

## Today's Content

- First +ve missing Integer ✓
- Subarray Basics ✓
- Max Subarray Sum ✓
- Water logging → { Will discuss in Adv3 }

Q8) Given  $\underline{N}$  array elements find front  $\underline{1 \text{ to } N}$  Integers.

$\boxed{1 \rightarrow N}$   
missing

$$\underline{\text{Ex1: }} \text{arr}[5] = 3 -2 \boxed{1} \boxed{2} 7 : 4$$

$$\underline{\text{Ex2: }} \text{arr}[7] = -9 \overbrace{2 \quad 6 \quad 4 \quad -8}^{\longrightarrow} \boxed{1} \boxed{3} : 5$$

$$\underline{\text{Quiz1: }} \text{arr}[8] = -2 \quad 4 \quad -1 \quad -6 \quad 3 \quad 7 \quad 8 \quad 4 \quad -3 : 1$$

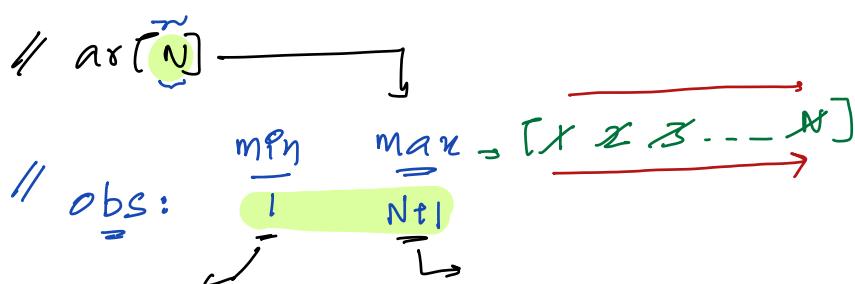
$$\underline{\text{Quiz2: }} \text{arr}[6] = \boxed{1} \quad 0 \quad -5 \quad -6 \quad 4 \quad \boxed{2} : 3$$

$$\underline{\text{Quiz3: }} \text{arr}[6] = \boxed{1} \quad \boxed{2} \quad \boxed{5} \quad \boxed{6} \quad \boxed{4} \quad \boxed{3} : \boxed{73}$$

$$\underline{\text{Ex1: }} \text{arr}[5] = -4 \quad 8 \quad 3 \quad -1 \quad 0 : 1$$

$$\underline{\text{Ex2: }} \text{arr}[4] = 4 \quad 2 \quad 1 \quad 3 : 5$$

$$\underline{\text{Ex3: }} \text{arr}[4] = -3 \quad -6 \quad -3 \quad -10 : 1 \checkmark$$



ans range  $\rightarrow [1 \quad N+1]$

// Sol1:

```
elc = 1; dc <= N; dc++ {
```

If elc is not present in  
arr[], dc is your ans  
return dc

}

TC:  $O(N+1) \cdot O(N)$

SC:  $O(1)$

```
elc = 1; dc <= N; dc++ {
```

If elc is not present in  
arr[], dc is your ans  
return dc

}

return N+1

TC:  $O(N) \cdot O(N)$

SC:  $O(1)$

We can optimize  
by storing entries  
in Set.

// Sol2: HashSet<int> hs;  
Insert all elements in hs

$\Gamma_1 \Rightarrow N$

```
elc = 1; dc <= N; dc++ {
```

If elc is not present  
in hs, return dc

}

return N+1

TC:  $O(N+N) = O(N)$

SC:  $O(N)$

Condition → You cannot use any inbuilt Library

Ex:  $\text{arr}[8] =$  -3 14 1 0 3 2 9 4

$[1 - 8]$

$\text{bool}[8] =$  T T T T F

$\text{ans} = 5$

Ex:  $\text{arr}[7] =$  3 -2 6 8 1 2 T

$[1 - 7]$

$\text{bool}[7] =$  T T T 0

$\text{ans} = 4$

Pseudo Code

$\Rightarrow O(N)$

```
bool ch[N];  
P = 0; P < N; P++) {
```

ele = arr[P]

