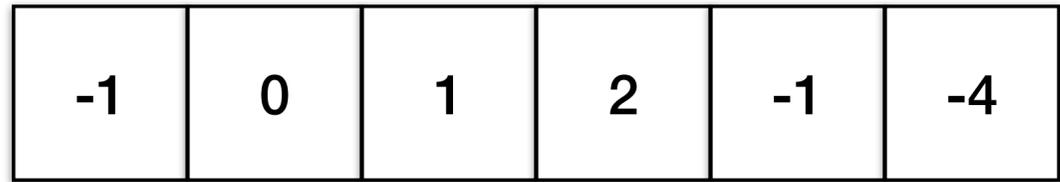


3 Sum - Test Case -1

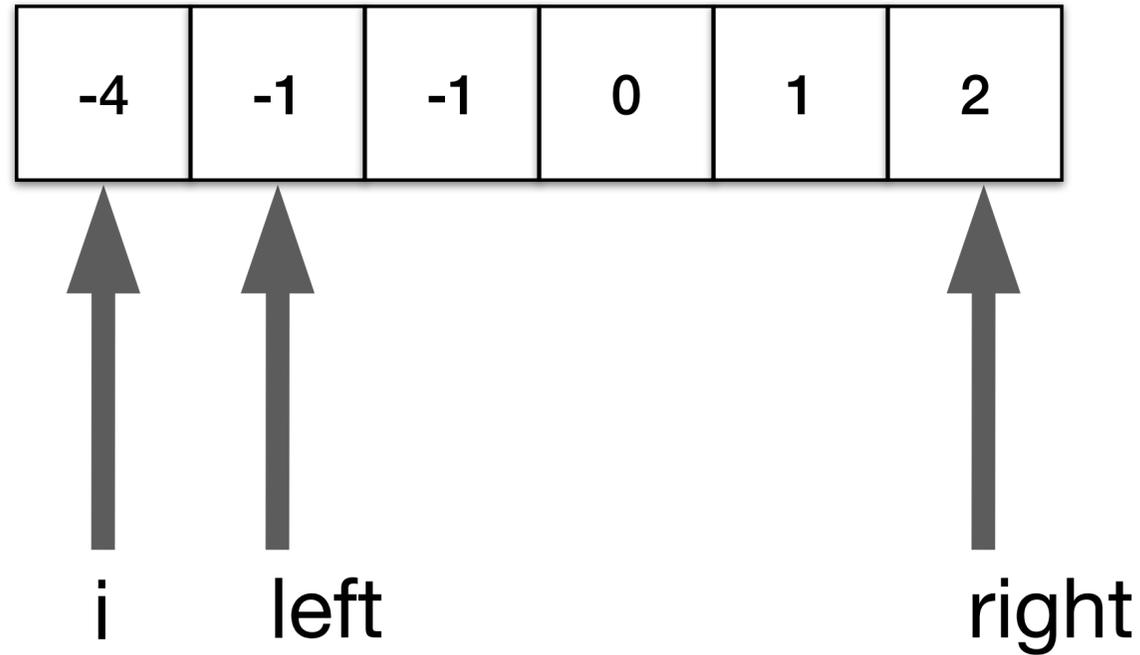


left



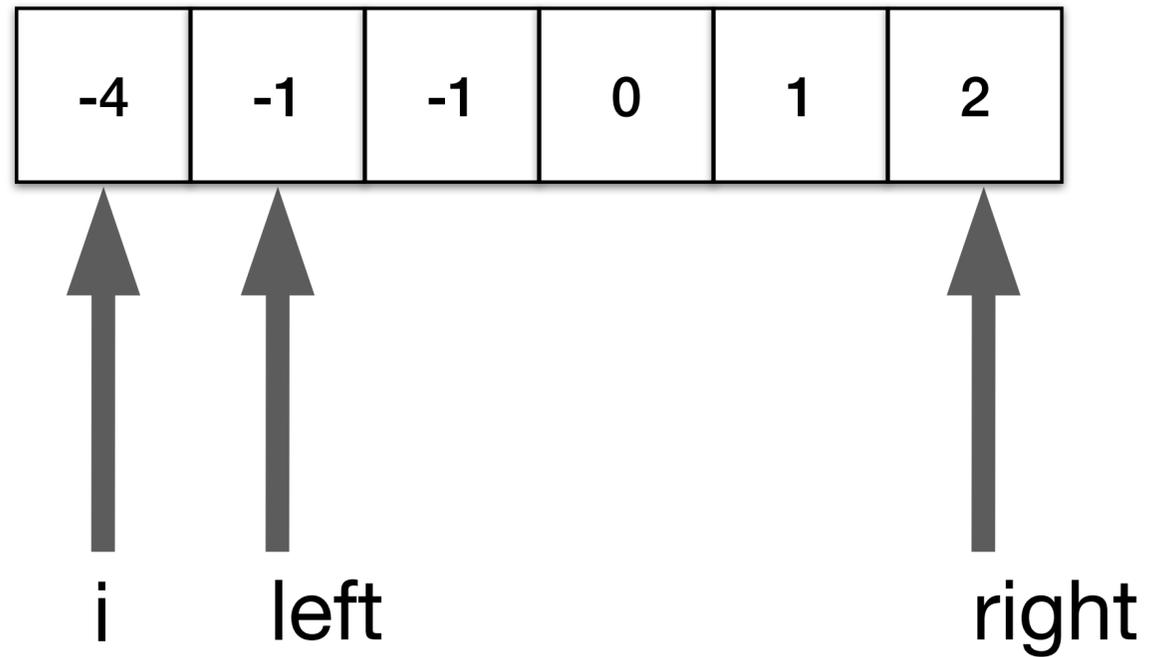
right

3 Sum - Test Case -1



Template :- Along with left and right pointers, We need to use one more pointer i to solve the problem and need to sort the array to get the maximum sum.

