Two pointers:

210 leetcodes

Time complexity for array and string is reduced to O(n²) to O(n).

What are two pointers?

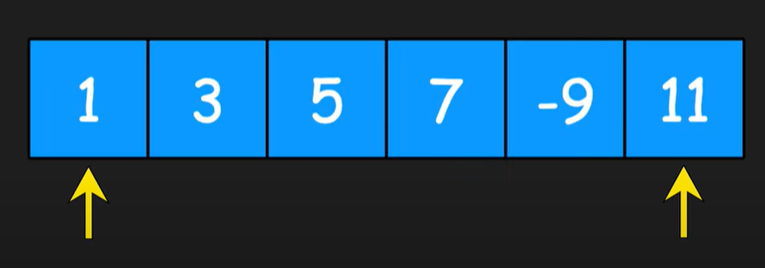
Pointer is a variable that points or represents an index to a position within an array, Linked list.

If I use linear comparison in data structure, I have to use the nested loop whose time complexity is O(n²).

To optimize code, I can use 2 points which points two indices within a data structure.

TYpes of 2 pointers:

1. Converging pointers:



Pointers are initially assigned to two ends of an array. These pointers will be moved towards each other.

Based on comparisons, the pointers will be moved.

These movement of pointers will be stopped when a condition is met or these have crossed each other.

These techniques of two pointers are used when I need to compare the elements within array or string.

Examples:

Check if the string is pallindrome.

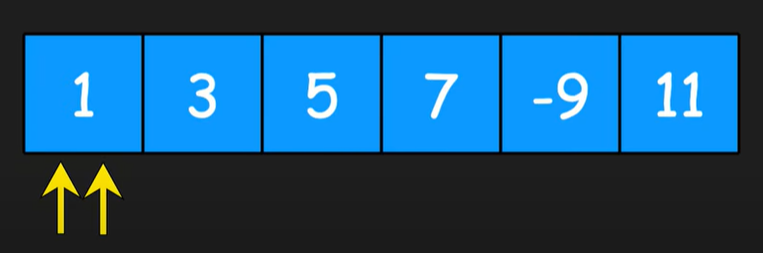
Set the pointer at index and other pointer at index (n-1)

If these match, move the pointers towards each other. Repeat the process.

If these do not match, then return False. Not a pallindrome.

pic

1. Parallel pointers:



Both pointers will be pointing at 0th index. Both of these will be moved to single direction.

Right pointer will move faster, skipping one index so that we have an idea what information lies as we go within an array.

The left pointer will be at the back, holding all the information.

This is used in sliding window.

<pic>

It is used to check subarrays, to meet specified criteria.

1. Triggered based pointers:

In this type, right pointer will be moved independently based on a condtion. Once the condition is met, the left pointer will move which is used to find more information related to right pointer.

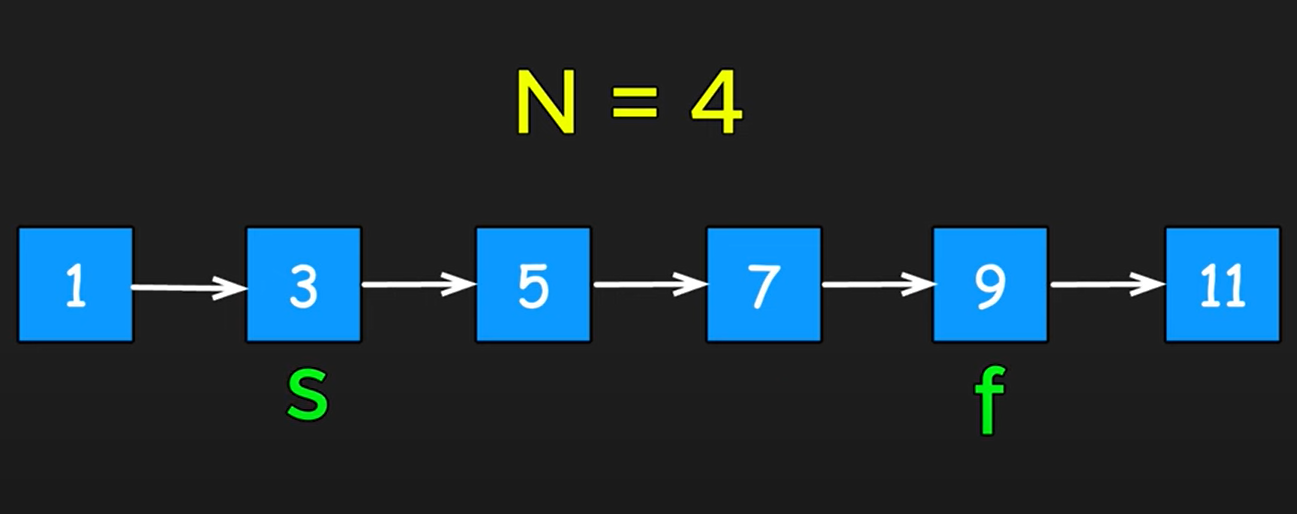
This is used, when I have to find the information in stages.

Ex: find nth node from end of linked list.

Right pointer will move till it reaches nth node. Once it reaches n, left pointer which is at 0th index will move.

The movement is such that, both the pointers will move together from their existing positions.

Each movement will have same difference in distance, which also can be referred as window.



R pointer reaches 7, as n= 4, while L will point to 1.

Once R reaches N and meet condition, both L and R pointer will move until R will reach end. At this point L pointer will be pointing to N.

When two pointers are used?

* Used in DS like arrays, strings, Linked list.
* If the DS is following predictable patterns like pallindrome or sorted array.

Keywords or things to notice in problem, so that I can apply two pointers:

1. If the array is sorted.
2. If the output needs to return the result of 2 values within array.

Leetcode examples:

1. Problem 283: Move zeros

Move all 0s to the end of the array but not altering the rest of numbers within an array.

Input: nums = [0,1,0,3,12]

Output: [1,3,12,0,0]

Naive approach: Create a new 0 array, when nums finds non zero element, copy to new arr, rest will be 0 and all 0s be at end.

Along with time, space complexity will be O(n).

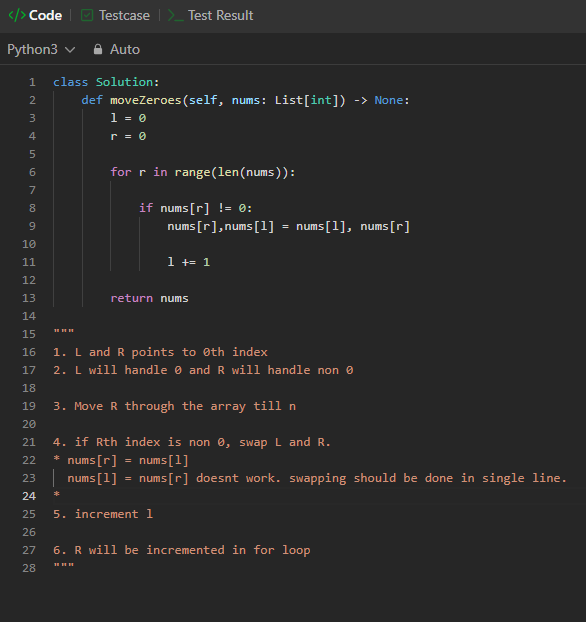
2 pointers in same:

Assign L R to 0th index. Move R to 1st index

When R finds non 0, then it will be swapped with L.

Now move both L and R.

L should point to 0 and R will be pointing at end of array.



Time complexity = O(n)

Space complexity = O(1)

Leetcode 11: Container with most water:

