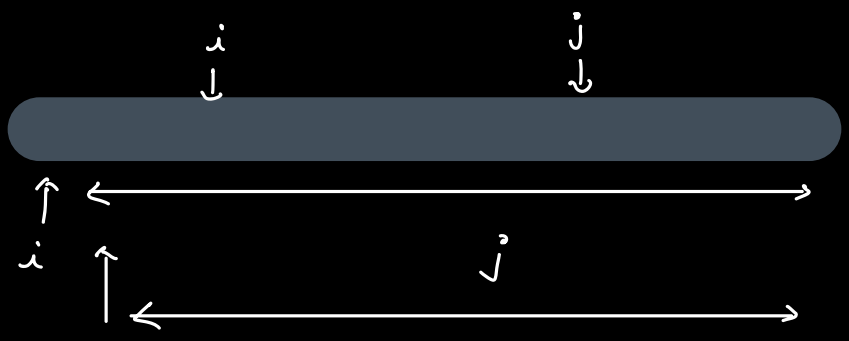
Q

	Brut Force	HashMap / Set
TC:	$O(N^2)$	$O(N)$
SC:	$O(1)$	$O(N)$



$A[i] + A[j] > K$

$A[i] + A[j] < K$

$A[i] + A[j] > K$

$j--;$

$A[i] + A[j] < K$

$i++$

```
boolean twoSum (int[] A, K) {
    int i = 0 , j = N-1;
    while ( i < j) {
        if( A[i] + A[j] == K){
            ret true;
        }
        if ( A[i] + A[j] > K) {
            j--;
        }
        else {
            i++;
        }
    }
    ret false;
}
```

TC: O(N)

SC: O(1)

Q Diff K

Facebook Given a sorted array.

