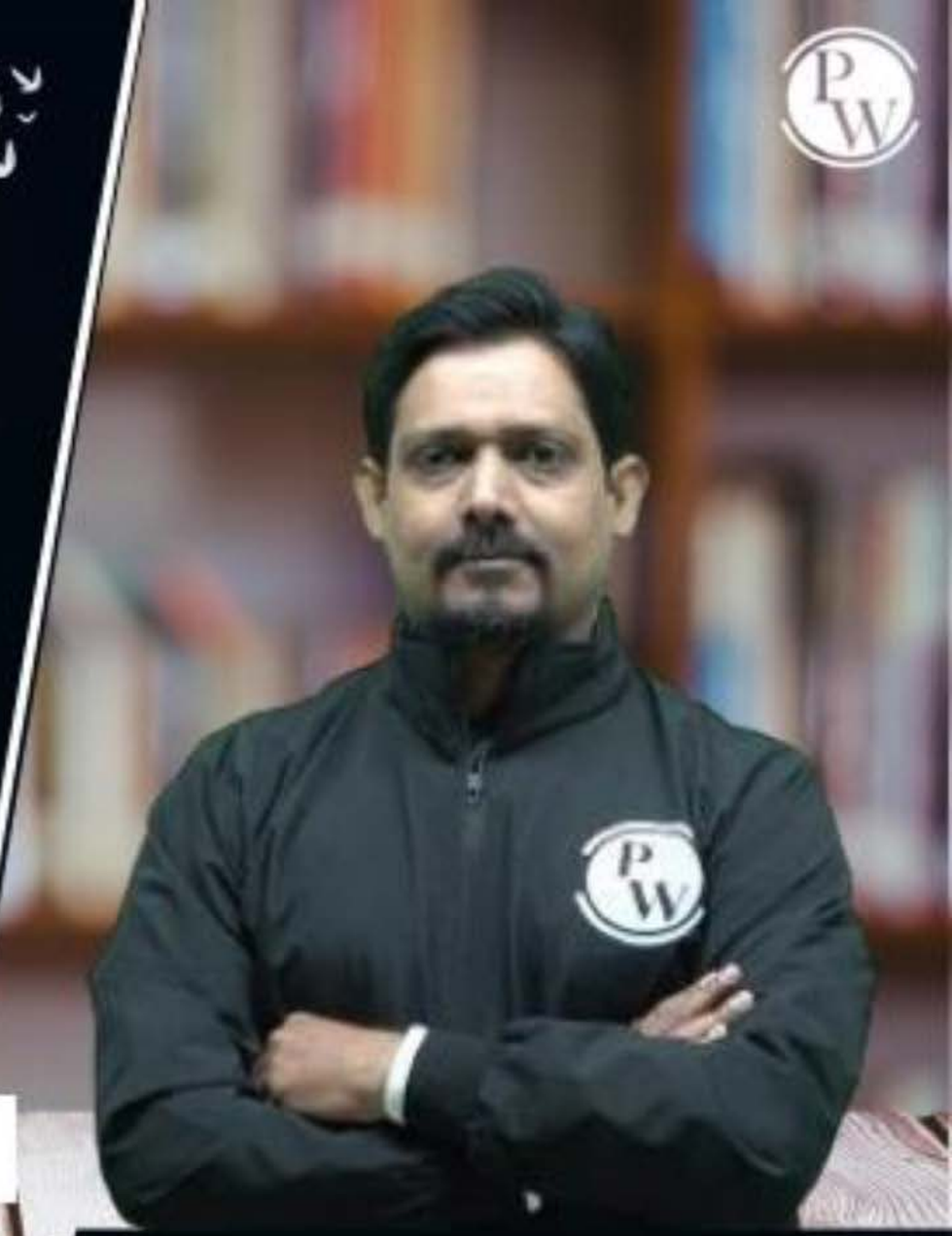
Computer Science & IT

Data Structure & Programming

Stack

Lecture No. 01



By- Abhishek Sir

Recap of Previous Lecture

- Topic: 3-D matrix
- Topic: Tridiagonal Matrix
- Topic: Practice Problem
- Topic
- Topic