3 Sum - Test Case -1



$$\text{CurrentSum} = -4 + (-1) + 2 = -3$$

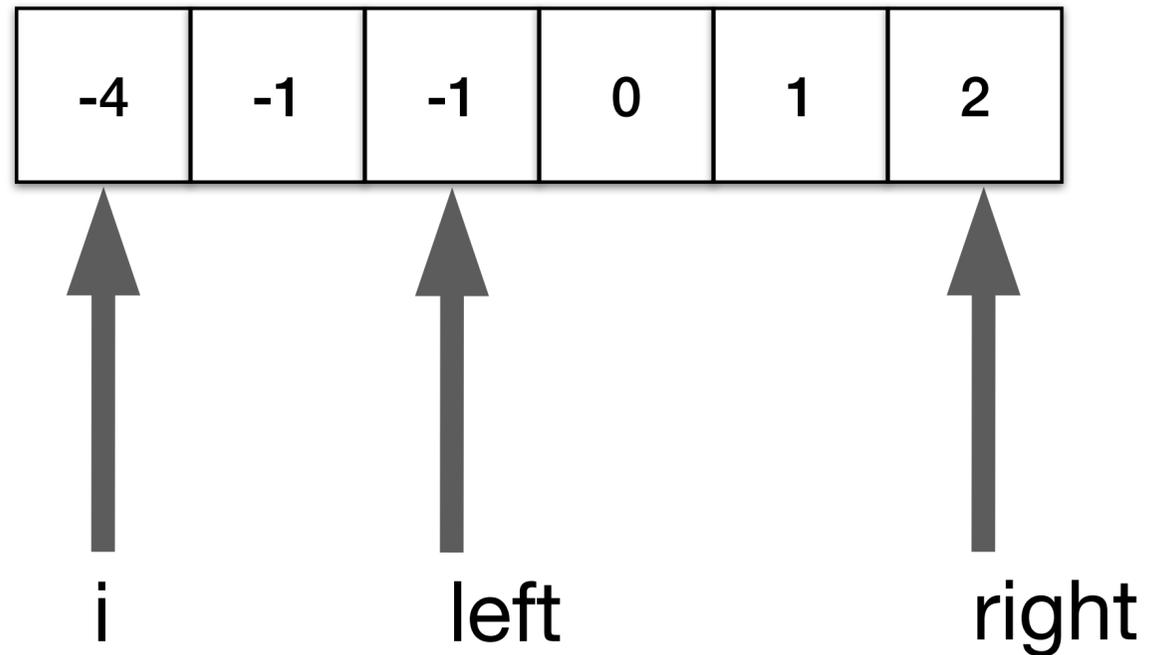
$$\text{Tagret} = 0$$

$$-3 < 0 \text{ OR } \text{currentSum} < \text{target}$$

$$\text{Output} = []$$

Less than check scenario in opposite direction pointers template

3 Sum - Test Case -1



$$\text{CurrentSum} = -4 + (-1) + 2 = -3$$

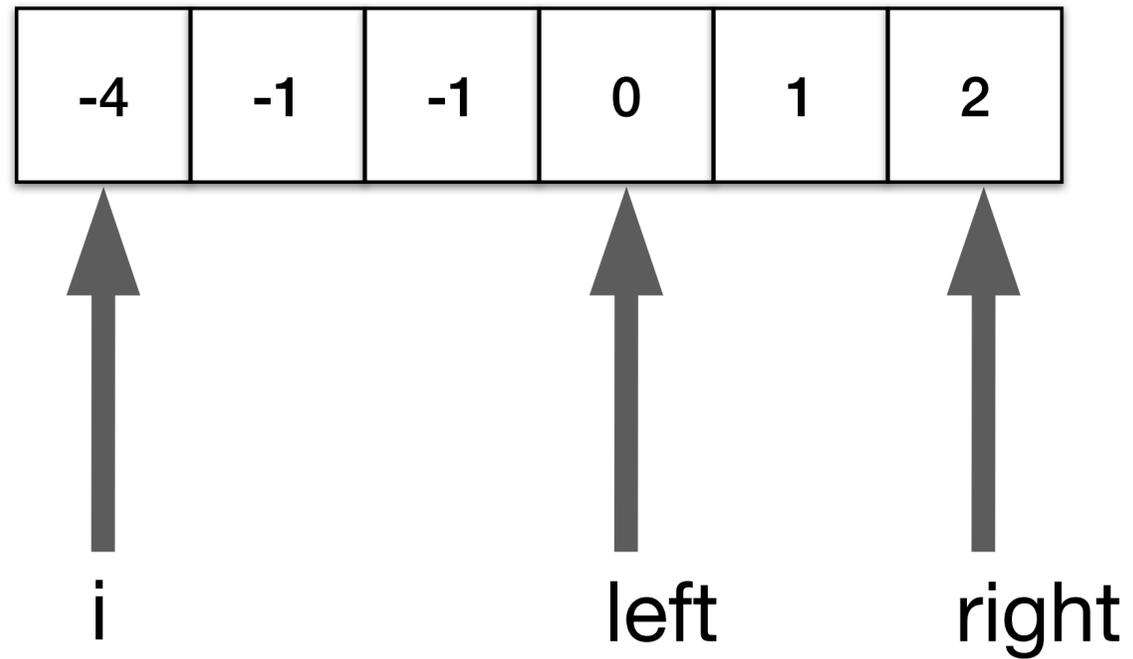
$$\text{Tagret} = 0$$

$$-3 < 0 \text{ OR } \text{currentSum} < \text{target}$$

$$\text{Output} = []$$

Less than check scenario in opposite direction pointers template

