ADT Stack: Definition and Operations

ADT Stack

- What is a stack ?
  - A ***stack*** is a special kind of list in which all insertions and deletions take place at one end, called the top.
  - The stack is also known as:
    1) ***pushdown list*** or
    2) ***LIFO*** (Last-in-first-out) ***list***

chrisp 12

Operations: PUSH, POP, && TOP

1. **PUSH(x,S)**
   - inserts the element x at the top of the stack
2. **POP(S)**
   - deletes the top element of the Stack S.
3. **TOP(S)**
   - returns the element at the top of the Stack S

chrisp 14

Utility Operations:

1. **initializeStack(). Initializes the stack to be empty.**
2. **isEmpty(). Returns TRUE if the stack is empty; otherwise returns FALSE.**
3. **isFull(). Returns TRUE if the stack is full; otherwise returns FALSE.**

Implementations:

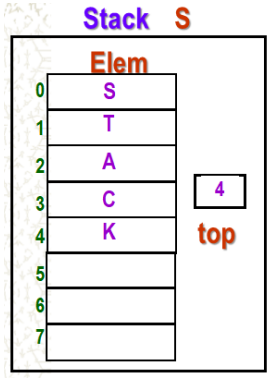
1) **Array Implementation [4 versions]**
2) **Linked list Implementation**
3) **Cursor-base Implementation**

Caution about the stack:

- **Only the top element can be accessed, hence the stack cannot be traversed.**
- **If there is a need to access all the elements, pop the top element so that next element will become top element. Repeat the process until each element has the chance to be top element.**

Array Implementation of Stack

View 1: Stack will grow from 0 to max-1

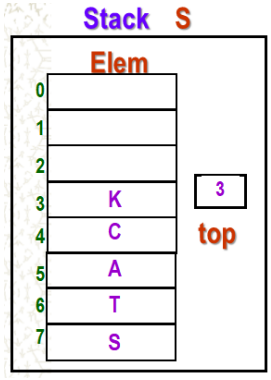


//Illustration of a stack with 5 elements

Scenarios:

1. Empty Stack → top is -1
2. Full Stack → top is max - 1
3. PUSH(x,S) → top increases
4. POP(S) → top decreases

View 2: Stack will grow from max-1 to 0

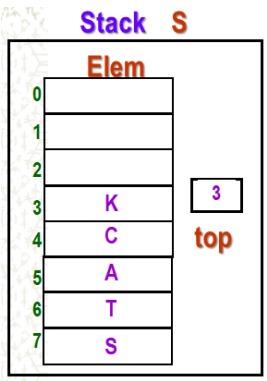


//Illustration of a stack with 5 elements

Scenarios:

1. Empty Stack → top is max
2. Full Stack → top is 0
3. PUSH(x,S) → top decreases
4. POP(S) → top increases

Practice Exercise:



1) Write an appropriate definition of data type Stack. Include the macro MAX representing the size of the stack.
2) Write the code of the following stack Operations using the definition in #1 and a assuming that the stack will grow from max-1 to 0.
    a) **initializeStack()**
    b) **PUSH();**
    c) **POP();**
    d) **TOP()**
    e) **isEmpty()**
    f) **isFull()**