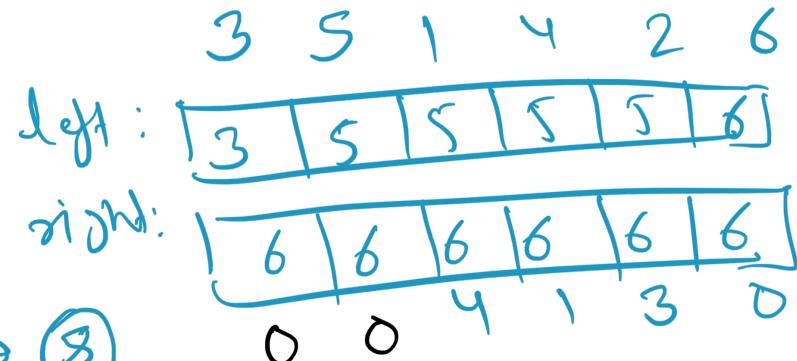
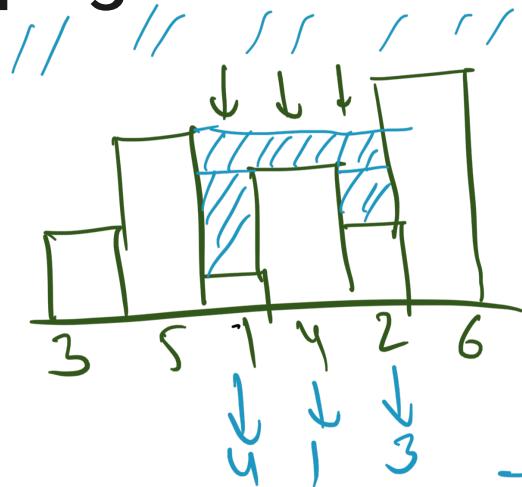