3 Sum - Test Case -1



$$\text{CurrentSum} = -4 + 0 + 2 = -2$$

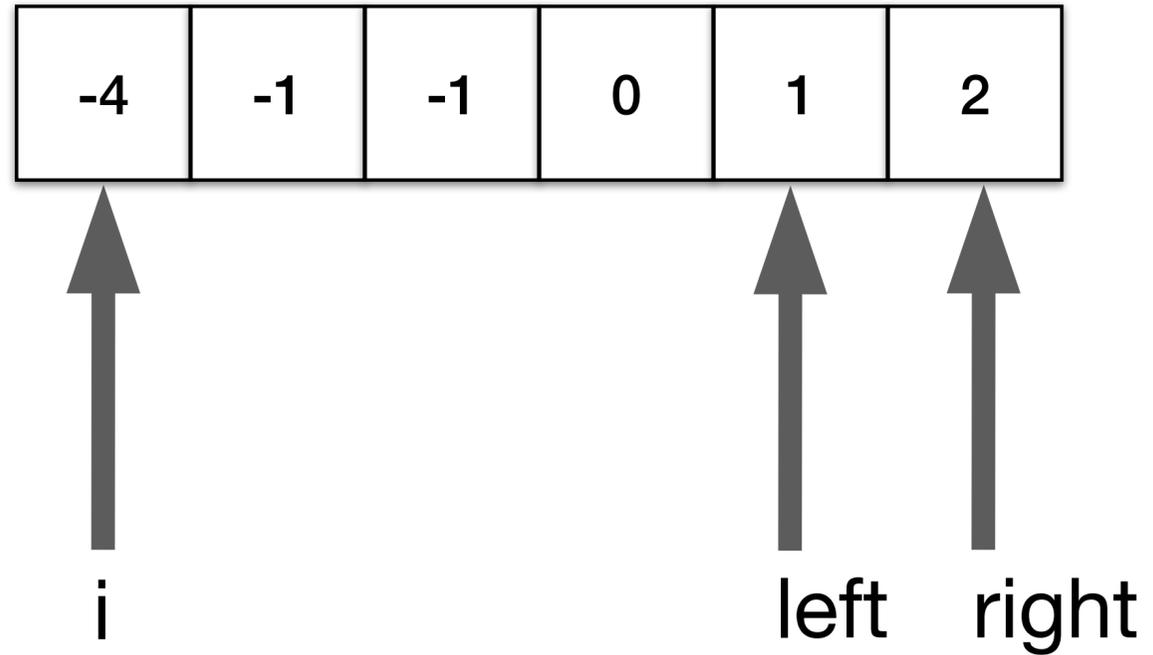
$$\text{Tagret} = 0$$

$$-2 < 0 \text{ OR } \text{currentSum} < \text{target}$$

$$\text{Output} = []$$

Less than check scenario in opposite direction pointers template

3 Sum - Test Case -1



$$\text{CurrentSum} = -4 + 1 + 2 = -2$$

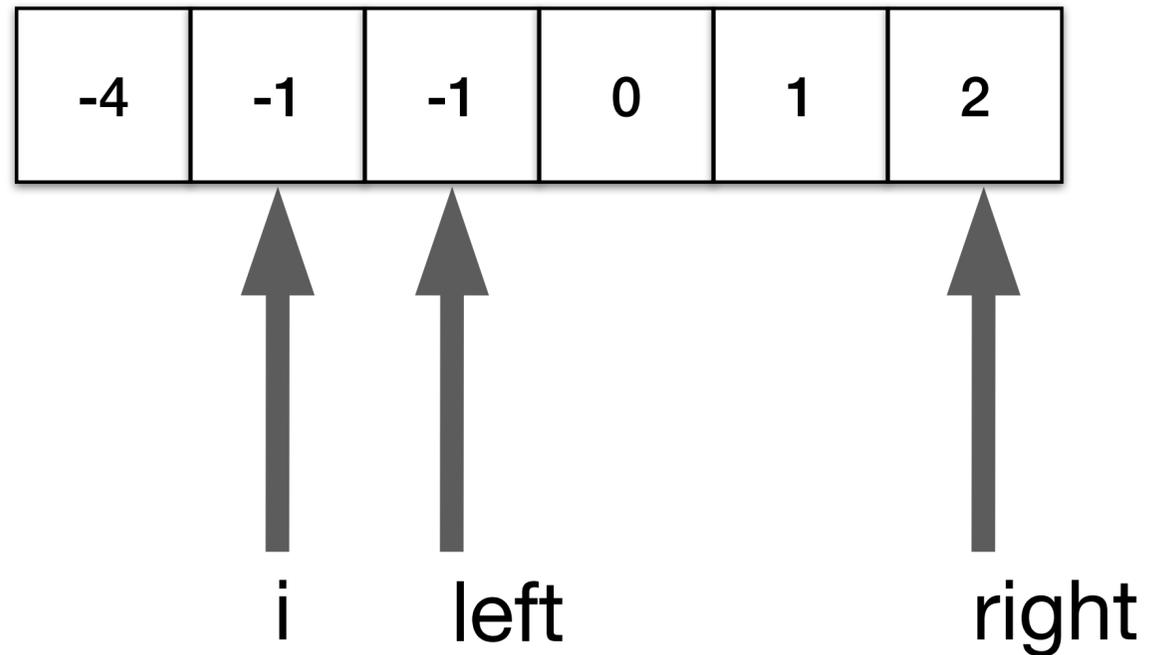
$$\text{Target} = 0$$

$$-1 < 0 \text{ OR } \text{currentSum} < \text{target}$$

$$\text{Output} = []$$

Less than check scenario in opposite direction pointers template

