

(Editorial Soln) (Optimized)

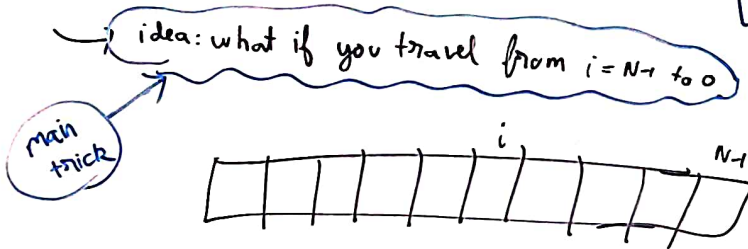
In the earlier problem we saw how segment tree was not required at all. And a plain tree set was used to find the best soln.

In the  $O(N \log N)$  solution. (Step 1)  $\rightarrow$  We were already populating the positions of  $minK, maxK$  into the TreeSet.

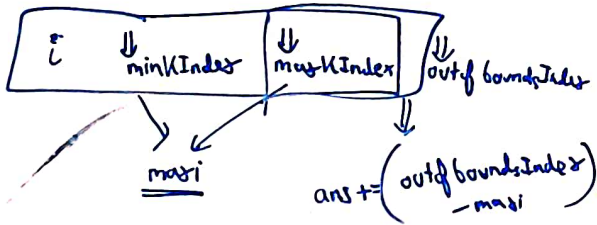
(Step 2)  $\rightarrow$  for any index  $i$  we

$\left[ \begin{matrix} \text{minTreeSet.ceiling}(i) \\ \text{maxTreeSet.ceiling}(i) \end{matrix} \right] \rightarrow$  take the max of it. then check for out of bounds index.

Optimised  $O(N)$  soln.  $\rightarrow$



At each  $i$  update for each type the leftmost index



$\begin{matrix} minKIndex = \min(minKIndex, i) \\ maxKIndex = \min(maxKIndex, i) \\ outOfBoundsIndex = \min(outOfBoundsIndex, i) \end{matrix}$

Which of these match is happening

These  $\begin{pmatrix} minKIndex \\ maxKIndex \\ outOfBoundsIndex \end{pmatrix} \rightarrow$  these 3 can be updated at  $O(1)$  at index visit  $\rightarrow$  Hence  $O(N)$