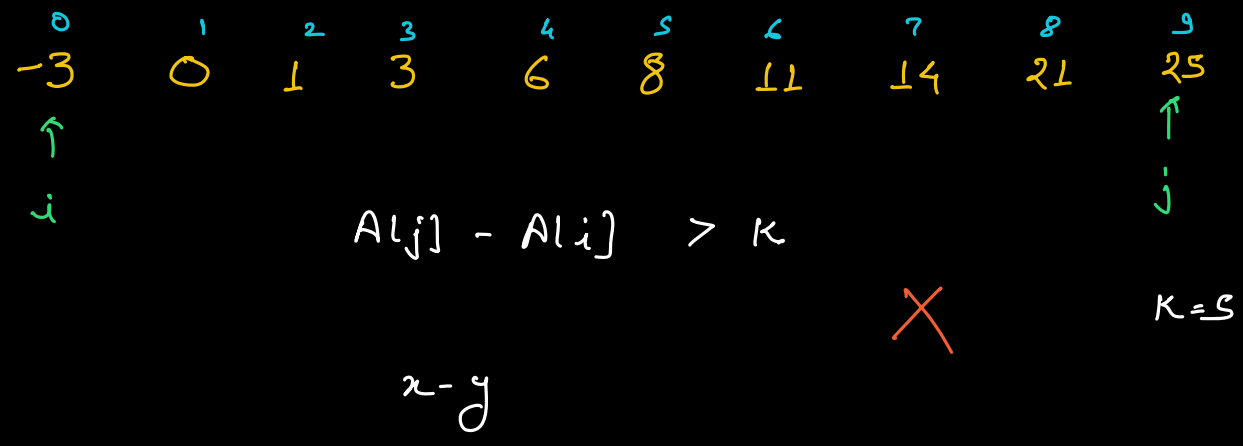
i, j

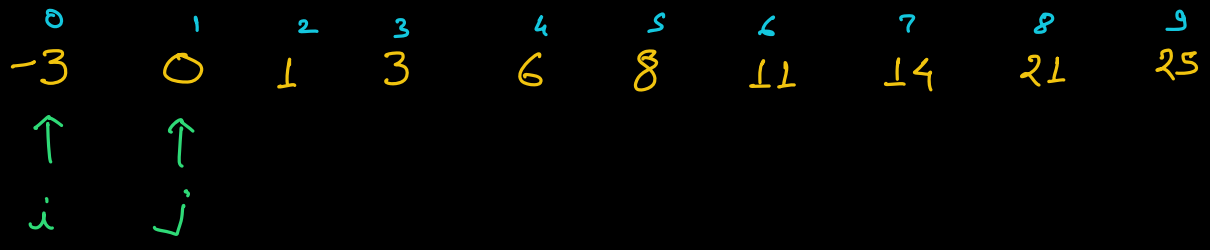
such that

$A[i] - A[j] == K \quad (K >= 0)$

$i \; != \; j$

0	1	2	3	4	5	6	7	8	9	
-3	0	1	3	6	8	11	14	21	25	K=5






```
boolean diffK (int[] A, K) {
    int i = 0, j = 1;
    while ( j < N)
        if ( A[j] - A[i] == K)
            ret true;
    }
    if (A[j] - A[i] > K) {
        i++;
        if( i == j) j++;
    }
    else {
        j++;
    }
}
ret false;
}
```

TC : O(N)

SC : O(1) (Brute)

→ Count no of pair

Q Given 3 sorted arrays A, B, & C
Find i, j & k such that

max (A[i], B[j], C[k]) - min (A[i], B[j], C[k])
is minimized.

	0	1	2	3	
A :	3	14	16	(23)	→ a
B :	-6	(23)	24	30	→ b
C :	-15	15	(26)	31	→ c

i	j	k	max (A[i], B[j], C[k])	min (A[i], B[j], C[k])	diff
0	0	0	3	-15	18 ←
0	0	1	15	-6	21
0	0	2	26	-6	32
⋮			⋮	⋮	⋮
1	2	3	31	14	1
3	1	2	26	23	3

	0	1	2	3
A :	3	14	16	(23)
B :	-6	(23)	24	30
C :	-15	15	(26)	3

max(A[i], B[j], C[k]) - min(A[i], B[j], C[k])
↓ ↓
x - y

	0	1	2	3
A :	3	14	16	23
B :	-6	23	24	30
C :	-15	15	26	31

max - min
X

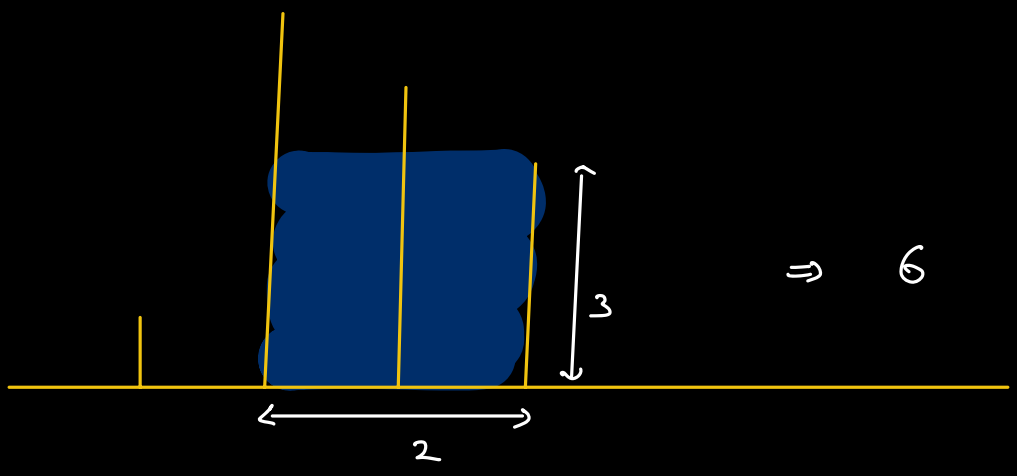
i	j	k	max	min	diff
0	0	0	3	-15	18
0	0	1	15	-6	21
0	1	1	23	3	20
1	1	1	23	14	9
2	1	1	23	15	8
2	1	2	26	16	10
3	1	2	26	23	3 ←
3	2	2	26	23	3

TC : O(a+b+c)

SC : O(1)