3 Sum - Test Case -1



$$\text{CurrentSum} = -1 + (-1) + 2 = 0$$

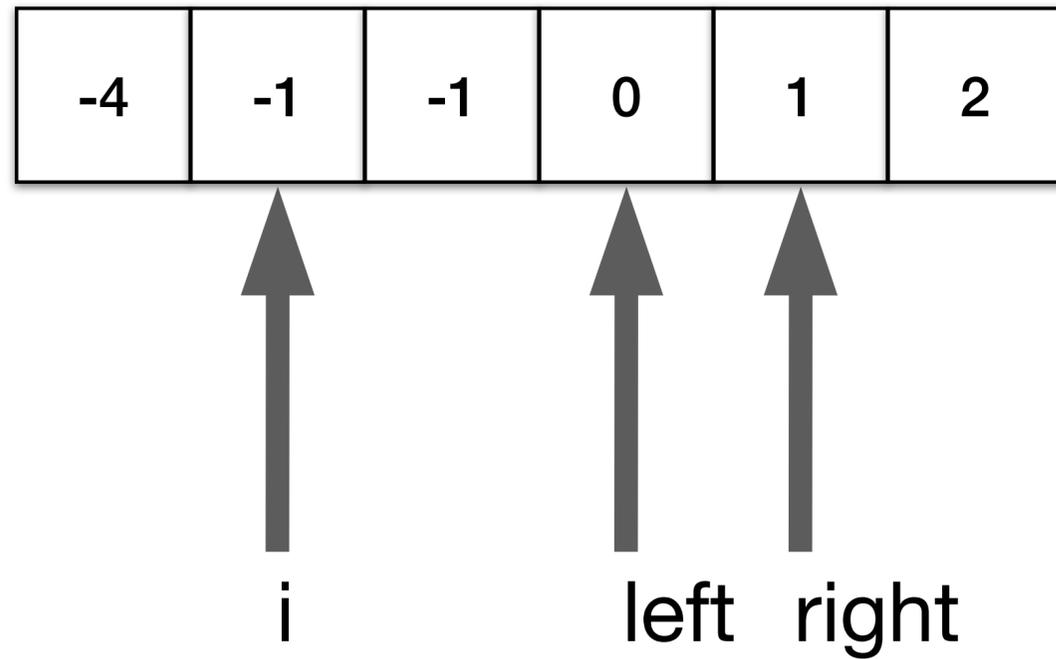
$$\text{Target} = 0$$

$$0 = 0 \text{ OR } \text{currentSum} == \text{target}$$

$$\text{Output} = [-1, -1, 2]$$

Equality check scenario in opposite direction pointers template

3 Sum - Test Case -1



$$\text{CurrentSum} = -1 + 0 + 1 = 0$$

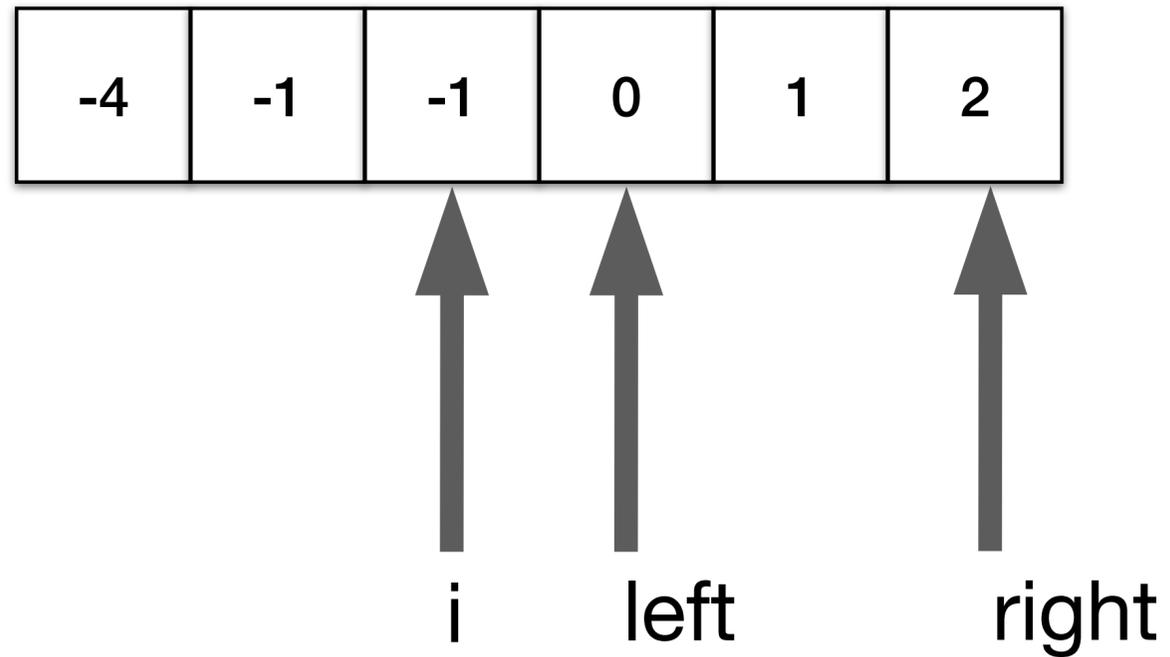
$$\text{Tagret} = 0$$

$$0 = 0 \text{ OR } \text{currentSum} == \text{target}$$

$$\text{Output} = [-1, -1, 2], [-1, 0, 1]$$

Equality check scenario in opposite direction pointers template

