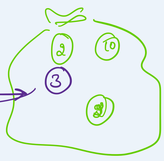
Q Given k- lists [all sorted]

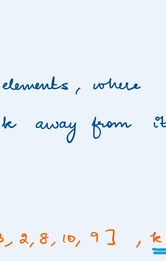
[1, 3, 7, 9, 15]
[2, 6, 10]
[10, 12]
[2, 7, 11, 19]

Merge k- sorted lists.

[1, 2, 2, 3, 6, 7, 7, 9, 10, 10, 11, 12, 15, 19]



Pair
→ ele
→ list
→ idx



Q Nearly sorted array

Given an array of n elements, where each element is almost k away from its target position.

→ Sort the array.

Eg: arr → [6, 5, 3, 2, 8, 10, 9] , k = 3

output → [2, 3, 5, 6, 8, 9, 10]

[6, 5, 3, 2, 8, 10, 9]

[2 3 5 6 8 9 10]

TC → O(n log k)

SC → O(k)

Q Find the k^{th} smallest element in a row-wise & col-wise sorted 2-D array.

Eg:

10	20	30	40
15	25	35	45
24	29	37	48
32	33	39	50

k = 3 → 20
k = 4 → 30

smallest ele → arr[0][0]

A[i][j] → min
→ A[i+1][j]
→ A[i][j+1]

TC → O(k) + O(k log k)

SC → O(k)

Q Top k frequent elements

Given an integer array, and a value (k), find the k most frequent elements.

Eg: [1, 1, 1, 2, 2, 3] k = 2
ans → [1, 2]

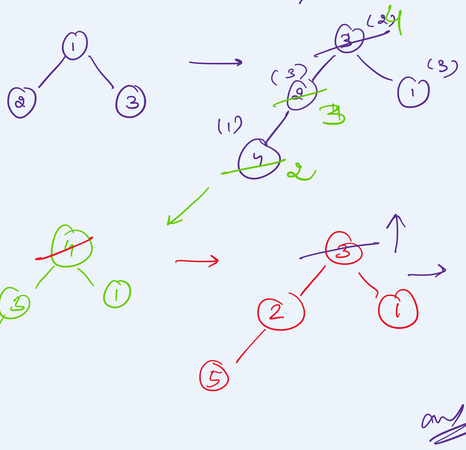
Eg: [1] k = 1
ans → [1]

Hash map (element → freq)

k = 3 [1, 1, 1, 2, 2, 2, 3, 3, 4, 5, 5, 5, 5]

→ Hashmap →

1	3
2	3
3	2
→ 4	1
→ 5	4



TC → O(n log k)

SC → O(n + k)