Row major order

Stack

Slide

Topics to be Covered

- Topic — Tridiagnol Matrix column major
- Topic — stack
- Topic — opeoaham on stack
- Topic — Application
- Topic

Slide

Topic : Question

9×9

#Q Consider the LTM a[-4..4][0...8] Non zero elements are stored in row major order. What is the address of A[0][3]. Base address is 1000 and size of each element is 2B.

Lower bound 1

$$BA + \left[\frac{i(i-1)}{2} + j - 1\right] \times Size$$

a[-4..4][0..8]

+5 +1

a[1..9][1..9]

A[0][3]

+5 +1

A[5][4]

$$1000 + \left[\frac{5 \times 4}{2} + 3\right] \times 2 = 1000 + 13 \times 2 = 1026$$

Slide

Topic : Question

a_{11}	a_{21}	a_{12}	a_{22}	a_{32}	a_{23}	a_{33}	a_{43}	a_{34}	a_{44}

a[i][j]

#Q Assume that Tridiagonal Matrix of order $(n \times n)$is mopped into a one dimensional array using column major order, then calculate the location of an element a(i,j)(L0 is base address and size is 1 Byte)?

(A) $L_0+(2j+i-3)$

(B) $L_0+(2i+j-3)$

(C) $L_0+(2i-j-3)$

(D) $L_0+(2j+i+3)$

(A)

$$\begin{bmatrix} a_{11} & a_{12} & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 \\ 0 & a_{32} & a_{33} & a_{34} \\ 0 & 0 & a_{43} & a_{44} \end{bmatrix}$$

Analysis :

i=2 j=3 Ans - 0

i=3, j=3 Ans- 1

i=4, j=3 Ans = 2

i - j + 1

Slide

We are in j^{th} column, No. of columns completed is $(j-1)$

$(j>1)$ No. of elements : $3(j-1)-1$ ← first column Not working

on j^{th} column

No. of elements arranged $i-j+1$

By default it will be zero

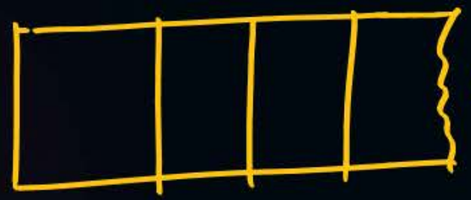
$$Lo + [3(j-1) - 1 + i - j + 1] \times 1$$

$$Lo + [3j - 3 + i - j] = \boxed{Lo + [2j + i - 3]}$$

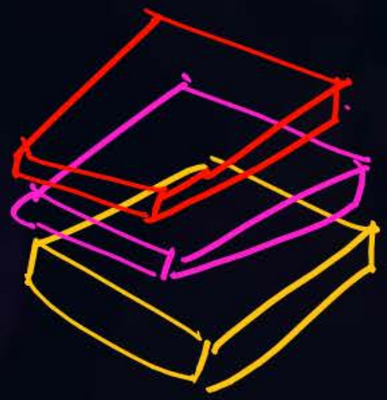
Topic : Stack as LIFO

1. Stack is a Linear Data structure

2. Stack is "one ended data structure"



Basic operation (push, pop)