3 Sum - Test Case -1



$$\text{CurrentSum} = -1 + 0 + 2 = 1$$

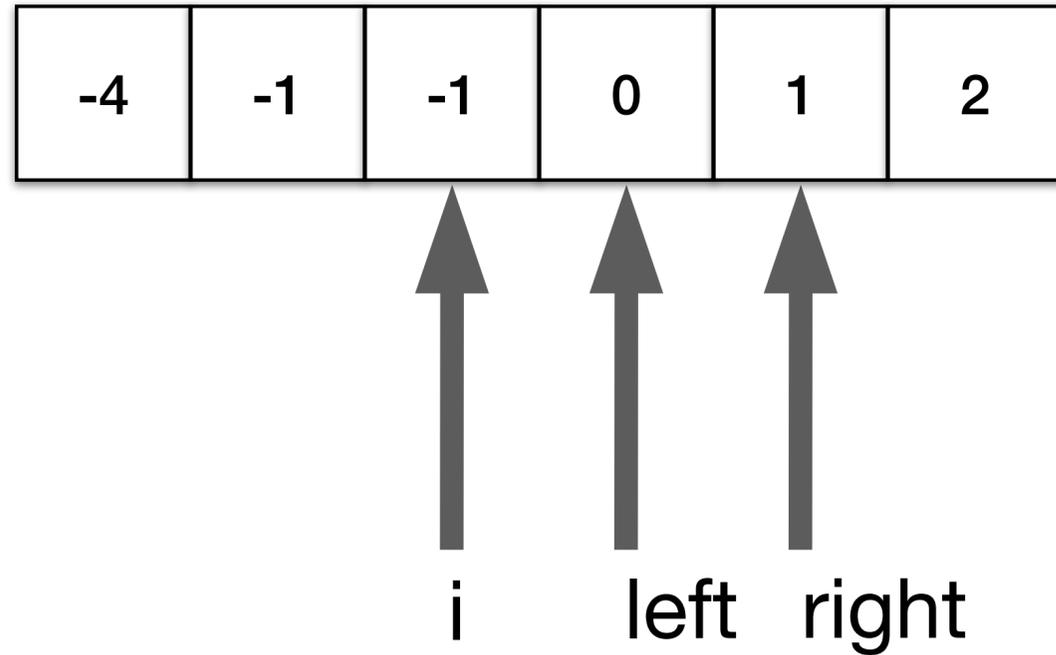
$$\text{Tagret} = 0$$

$$1 > 0 \text{ OR } \text{currentSum} > \text{target}$$

$$\text{Output} = [-1, -1, 2], [-1, 0, 1]$$

Greater than check scenario in opposite direction pointers template

3 Sum - Test Case -1 (Duplicate output Scenario)



$$\text{CurrentSum} = -1 + 0 + 1 = 0$$

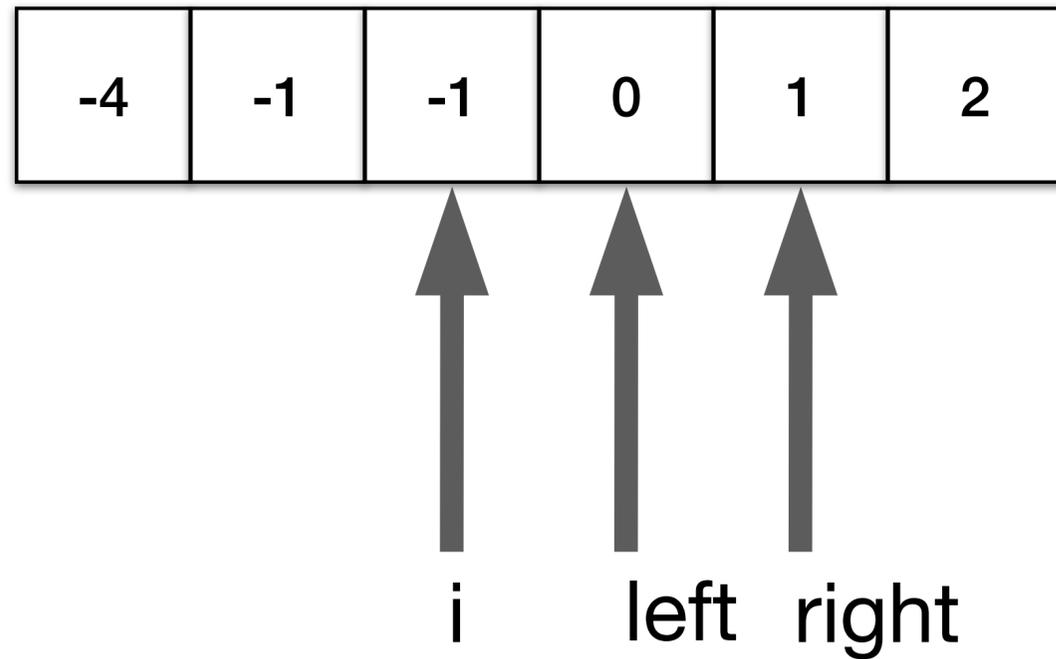
$$\text{Target} = 0$$

$$0 = 0 \text{ OR } \text{currentSum} == \text{target}$$

$$\text{Output} = [-1, -1, 2], [-1, 0, 1], [-1, 0, 1]$$

