Two Pointers Approach

↳ Sorted array → [20, (40, 50, 75, 120, 145, 200)]
0 (1 2 3 4 5 6)

(i, j) ⇒ arr(i) + arr(j) = target

a + b = target ⇓ entered by user

target = 90
(1, 2) ⇒ 40 + 50
= 90

a = 20 ←
* target - b = 90 - 20 = 70 ←
↳ searching on ← sorted array remaining elements

- Linear Search → O(n²) → ⇓ Brute force approach
- Binary Search → O(n log n)

Two Pointer Approach

(l, r) → ending index

Starting index

0	1	2	3	4	5	6
20	40	50	75	120	145	200
l	l	r	r	r	r	r

target = 90

145 + 20 = 165

120 + 20 = 140

75 + 20 = 95

70 < 90

40 + 50 = 90

```
for i = 0 to n-1:
    if arr(l) + arr(r) == target:
        return l, r
    elif arr(l) + arr(r) > target:
        r = r - 1
    else:
        l = l + 1
```

20 + 200

220 > 90

return -1, -1

↳ No such kind of pair exists which adds up & give the given target value.

O(n)