

2. Two Sum II Input Array Is Sorted- 20/08/24 (Easy)

input is sorted

Input: num:

| | | | |
|---|---|----|----|
| 0 | 1 | 2 | 3 |
| 2 | 7 | 11 | 15 |

target: 9

Output: {1, 2} → 1-based index

The diagram illustrates the input array [2, 7, 11, 15] with indices 0, 1, 2, 3. The target is 9. The output is the 1-based index {1, 2}, which corresponds to the values 7 and 11 in the array.

Approach :-

| | | | |
|---|---|----|----|
| 0 | 1 | 2 | 3 |
| 2 | 7 | 11 | 15 |

 target = 9

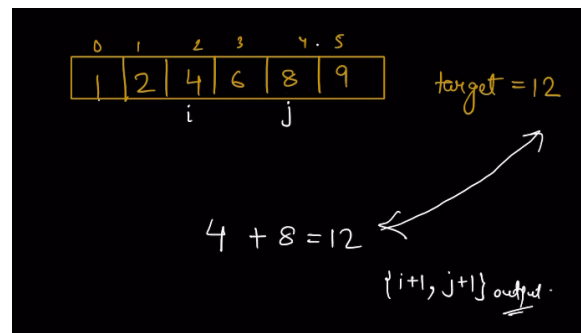
i j

The diagram shows the array [2, 7, 11, 15] with indices 0, 1, 2, 3. The target is 9. The variables i and j are shown below the first and last elements of the array, respectively.

add $i+j$ and see if $==$ target
if number is greater then $j--$
else $i++$

$$2 + 15 = 17$$

$$17 > 9$$



```

i = 0 ;
j = n-1 ;

while (i < j) {
    num[i] + num[j] > target
        j-- ;
    num[i] + num[j] < target
        i++ ;
}

```

Solution

```

class Solution {
public:
    vector<int> twoSum(vector<int>& numbers, int target) {
        int n = numbers.size();
        int i=0;
        int j=n-1;

        while(i<j){
            int sum = numbers[i]+numbers[j];

```

```
        if(sum>target){
            j--;
        }else if(sum<target){
            i++;
        }else{
            return {i+1,j+1};
        }
    }
    return {};
};
```