Stack of books

Slide

Topic : Stack as LIFO

3. Stack based on logical property LIFO (Last in first out)

4. Inserting element in stack push operation

5. Deleting element from stack called pop operation

6. To Identify in which position or location push & pop will be done, one variable is maintained called top.

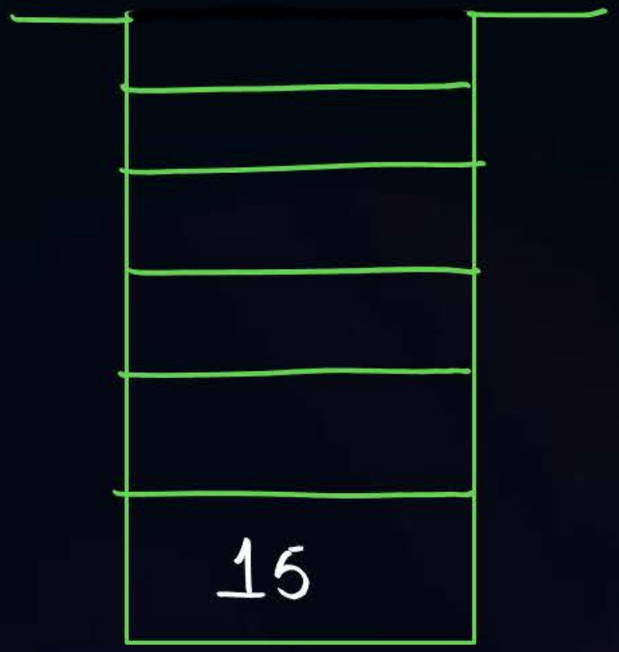
Slide

Topic : Stack as LIFO

CO → In computer there is a special purpose Register called as stack pointer points to top of (Address) Run-time stack.

Slide

Topic : Stack as LIFO

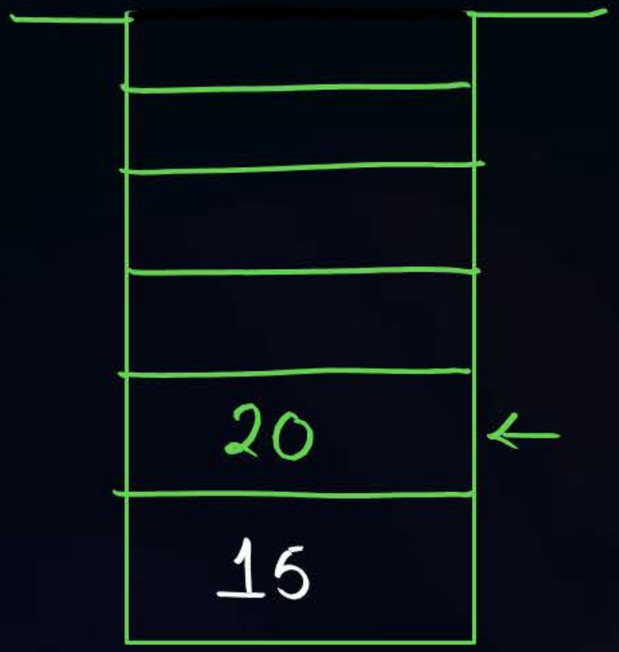


push / pop

push(15)

Slide

Topic : Stack as LIFO

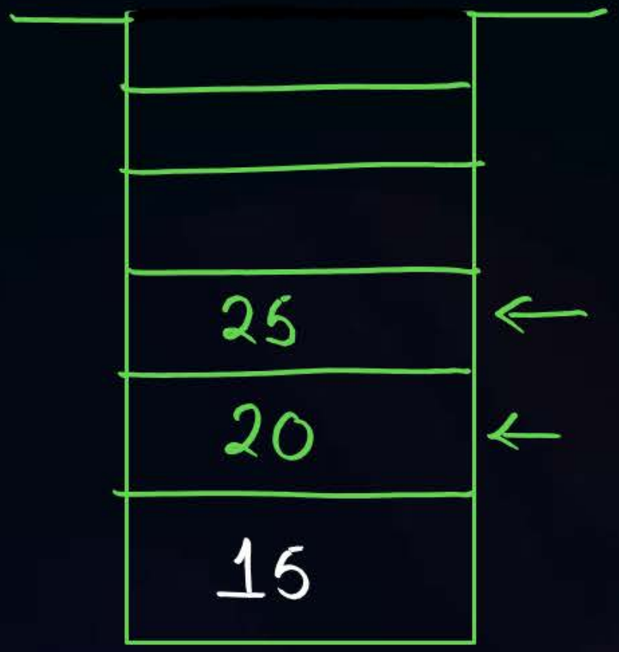


push / pop

push(15), push(20)

