Max Chunks to make an array Sorted-1

Given an arr[N], containing permutation of the integers in the range [0, n-1]. you have to split the array into maximum possible no. of chunks, such that after individual sorting of those chunks, the whole array gets sorted

Ex:- {1, 0, 2, 3, 4}

The key Understanding here is, if we sort the elements ⇒ the obtained answer will be our indexes,

Ex:- {1, 0, 2, 3, 4} Sort ⟹ {0, 1, 2, 3, 4}

↳ These are our indexes only

So, the question array will be a permutation of indexes only, it won't go beyond range of indexes.

Ex 2: {2, 0, 1, 4, 3, 6, 7, 5, 8} Sort ⟹ {0, 1, 2, 3, 4, 5, 6, 7, 8}

↳ indexes only

So, what are chunks here?

"partitions"

arr[5] : {1, 0 | 2 | 3 | 4} ⇒ if we split the array into 4 parts and sorting each part individually the whole array will be sorted

Sorting individual chunk ⇓

{0 | 1 | 2 | 3 | 4} ⇒ So, the ans = 4 chunks

But, we can split as: {0 1 | 2 3 4} as well, like 2 chunks, this will also give us valid result of sorted array

But, we need to split the array as Maximum Possible Chunks

arr[9] : { 2 , 0 , 1 | 4 , 3 | 6 , 7 , 5 | 8 }

|| Sort individually

{ 1, 0, 2 | 3, 4 | 5, 6, 7 | 8 } ⟹ whole array is sorted

So, ans = 4 chunks

How

arr[9] : { 2 , 0 , 1 , 4 , 3 , 6 , 7 , 5 , 8 }

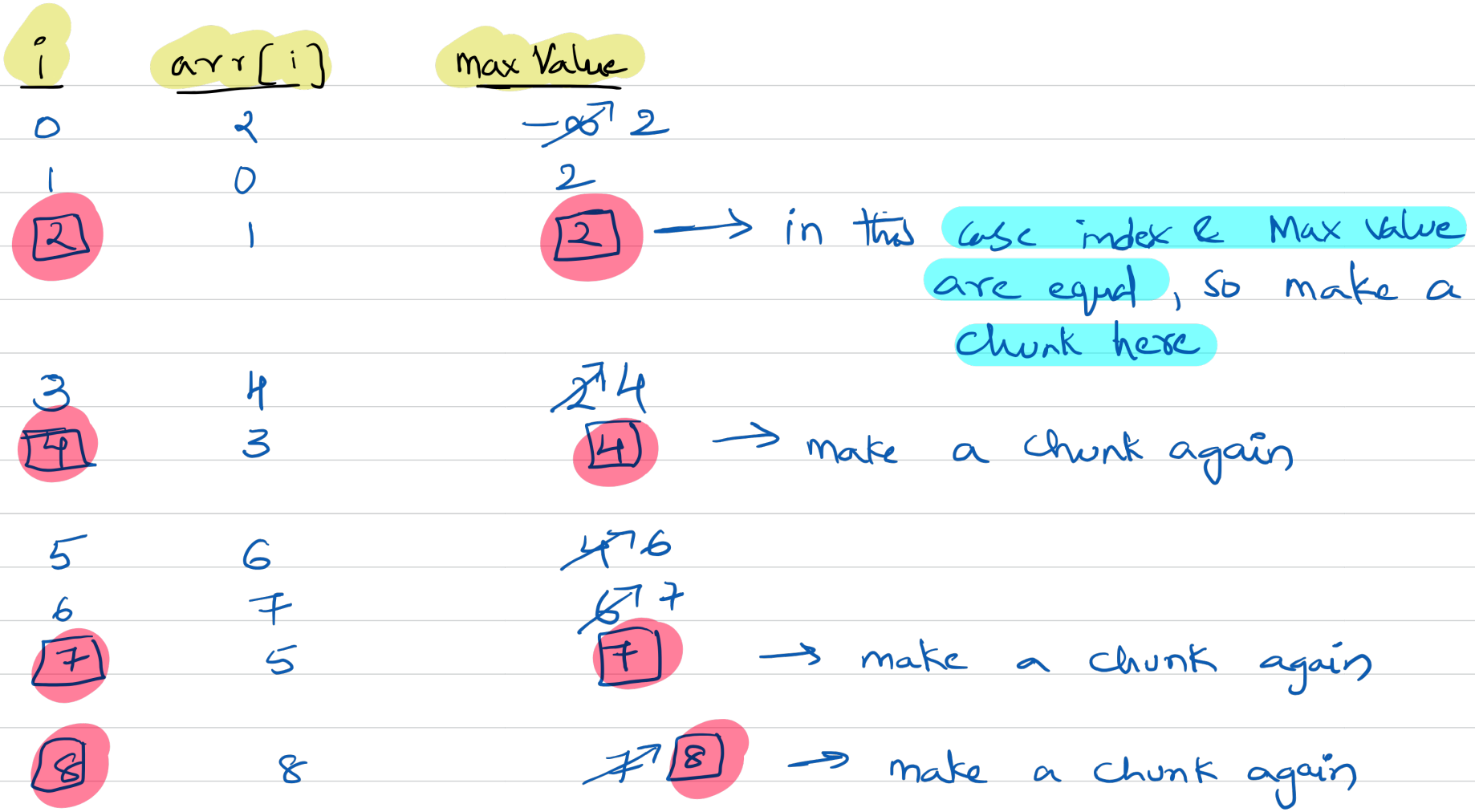
0 1 2 3 4 5 6 7 8

Steps

① if you want to make chunk, look for maximum index of the chunk

arr[9] : { 2 , 0 , 1 | 4 , 3 | 6 , 7 , 5 | 8 }

0 1 2 3 4 5 6 7 8



So, simply if you look for (max value & Index position) You can make chunks

Note:- if already a number positioned in its perfect index, Chunk + +