if ( $1 \leq \text{ele} \leq 9$   
 $\text{ele} \leq N$ ) {

ch[ele - 1] = T

P = 0; P < N; P++) {

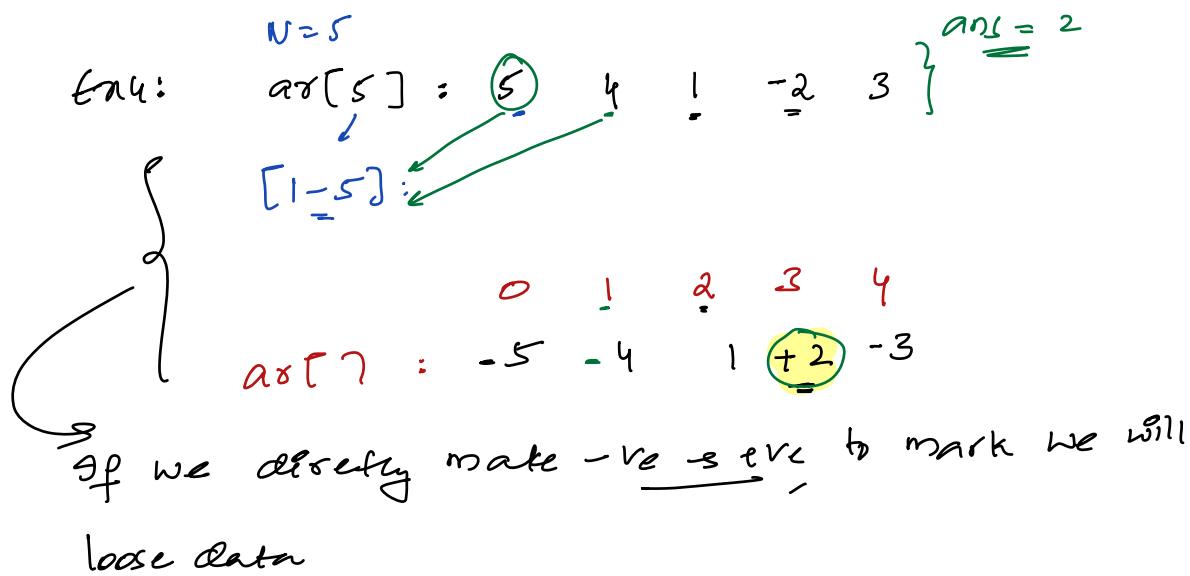
if ( $\text{ch}[P] = F$ ) {

return P + 1

return N + 1

Condition 2 : No Extra Space / No Libraries

bool  $[N] \Rightarrow$  ? To check if element present or not  
arr  $[N]$   $\downarrow$  Somehow let's try to store present  
or not data in same array.



→ Because of -ve numbers above logic fails.

→ iterate & delete -ve numbers?

$$\text{Exn: } \underset{\downarrow}{\underline{N=5}} = \text{arr}[5] : \quad \begin{matrix} 5 & 4 & 1 & -2 & 3 \end{matrix}$$

$$\text{ans} \rightarrow \underset{\downarrow}{\underline{[1 \ 6]}} :$$

Replace all  $x=0$

with  $\_?$

$$\begin{matrix} 0 & 1 & 2 & 3 & 4 \\ 5 & 4 & 1 & \textcolor{red}{7} & 3 \end{matrix}$$

$$\text{arr[]} : \quad \begin{matrix} 0 & \textcircled{1} & 2 & 3 & 4 \\ -5 & 4 & -1 & -7 & \textcircled{-3} \\ \hline \text{ans} = 3 \end{matrix}$$

$$\text{Exn: } \underset{\downarrow}{\underline{N=8}} = \text{arr}[8] : \quad \begin{matrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 3 & -6 & 4 & -4 & 1 & -8 & 9 & 2 \end{matrix}$$

$x=0$  with  $= 8+2$

$$\begin{matrix} 0 & 1 & 2 & \textcircled{3} & \textcircled{4} & \textcircled{5} & \textcircled{6} & 7 \\ -3 & -10 & -4 & -10 & 1 & 10 & 9 & 2 \\ \hline \text{ans} = 5 \end{matrix}$$

$\text{arr}[0] = 1$

### Pseudocode

1) Step1: Iterate in array,  $\text{ar}[i] \neq 0$   $\text{ar}[9] = N+2 \swarrow$

replace

$\text{ar}[9] = N+2 \swarrow$

2) Step2: Modify the array.  $T_C: O(N+N_1N)$

$S_C: O(1)$

$i = 0; i < N; i++ \{$

$\underline{\text{ele}} = \text{abs}(\text{ar}[i])$

$\underline{\text{if }} 1 \leq \underline{\text{ele}} \text{ and } \underline{\text{ele}} \leq N$

$\underline{\text{ind}} = \underline{\text{ele}} - 1;$

$\text{ar}[\underline{\text{ind}}] = -1 * \text{abs}[\text{ar}[\underline{\text{ind}}]]$

