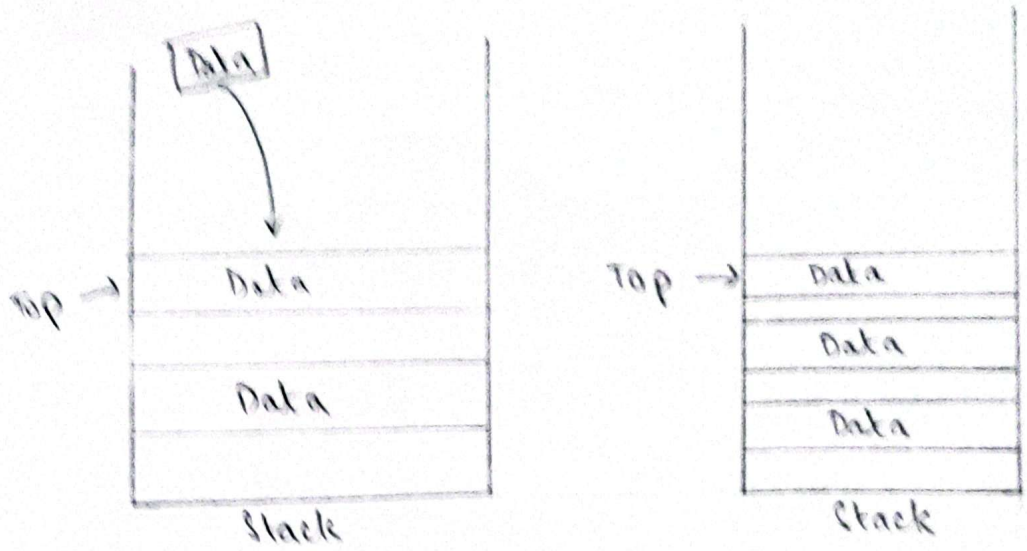

Fig:- Push operation

Algorithm for Push operation:-

Let MAXSIZE be the maximum size of a stack and Top be a stack pointer used to point the top most element of the stack.

1) Check the overflow condition of the stack.
   If top = MAXSIZE-1 then,
      print "Stack is full" & exit the program
   Else
      Increase top by 1 ie top = top+1
2) Read the element to be inserted as item.
3) Set stack [TOP] = item
4) Stop.

④ **Pop :-**

It is the process of deleting or removing of data items from a stack.

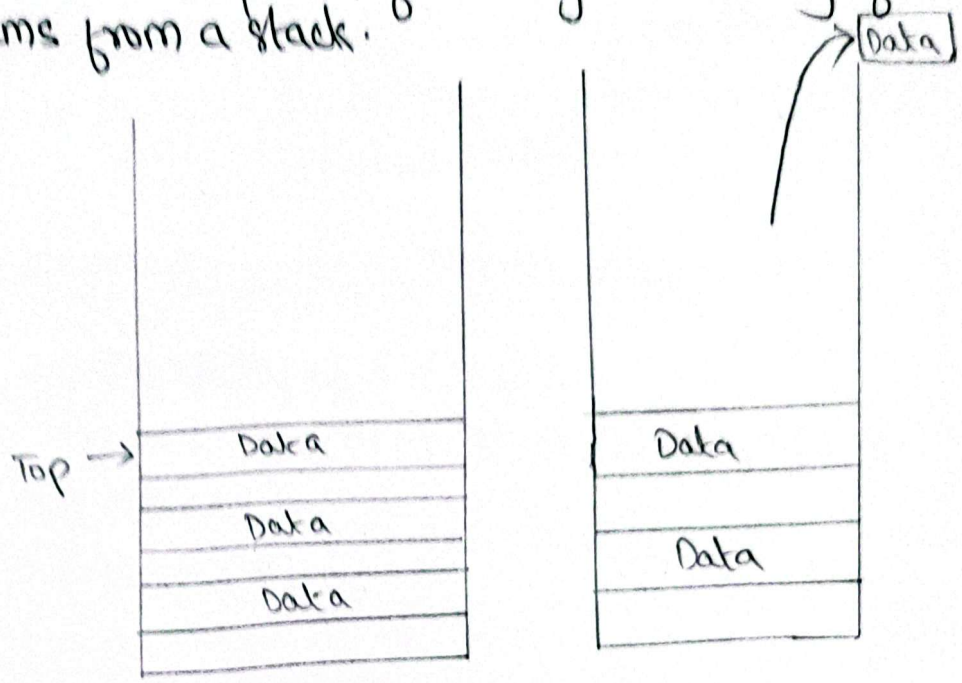

Fig :- Pop operations

Algorithm for Pop operation :-

Let MAXSIZE be the maximum size of a stack and top be a stack pointer used to point the top most element of the stack.

1) Check the stack underflow condition as

```
If top == -1 then
   Print "Stack is empty" & Exit.
Else
   Set item = Stack [Top]
   Decrement Top by 1 as Top = Top-1
```

2) Print the item as deleted data element.

3) Stop

⑤ **Full of the Stack :-**

If the top points MAXSIZE-1 in the index of a stack, then it is called full of the stack. No push operations can be performed after the full of the stack and is called overflow condition

⑥ **Empty of the stack :-**

If Top points to -1 (ie less than 0) index of the stack then it is called empty of the stack. If we attempt to perform Pop operation further is an empty stack it is called underflow condition.

Displaying data items : (Algorithm)

1) Check for the empty of the stack
   If Top == -1 then print "stack is empty" & exit.
2) for (i = Top; i >= 0; i--)
   Print Stack [i]
3) Stop