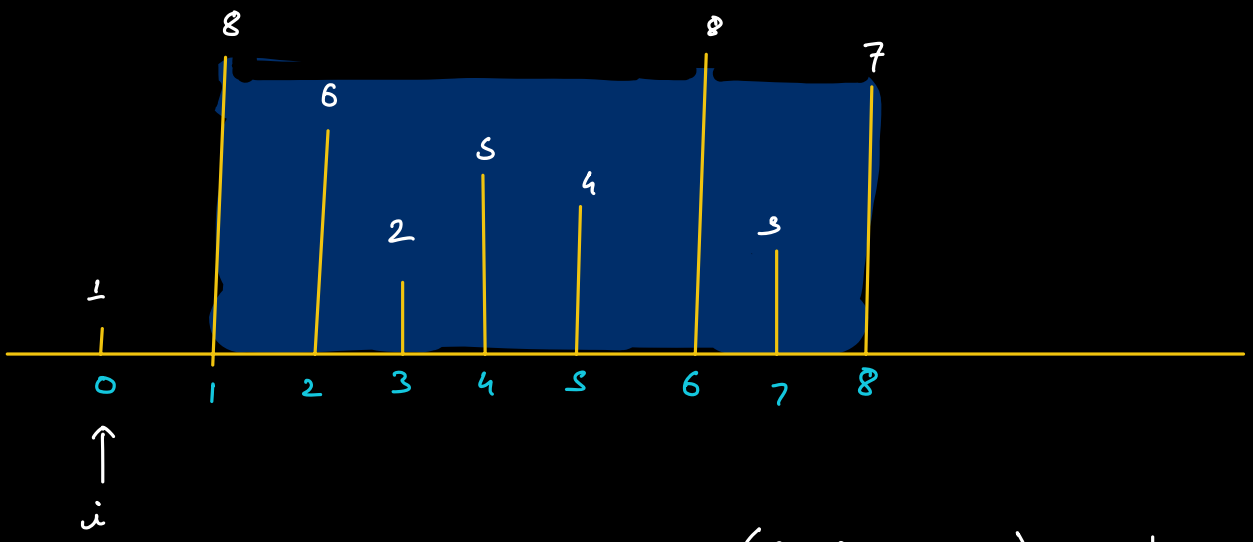
Q Container with most water

A : 1 5 4 3



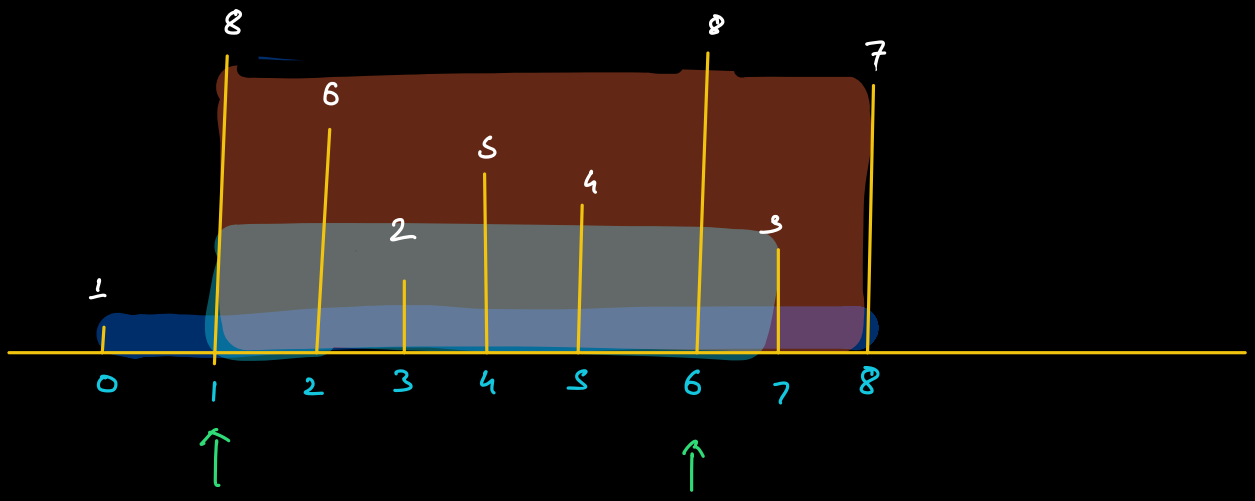
$\Rightarrow$ 6

A : 1 8 6 2 5 4 8 3 7



$$\min(A[j], A[i]) \times |j - i|$$

ht $\times$ wt




```
i = 0 ;
j = N-1;
while ( i < j ) {
    ans = max(ans, min(A[i], A[j]) * (j-i));
    if( A[i] < A[j]) i++;
    else j--;
}
ret ans;
```

TC: O(N)

SC: O(1)
(Extra)