13_08-Problem Solving with Arrays

Saturday, 13 August 2022, 10:15 AM

1. Sparse matrix
2. Rearrange -ve elem in left } Pivot
3. LCS (Longest Common Subsequence)
4. Cyclic rotation.

(1) Sparse matrix:-

[[0, 1, 2], [5, 0, 8], [1, 0, 0]]

[0, 1, 2], [5, 0, 8], [1, 0, 0]

>= 4 (CH)

0 1 0 5 0 8 1 0 0

no. of zeros = 4

total no. of elem = 9

(total no. of elem / 2) => 4.5 => 4

m = 3, n = 3

total: (m * n) / 2

TC: O(n)

Q

I/P [3, -2, 5, -4, 1, 6]

O/P [-2, -4, 3, 5, 1, 6]

-ve +ve

Hint: 2 pointers

Approach

0 -> pivot

[3, -2, 5, -4, 1, 6]

i j

i = -1, j = 0, pivot = 0

j = 2

Step1:- arr[0] => 3

3 < 0 => no/false } j++

arr[1] => -2

-2 < 0 => true } i++

swap arr[i] arr[j], j++

[-2, 3, 5, -4, 1, 6]

i j

arr[2] => 5

5 < 0 => false j++

arr[3] → -4

-4 < 0 ⇒ true

i++

swap

j++

[-2, -4, 5, 3, 1, 6]

-ve i +ve j

10

arr[4] → 1

1 < 0 ⇒ false

arr[5] → 6

6 < 0 ⇒ false

j++

20

j++ ? condition

algo

pivot

arr[j] < pivot } j++

swap

i++

arr = [1, 2, 3, 4]

OR [4 1 2 3]

[1 1 2 3] temp = 4 arr[0] = temp

[4 1 2 3] }

Q LCS (Longest Common Subsequence)

arr = [4, 1, 7, 2, 3, 8]

Sort()

OR 4

1
2
3
4

7
8
2

[1, 2, 3, 4, 7, 8]

+1 +1 +1 +1

Counter = 4

2

4

fc = 4 - 2