Equality check scenario in opposite direction pointers template

3 Sum - Test Case -1 (Duplicate output Scenario)

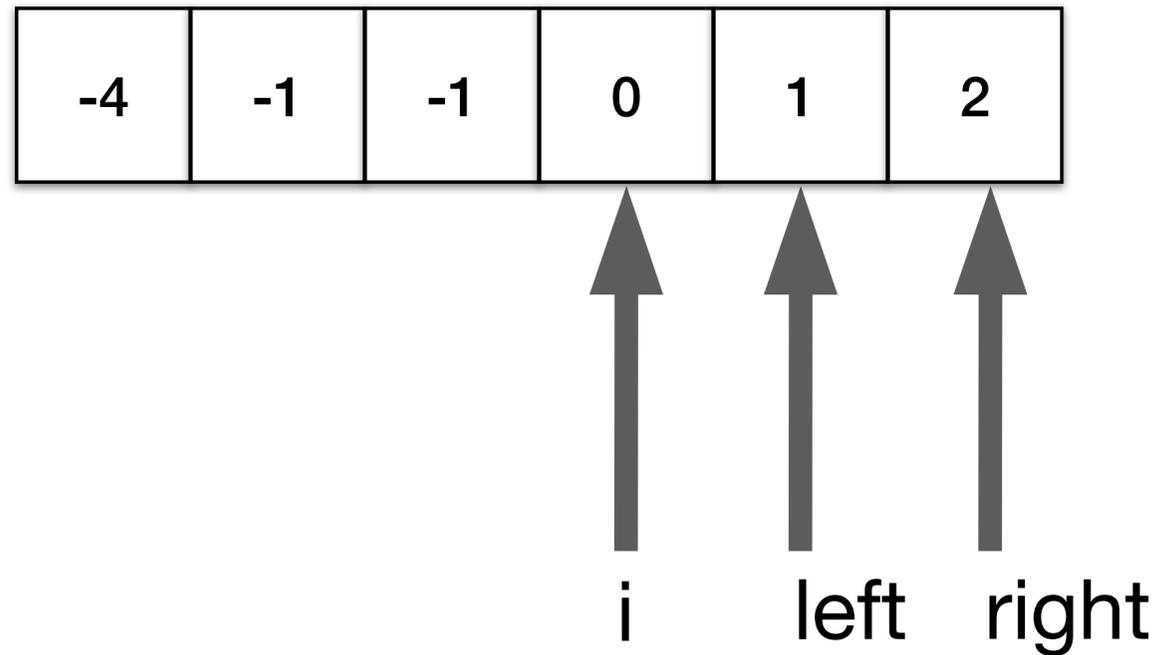


To handle this duplicate scenario we need to implement the below conditions

If $-1 == -1$ so we will skip calculations for this i index

Output = $[-1, -1, 2], [-1, 0, 1]$

3 Sum - Test Case -1



$$\text{CurrentSum} = 0 + 1 + 2 = 3$$

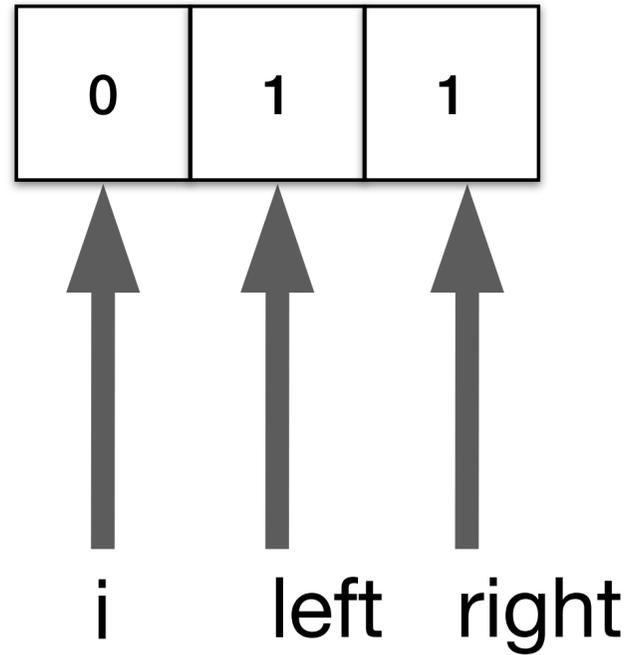
$$\text{Tagret} = 0$$

$$3 > 0 \text{ OR } \text{currentSum} > \text{target}$$