Slide

Topic : Stack as LIFO

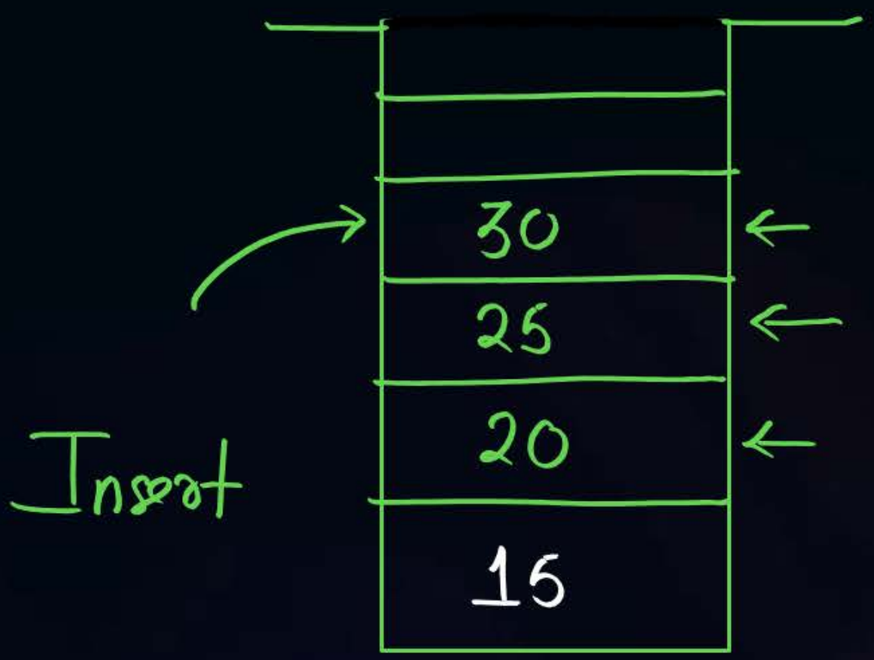


push / pop

push(15), push(20), push(25)

Slide

Topic : Stack as LIFO

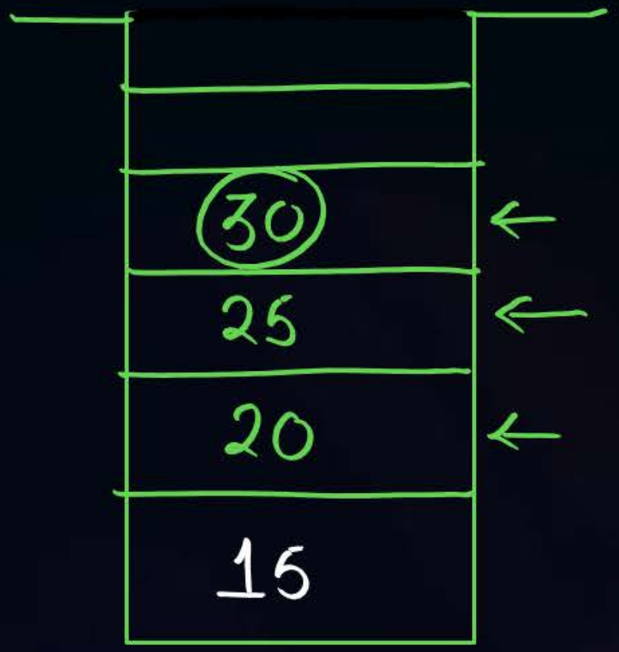


push / pop

push(15), push(20), push(25), push(30)