}

3) Step3: Iterate in array & get first the negative Integer

$(i = 0; i < N; i++) \{ \text{if } (\text{ar}[i] > 0) \text{ return } i + 1 \}$

return  $N + 1$

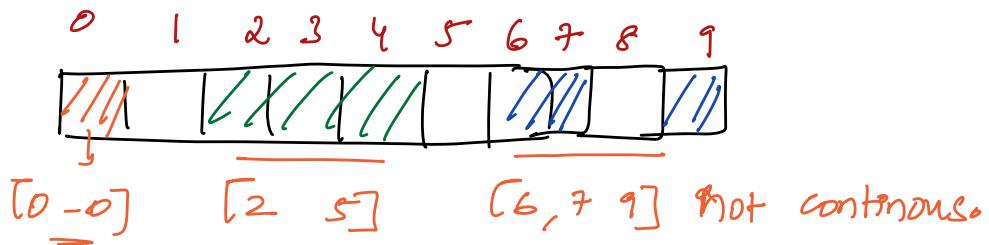
$\text{ar}[7] :$	-2	1	3	2	6	3	8
$\underline{N=7}$	0	1	2	3	4	5	6
	-9	-1	+3	2	6	-3	8

ans = 3

10-45PP

// 28) Given N array elements, Find Plan Subarray Sum

// Subarray: Continuous part of array.



Ex:  $\text{ar}[7] = -3 \quad 4 \quad 6 \quad 8 \quad -10 \quad 2 \quad 7$   
 $\qquad\qquad\qquad \text{ans} = 18$

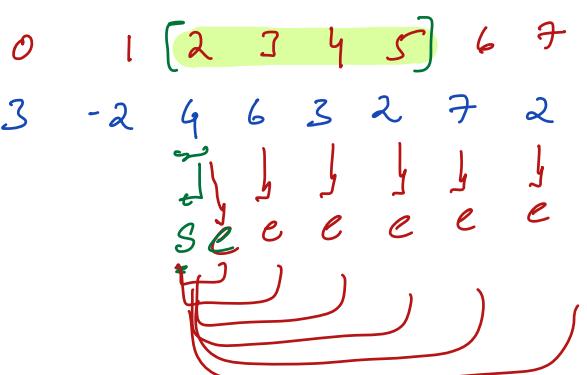
$\text{ar}[5] = \{-3, -2, -8, -1, -10\}$   
 $\qquad\qquad\qquad \text{ans} = -1$

// Generate every subarray  
to get of sum.

//  $s : c$   
 $= 2 \quad 5$

Ex:  $\text{ar}[8] = 3 \quad -2 \quad 4 \quad 6 \quad 3 \quad 2 \quad 7 \quad 2$

//  $s \leftarrow e$



// In general }

manSum = INT\_MIN

$s = 0; s < N; s++ \{$

$e = s; e < N; e++ \{$

// [s e], subarray found

$sum = 0$

$i = s; i < e; i++ \{$

$sum = sum + arr[i]$

$manSum = max(manSum, sum)$

TC:  $O(N^3)$

SC:  $O(1)$

$arr[]: \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8$   
 $\quad \quad \quad 3 \quad 2 \quad -1 \quad 6 \quad 4 \quad 8 \quad 3 \quad -2 \quad 7$