# Arrays - IV

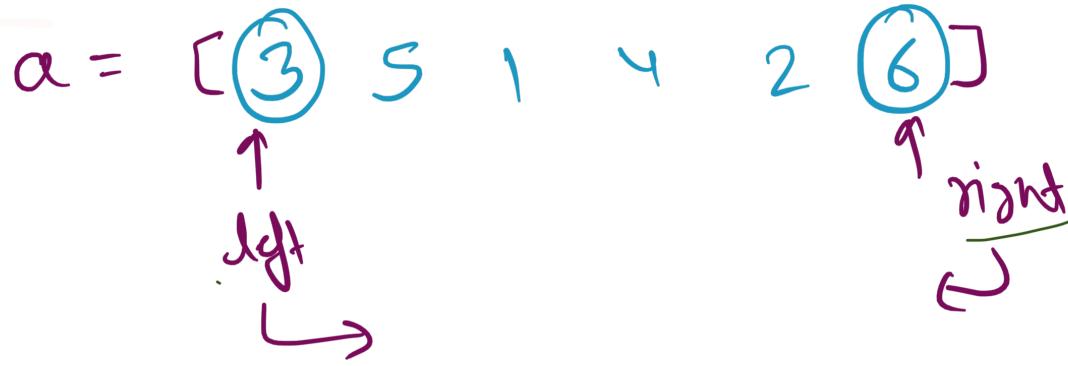


## Trapping Rainwater using O(1) Space Complexity



$\text{ans} += \min(\text{left}[i], \text{right}[i]) - a[i]$

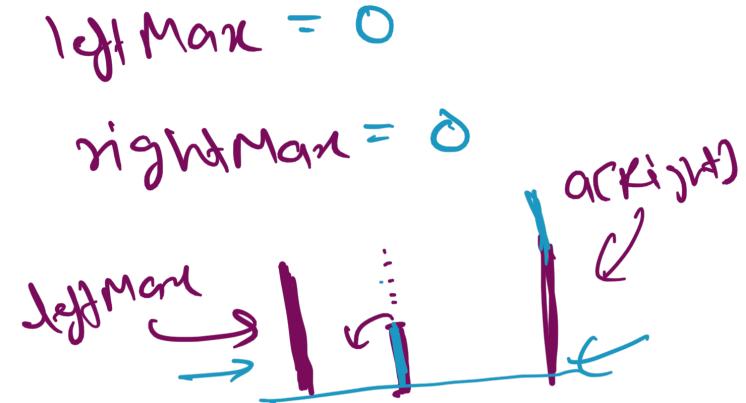
$O(N)$



```

int ans = 0
if ( a[left] <= a[right] ) { || a[right] > leftMax
    if ( a[left] > leftMax) leftMax = a[left];
    else ans += leftMax - a[left]
    left++;
}
else {
    if ( a[right] > rightMax) rightMax = a[right];
    else ans += rightMax - a[right];
    right--;
}

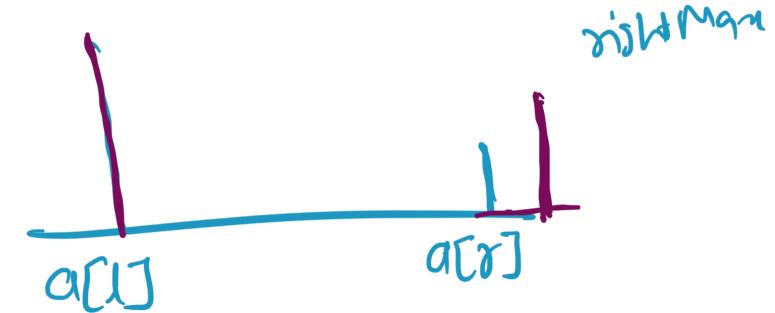
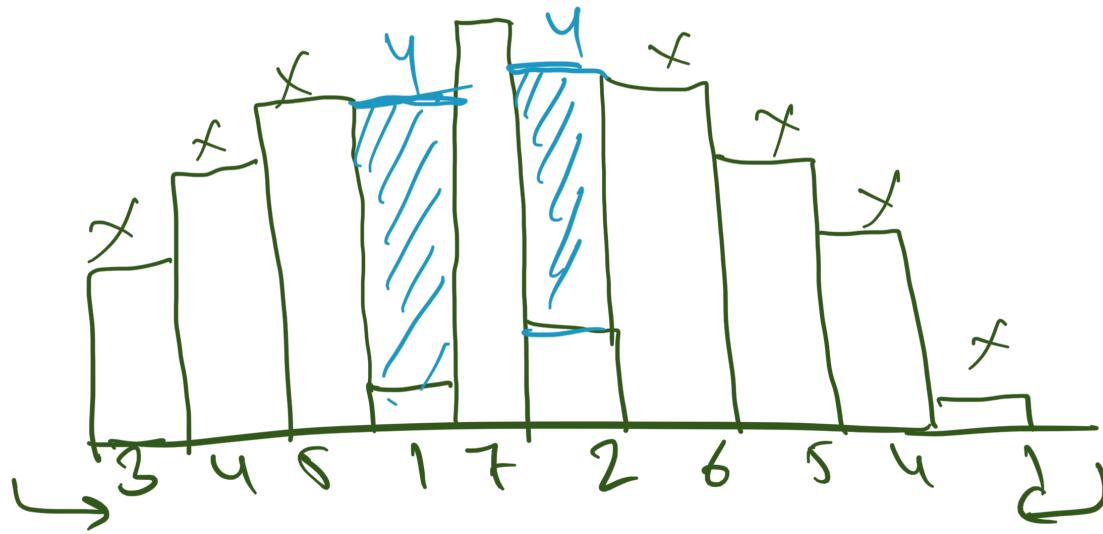
```



```
    }  
    return arr;
```

while ( $k < 8$ )

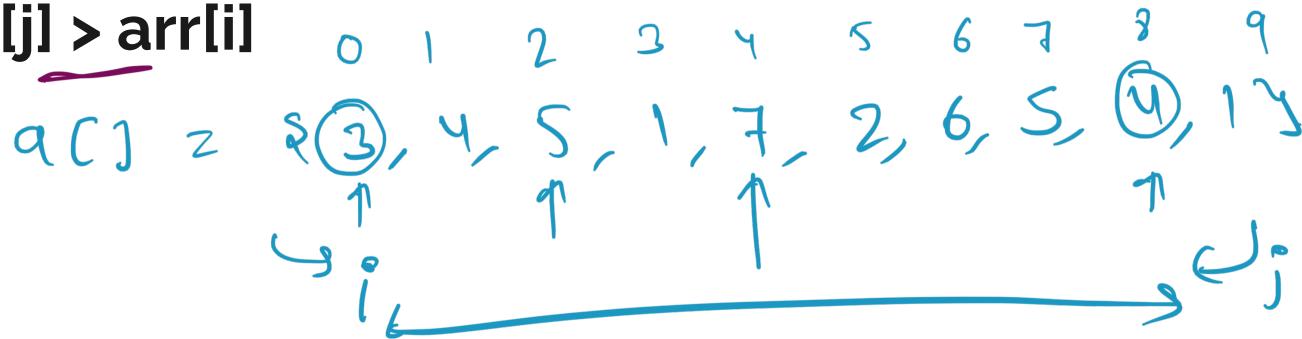
$$\text{ans} = \underline{\underline{4+1+3}} \\ = \underline{\underline{8}}.$$