push will happen on top of 30

Topic : Stack as LIFO



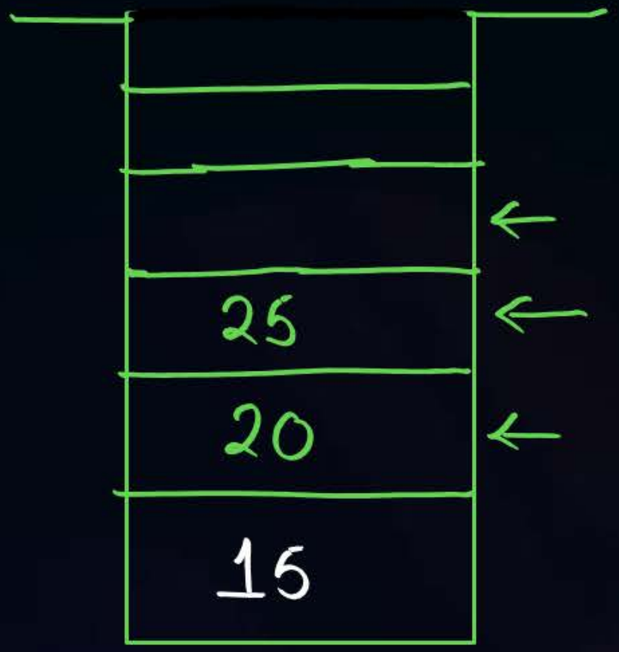
push / pop

push(15), push(20), push(25), push(30)

pop() will return 30

Slide

Topic : Stack as LIFO



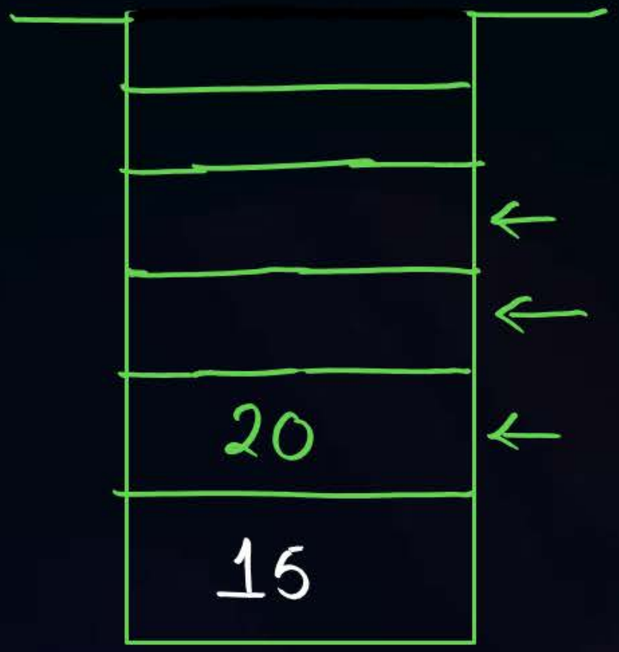
push / pop

push(15), push(20), push(25), push(30)

pop() will return 25

Slide

Topic : Stack as LIFO



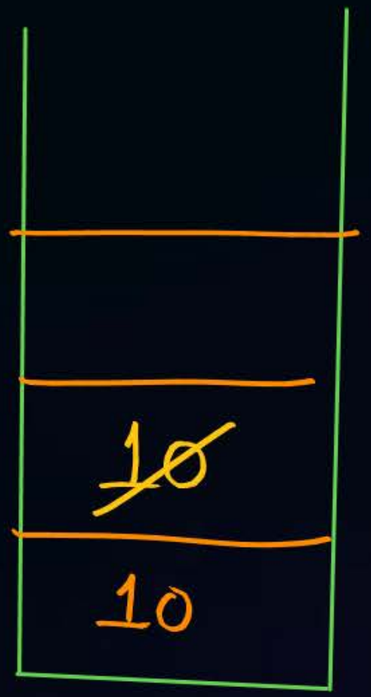
push / pop

push(15), push(20), push(25), push(30)

Slide

Topic : Stack as LIFO

Q if following operation performed on stack



push(10), push(20), push(10), 10 pop(), X = pop()

X = 20

push(40), 40 Y = pop(), push(10) 10 pop(), 10 Z = pop()

then X+Y+Z is 20+40+10 ?

Slide

Topic : Stack as LIFO

Array Implementation of stack

global variable

Macro

```
#define Max 100
int a[Max];
int top;
```

preprocessor : Wherever Max will will come in program replaced by 100;

Slide

Topic : Stack as LIFO

Array Implementation of stack

global variable

```
# define Max 100
int a[Max];
int top;
```

Intialization of stack

Initialize the top to -1

```
void init() {
        top= -1;
}
```

Slide

Topic : Stack as LIFO

IsEmpty(): IsEmpty function returns true (1) if stack is empty

Otherwise false(0)

```
int IsEmpty() {
        if(top == -1)
                return 1;
        else
                return 0;
        }
```

Slide

Topic : Stack as LIFO

int a[max]; a[0...99]

Push operation

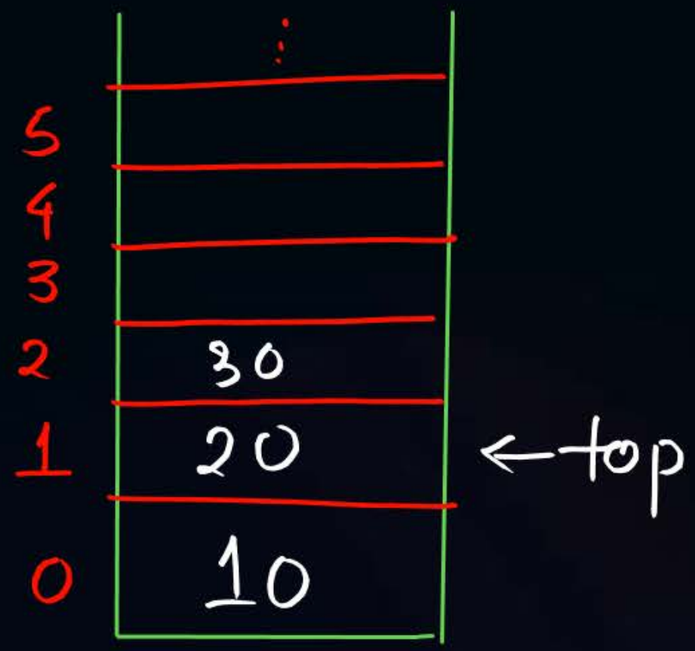
#define Max 100

if array limit is reached and new element is inserted results in stack overflow

```
void push( int data) {
    if( top ==Max-1) {
        printf("stack overflow");
        return; // terminate the program
    }
    top= top+1; // Increment the
    a[top] = data; // top & insert the
                   // value
}
```

Slide

Topic : Stack as LIFO



top = -1, ✓

push(10)

```
int main(){
  init();      // top=-1
  push(10); // top+1 = 0
  push(20); // top+1 = 1
  push(30); // top+1 = 2
  printf("%d", pop()); //
}
```

30 top=1

Slide

Topic : Stack as LIFO

if stack is empty

2 we try to

delete the element

then its result in

underflow.

pop removes top element of stack and return it if its Nonempty.

Slide

Topic : Stack as LIFO

if stack is empty

2 we try to

delete the element

then its result in

underflow.

```
int pop() {
    int data;
    if (IsEmpty()) {
        printf("Stack is empty");
        return -1;
    }
    data = a[top]; // take data from
    top = top - 1;  // stack, decrement the top
    return data;
}
```

Slide

2 mins Summary

Topic — Tridiagonal Column major order

Topic — Stack

Topic — Stack Implementation

Topic

Topic

Slide

THANK - YOU