$ans = INT\_MIN$

$i = 0; i < N; i++ \{$

$sum = 0$

$j = i; j < N; j++ \{$

$sum = sum + arr[j]$

$ans = max(ans, sum)$

// All subarray sums starting at index i

$\Rightarrow TC: O(N^2)$

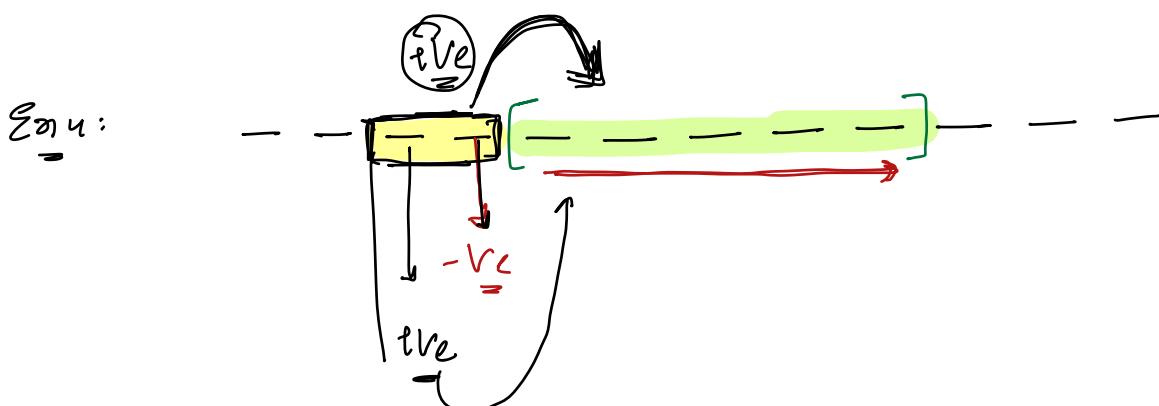
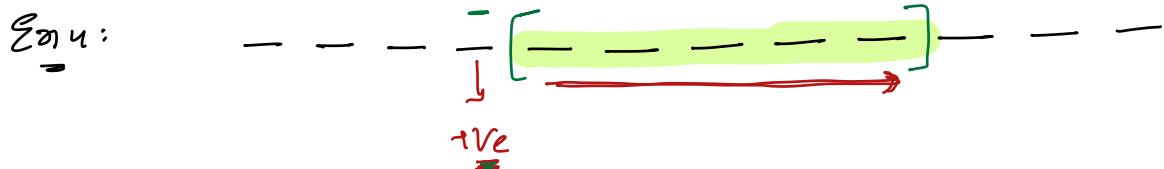
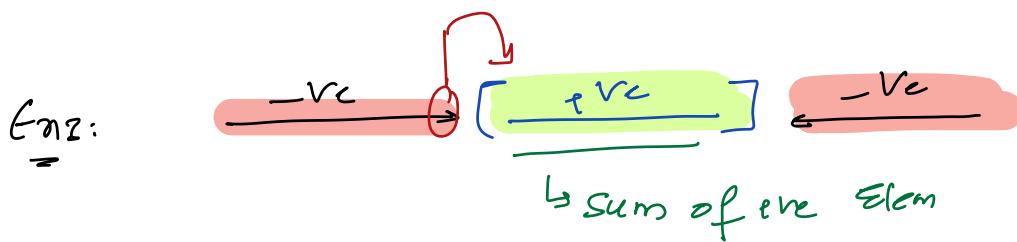
$\Rightarrow SC: O(1)$

// obs:

if all are +ve

$$\underline{\underline{E_{\text{q1}}}}: \quad 3 \quad 5 \quad 6 \quad 8 \quad +2 \quad \text{ans: sum of all Elements}$$

$$\underline{\underline{E_{\text{q2}}}}: \quad -8 \quad -2 \quad -9 \quad -11 \quad -10 \quad \begin{matrix} \text{if all -ve:} \\ \text{ans: man of arr} \end{matrix}$$



-	<u>5</u>	6	7	-3	2	-10	-12	8	12	21	-4	7
Sum = 0	+ <u>5</u>	11	<u>18</u>	<u>15</u>	<u>17</u>	<u>7</u>	<u>-8</u>	<u>8</u>	<u>20</u>	<u>41</u>	<u>37</u>	<u>44</u>
ans =	+5	11	18	18	18	18	18	18	20	41	41	44

	10	-20	-12	6	5	-3	8	-2
Sum = 0	10	-20	-12	6	5	-3	8	14
ans = -∞	10	10	10	11	11	16	16	

	-10	-3	-2	-14	ans = INT_MIN
Sum = 0	-10	-3	-2	-14	Sum = 0;
ans = -∞	-10	-3	-2	-14	$P = 0; P_{\text{min}}, P_{\text{max}} \in \{$
					Sum = Sum + arr[P]
Kadane's:	$\left\{ \begin{array}{l} TC: O(N) \\ SC: O(1) \end{array} \right.$				$P = \max(Ans, Sum)$
					If (Sum < 0) Sum = 0
					return Ans;

Doubts

Diagram illustrating a swap operation in an array. The array  $ar[7]$  contains elements 0 through 7. A swap operation is performed between index 1 (value 1) and index 5 (value 5). The value at index 1 is moved to index 5, and the value at index 5 is moved to index 1. The original value 5 is stored in a variable  $temp$ .

The array after the swap is shown with indices 0 through 7 and values 0, 5, 2, 3, 1, 6, 4, 7.

$i = 0; i < N; i++ \rightarrow O(\underline{n})$

if  $arr[i] == arr[i+1]$

$\downarrow$  continue

$j = i;$

while  $arr[\underline{j}] == arr[j+1]$  且  $arr[j] \neq N$  且  $arr[j+1] \neq N$

$\downarrow$   $n = arr[j]$

$\downarrow$  swap  $arr[\underline{n-1}] \leftarrow arr[j]$

$j$

$\}$