$$\text{Output} = [-1, -1, 2], [-1, 0, 1]$$

Greater than check scenario in opposite direction pointers template

3 Sum - Test Case - 2



$$\text{CurrentSum} = 0 + 1 + 1 = 2$$

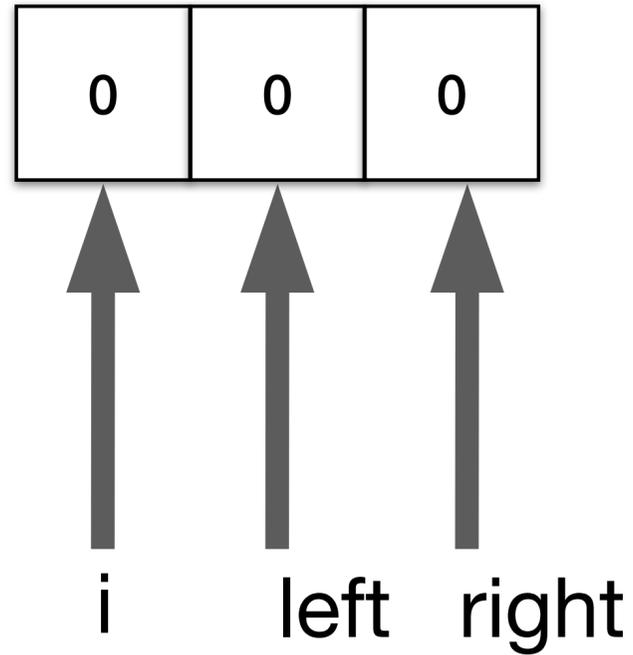
$$\text{Tagret} = 0$$

$$2 > 0 \text{ OR } \text{currentSum} > \text{target}$$

$$\text{Output} = []$$

Greater than check scenario in opposite direction pointers template

3 Sum - Test Case - 3



$$\text{CurrentSum} = 0 + 0 + 0 = 0$$

$$\text{Tagret} = 0$$

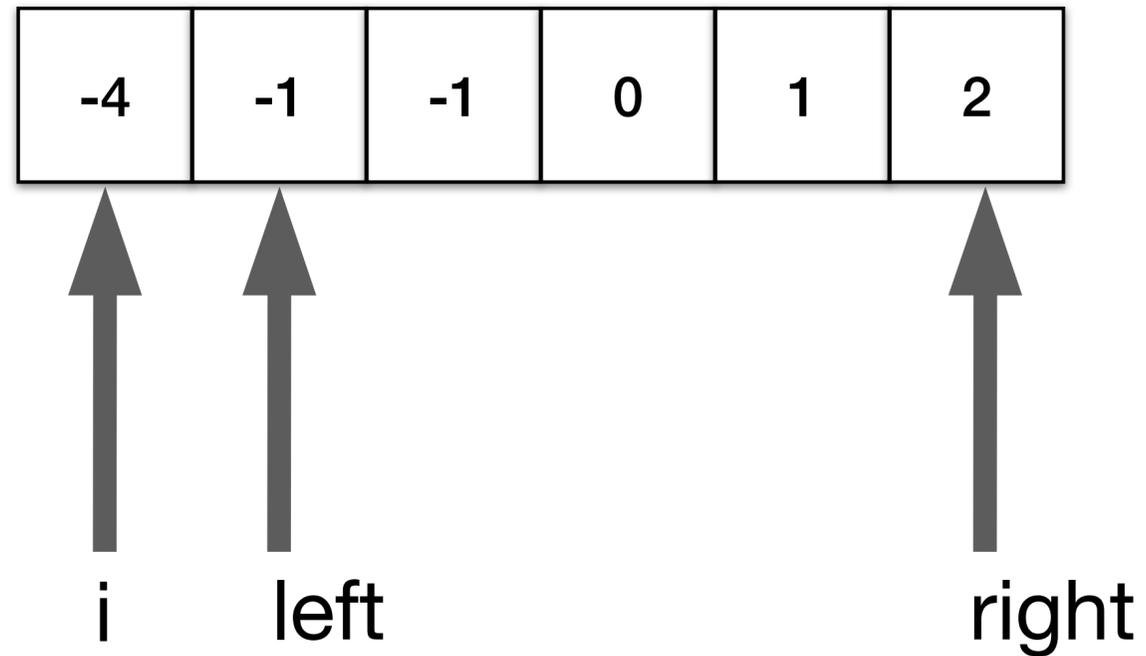
$$0 == 0 \text{ OR } \text{currentSum} == \text{target}$$