3 5 1 4 2 6  
↑  
↑ σ  
l

$$\text{leftman} = \emptyset \quad 35$$
$$\text{rightman} = 0$$

Given an array  $\text{arr}[]$ , find the maximum  $j - i$  such that  $\underline{\text{arr}[j] > \text{arr}[i]}$



$$j - i = 8 - 0 = \underline{0}.$$

```

for( int i=0; i<n; i++) {
    for( j= i+1; j<n; j++) {
        if( a[j]>a[i])
            ans= max(ans, j-i);
}

```

y

0 1 2 3 4 5 6 7 8  
 $a[] = \{ \underline{5}, \underline{4}, \underline{1}, \underline{7}, 2, 8, \underline{6}, \underline{3}, \underline{4} \}$

$\text{rightMax} = \{ 8, 7, 7, 8, 8, 8, 6, 4, 4 \}_{\text{right}}$

$\text{leftMin} = \{ 5, 4, 1, 1, 1, 1, 1, 1 \}$

int ans=  $\emptyset$   
 $\times 2 \times 4 \times 6$

$$j-i = 6$$



## Find Two Numbers in a sorted Array with the Given Sum

$a[] = \{1, 2, 7, 8, 10, 12\}$   
 $\underline{\text{sum}} = \underline{10}$ .

$\rightarrow O(N^2)$

```
for (i=0 )  
  for (j=i+1)  
    currSum = a[i] + a[j]  
    print ('i + j');
```

$\rightarrow O(n)$

$$\begin{aligned}a[l] + a[r] \\ 1 + 12 &= 13 \\ 1 + 10 &\rightarrow 11 \\ 1 + 8 &\rightarrow 9 \\ 2 + 8 &\rightarrow 10.\end{aligned}$$

if ( $\text{currSum} > \text{sum}$ )  
 r--;  
else if ( $\text{currSum} < \text{sum}$ )  
 l++  
else  
 return true

<u>Unsorted</u>	Array	Time	Space
Brute		$O(N^2)$	$O(1)$
Set		$O(N)$	$O(N)$
Sort		$O(N \log N + N)$	$O(1)$
		$\downarrow$	
		<u><math>O(N \log N)</math></u>	

## 3-Sum

Find Three Numbers in an unsorted Array with the Given Sum

$$a[] = \{1, 2, 4, 7, 3, 5, -2, 1\}$$

$\leftarrow$  Sum = 9

```
for (i=0;  $\rightarrow$   
     for (j=i+1; ...  
          for (k=j+1; ...
```

$O(N^3)$

$3 \rightarrow \text{sum}$   
 $\downarrow$   
 $[2 \rightarrow \text{sum} \rightarrow k]$

for (  
  $\text{sum} - k$ )

$a, b, c \rightarrow x$

$$a+b+c = x$$

$$[a+b=k]$$

find  $c = x - k$

$$\begin{array}{r} a+b+c \\ x-c+c \\ \hline x \end{array}$$

For every  $c$   
find  $a+b$  such that  $\underline{a+b = x-c}$

## java. util. Arrays

Arrays.sort(a)  $\rightarrow O(N \log N)$

$a[] = \{1, 2, 4, 7, 3, 5, -2, 1\}$   
Sum = 9  
 $\rightarrow \{-2, \underline{\underline{1}}, \underline{2}, 3, \underline{4}, 5, \underline{7}\}$

$O(N \log N + N^2) \rightarrow \overline{O(N^2)}$

$$K = \text{sum} - a[i] \quad | \quad 9 - 1 = 8 \quad [\text{for } i \rightarrow N]$$
$$= 9 - (-2) = 11 \quad [\text{twoSum}() \rightarrow O(N)]$$

---

# Practice Problems

[https://practice.geeksforgeeks.org/explore?page=1&category\[\]=%5B%5D&sortBy=submissions](https://practice.geeksforgeeks.org/explore?page=1&category[]=%5B%5D&sortBy=submissions)

Moore' Voting  
Algorithm: