Monotonic Data Structures ---

Concepts & Qns

by . codestorywithMIK

∞ Instagram → codestorywithmiK

X → CSwithMIK

WhatsApp → codestorywithMIK

Motivation :- Until when, you will run away from hard work. Until when you will skip tough topics. One day they will all come to you and you will have to face them.

So, better to stop running

start facing...

Video - 2

What is Monotonic Data Structure

Montonic Stack

What does it even mean by 'monotonic' in Data Structures ???

{1, 2, 3, 4}

In mathematics, a sequence is called

- Monotonic increasing if each element in the sequence is greater or equal to the one before it.

- Monotonic decreasing if each element is less or equal to the one before it.

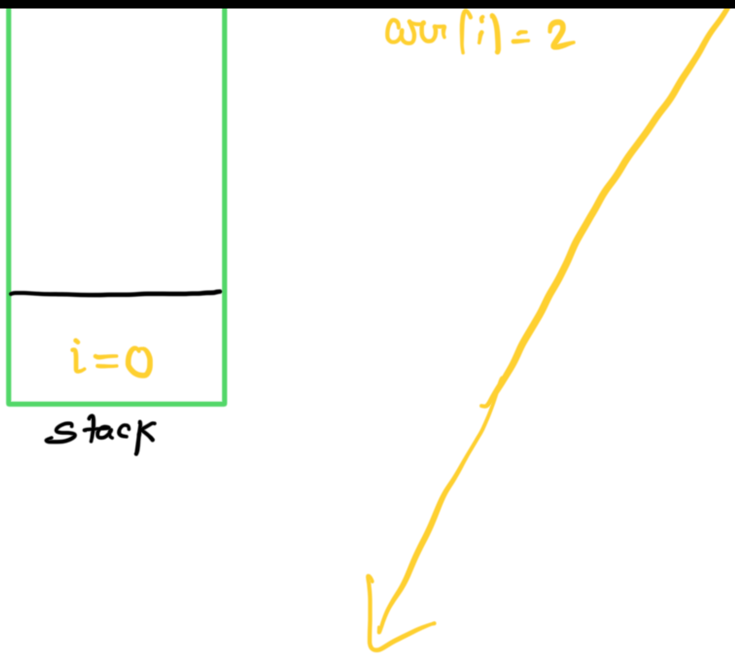
"We often use this idea in stacks and queues to achieve our goal"

Monotonic Stack...

A stack that always remains in increasing (monotonic increasing) or decreasing (monotonic decreasing) as elements are pushed or popped from stack...

Montonic Increasing stack ...

$$arr = \{\overset{0}{4}, \underset{i}{\overset{1}{2}}, \overset{2}{5}, \overset{3}{1}, \overset{4}{3}\}$$



```
Stack<int> St ; //stores index of the element

for (int i= 0;  i<n ;  i++){

        while(!St.empty() && array[St.top()] > arr[i]){
                  St.pop(); //we want increasing
        }

        St.push[i]; // maintains increasing order
}
```

: Template :

How does this even help us??

let's suppose you are asked to find NSEL (Next Smaller Element to Left)

$$\begin{array}{ccccccc} & 0 & 1 & 2 & 3 & 4 & \\ arr = \{ & 4, & 2, & 5, & 1, & 3 & \} \\ & & & & & i & \end{array}$$

result = {-1, -1, 2, -1, 1}

Brute Force:-

```
for ( i = 0;  i<n;  i++) { n
    for (j = i-1 ;  j>=0 ;  j--) n
        if (arr[j] < arr[i]) ;
            break ;
```

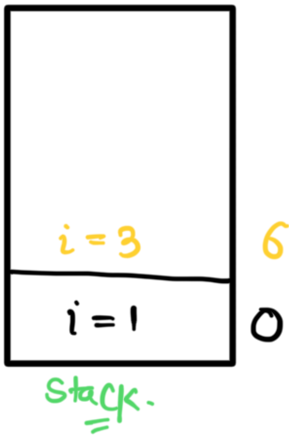
$T.C = \boxed{O(n^2)}$ T.L.E.

$S.C = O(1)$

Optimal - using monotonic stack ...

$\{\overset{0}{4}, \overset{1}{0}, \overset{2}{7}, \underset{i}{\overset{3}{6}}, \dots\dots\dots\} \Leftarrow$

$\{-1, -1, 0, 0$



(·) Why ✓

(·) NSEL ...

(Monotonic Inc. Stack)

T.C = pushed once
popped once

$O(2*n) \simeq O(n)$

S.C = O(n)

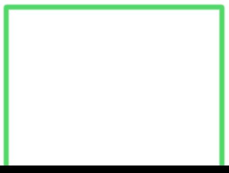
(·) NSER (Next Smaller Element to the Right)

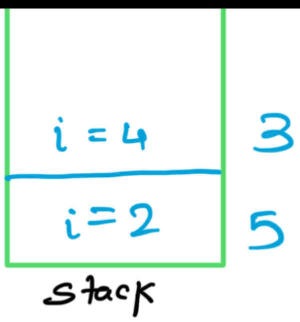
O(n).

for(i = n-1; i>=0; i--)

Monotonic Decreasing Stack :-

arr = $\{\overset{0}{4}, \overset{1}{2}, \overset{2}{5}, \overset{3}{1}, \underset{i}{\overset{4}{3}}\}$





```
Stack<int> st ; // stores index of the element

for ( int i= 0;  i<n ;  i++) {

      while ( ! st.empty() &&  array[st.top()] < arr[i]) {
                  st.pop() ; //we want decreasing
      }

      st.push[i] ; // maintains decreasing order
}
```

: Template :

How does this even help us??

let's suppose you are asked to find

NGEL (Next Greater Element to Left)

(NGER)

$$arr = \{\overset{0}{4}, \overset{1}{2}, \overset{2}{5}, \overset{3}{1}, \overset{4}{3}\}$$

$$result = \{-1, 4, -1, 5, 5\}$$

Brute Force

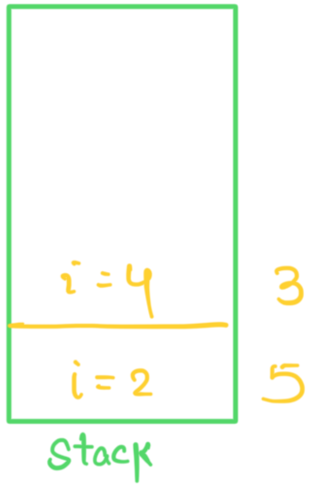
```
for(i=0; i<n; i++){
    for(j=i-1; j>=0; j--){
        arr[j] > arr[i]
        break
```

$T.C = O(n^2)$

$S.C = O(1)$

Optimal - using monotonic stack ...

$$arr = \{\overset{0}{4}, \overset{1}{2}, \overset{2}{5}, \overset{3}{1}, \overset{4}{3}\}$$



result = {-1, 4, -1, 5, 5}

$T.C = O(2n) \sim O(n)$

$S.C = O(n)$

$\left.\begin{array}{l}\text{NGEL}\\ \text{NGER}\end{array}\right\}$ Monotonic Decreasing stack

$\left.\begin{array}{l}\text{NSEL}\\ \text{NSER}\end{array}\right\}$ Monotonic Increasing Stack.

Remember...

Stacks → Any Problem where Next Greater/Smaller to left or right

Greater → Monotonic Decreasing

Smaller → Monotonic Increasing

Leetcode 739 ✓

Leetcode 84 ✓

Leetcode 503 ✓

Leetcode 901 ✓

Leetcode 42 ✓

NSEL

NSER

NGEL

NGER

etc. etc. etc.