- **Two Pointer :-**

i) Triplet Sum :- LeetCode - 15

Array must be sorted.

```
a :   -3   -1   0   1   2
      ↑    ↑             ↑
      F    S             E
```

F → Fixed : S → Start E → End.

- In sorted array, if we have to increase value then left++

  Sum ↑ → left++ ;

  Similarly

  Sum ↓ → End-- ;

- $x + y + z = 0 \longrightarrow y + z = -x$ → Approach.

  as it is triplet sum ∴ Outer loop will be $i < n-2$.

ii) Triplet Sum Closet :- LeetCode - 16.

```
a :   -3   -1   0   1   2   5 ;  closet = 1
      ↑    ↑             ↑
      F    S             E
```

- Let diff = $10^9 + 7$;
- Now iterating i.e; Outer Loop $i < n$
- S = i+1 ; E = n-1 ;
- if (Math.abs(diff) > Math.abs(sum - target))
  - diff = sum - target
  - res = Sum ;
- if (sum < target) S++ ;
- else E-- ;

Output :-

(-3) + (0) + (+5) = (2) > closet → 1.

iii) Duplicates in O(N) : O(1) ⇒ Leetcode 442

Condition :- $1 <= A[i] <= N$ → Range

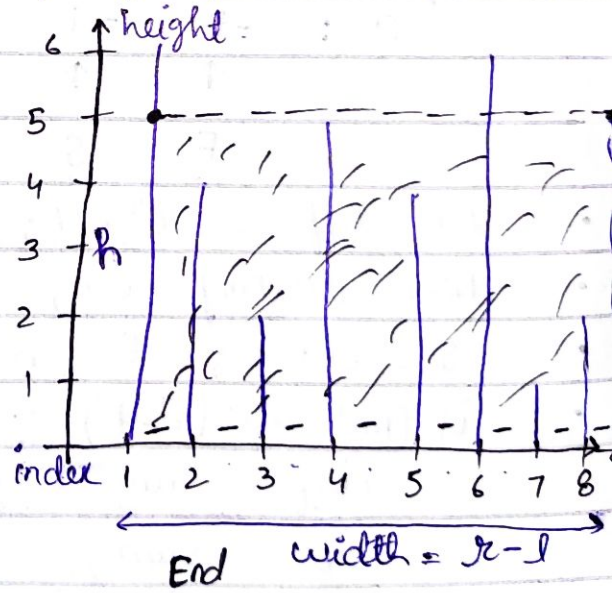
E → element
V → visited.

a:	4	3	2	-7	8	2	3	1
	0	1	2	3	4	5	6	7

- Check element at i^{th} index → A[0] = 4
- mark visited at element - 1 ∴ 4 - 1 ⇒ ③ array[3] ⇒ -7
- element = abs(A[i]);
- if (array[element - 1] > 0) -1 * array[element - 1]
- else res.push(element).

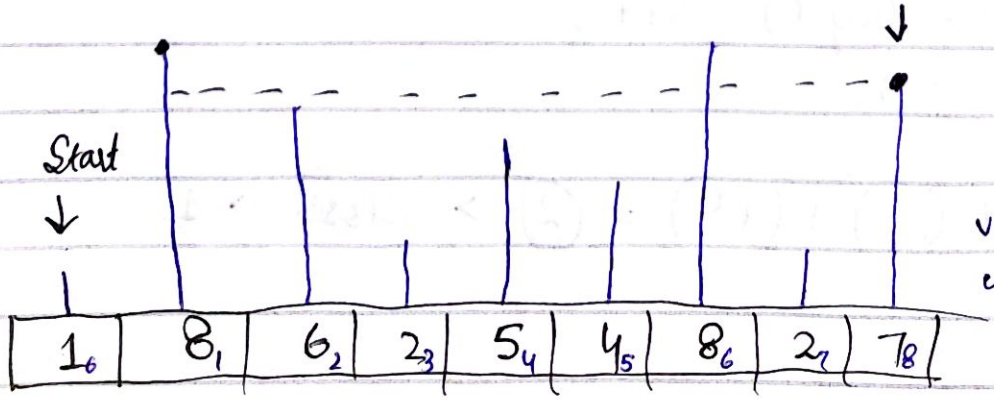
iv) Container with most water :- Leetcode 11

a: 6_1 4_2 2_3 5_4 4_5 6_6 1_7 3_8 5_9



maxArea = minheight * width
minheight = min(l, r)
width = r - l
∴ l = left ; r = right.

Conditions :-
if (l < r) l++ ;
else r-- ;



Pair → index
value → Height.
width → End - start.