$$\text{Output} = [0, 0, 0]$$

Equality check scenario in opposite direction pointers template

Dynamic Parameters

Step 1 :- Template Implementation



$i = 0$

$left = 1 = i + 1$

$right = 5 = \text{nums.length} - 1$

`sort(nums)`

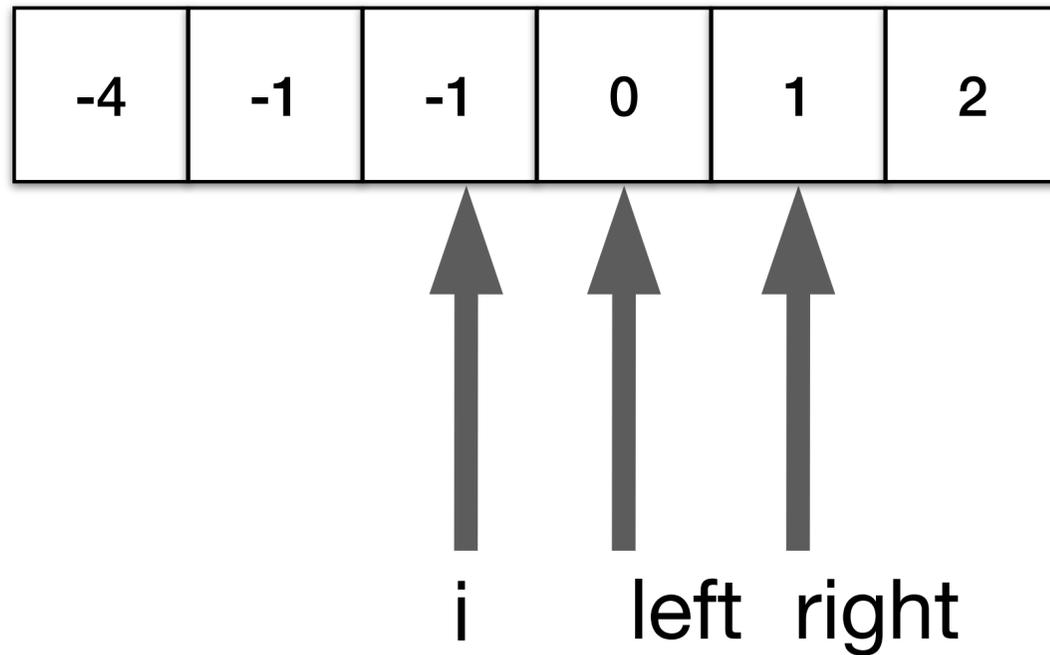
Pointers Details :-

Loop for i :- start from 0 to $\text{nums.length} - 2$

$left = i + 1$

$right = \text{nums.length} - 1$

Step 2 :- I/O parameters



$$\text{CurrentSum} = -1 + 0 + 1 = 0$$

$$\text{Target} = 0$$

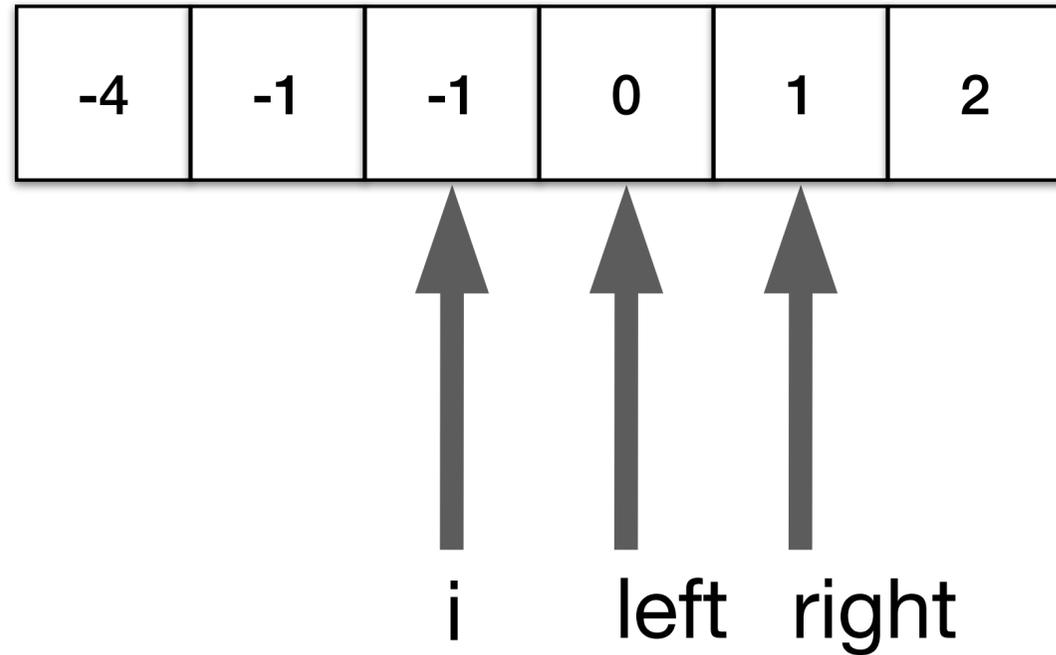
$$\text{Output} = [-1, -1, 2], [-1, 0, 1]$$

$$\text{Param1} = \text{nums}[i] + \text{nums}[\text{left}] + \text{nums}[\text{right}]$$

$$\text{Param2} = 0$$

$$\text{Output} = \text{list of}(\text{nums}[i], \text{nums}[\text{left}], \text{nums}[\text{right}])$$

Step 3 :- Skip the duplicates



If $-1 == -1$ so we will skip calculations for this i index

If $\text{nums}[i] == \text{nums}[i-1]$ then continue the loop

Steps Completed

3 Sum - Corner Test Case

-10	-5	-5	-4	-4	-3	-2	-2	0	0	1	2	2	2	2	5	5
-----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---

↑
i

↑
left

↑
right

$$\text{CurrentSum} = -5 + 0 + 5 = 0$$

$$\text{Target} = 0$$

$$0 == 0 \text{ OR } \text{currentSum} == \text{target}$$

$$\text{Output} = [-5, 0, 5]$$

Equality check scenario in opposite direction pointers template

3 Sum - Corner Test Case

-10	-5	-5	-4	-4	-3	-2	-2	0	0	1	2	2	2	2	5	5
-----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---

↑
i

↑

left

↑

right

$$\text{CurrentSum} = -5 + 0 + 5 = 0$$

$$\text{Target} = 0$$

$$0 == 0 \text{ OR } \text{currentSum} == \text{target}$$

$$\text{Output} = [-5, 0, 5], [-5, 0, 5]$$

Equality check scenario in opposite direction pointers template

Step 4 :- corner test cases scenario

-10	-5	-5	-4	-4	-3	-2	-2	0	0	1	2	2	2	2	5	5
-----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---

↑
i



left



right

while `nums[left] == nums[left+1]`
so `left++`

while `nums[right] == nums[right-1]`
so `right--`