167. Two Sum II - Input Array Is Sorted

https://www.youtube.com/watch?v=f7GpcuyYRIQ

https://leetcode.com/problems/two-sum-ii-input-array-is-sorted/

O(n)

O(n) significa que o tempo de **execução do algoritmo cresce linearmente com o tamanho da entrada**. Se a entrada dobrar, o tempo de execução também dobrará aproximadamente.

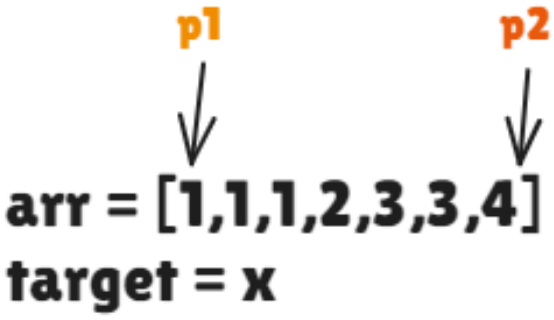
o código só vai precisar percorrer o array somente uma vez.

- n (tamanho do array)

---

target

- target é o resultado que quero alcançar com a soma.
- para fazer a soma, vamos trabalhar com dois ponteiros:
  - `p1` no index (0)
  - `p2` no index (n-1)



resultado

- o que fazer com o resultado da soma quando ele não é igual ao target? $p1 + p2 = p$

  $p < target$: ***o p1 é movido para frente.***

  $p > target$: ***o p2 é movido para trás.***

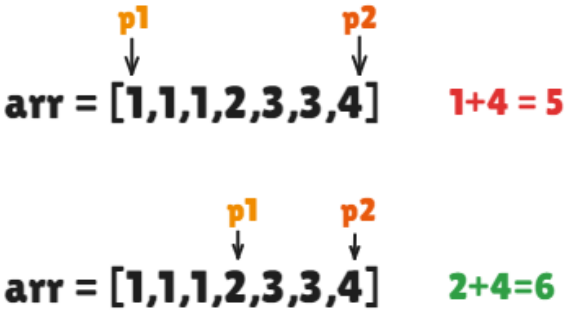
---

exemplo 1:

`target = 6`

`arr = [1,1,1,2,3,3,4]`

- nesse exemplo o primeiro resultado é menor que o target ⇒ então movemos o `p1`

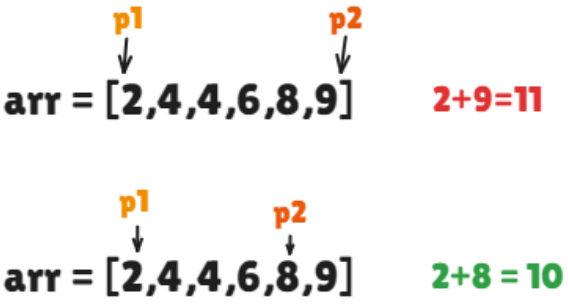


exemplo 2:

`target = 10`

`arr = [2,4,4,6,8,9]`

- já nesse exemplo o primeiro resultado é maior que o target ⇒ então movemos o `p2`



```
def sum (target, arr):
    p1 = 0
    p2 = len(arr) - 1

    while(p1 < p2):
        sum = arr[p1] + arr[p2]

        if(sum > target):
            p2-=1

        if(sum < target):
            p1+=1

        if(sum == target):
           return [p1 + 1, p2 + 1]

sum(6, [1,1,1,2,3,3,4]) # [4,7]
sum(10, [2,4,4,6,8,9]) # [1,5]
sum(-1, [-1,0]) # [1,2]
sum(9, [2,7,11,15])  # [1,2]
sum(200, [3,24,50,79,88,150,345]) #[3,6]
```