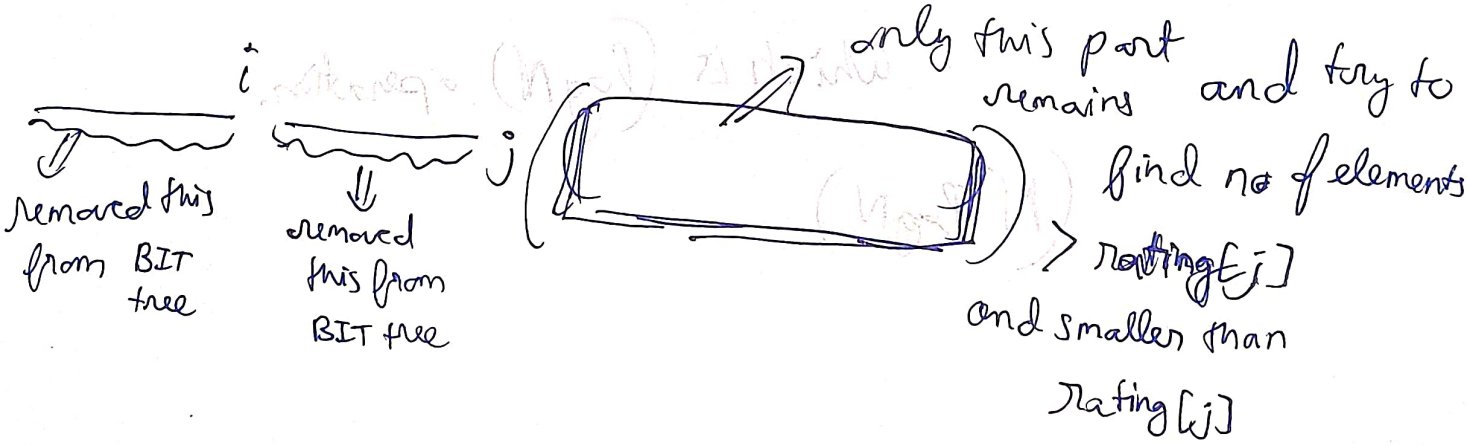
My idea

I was trying to solve the question from i's perspective.

i → For any given i I would choose a j



In my Approach N^2 loop is running

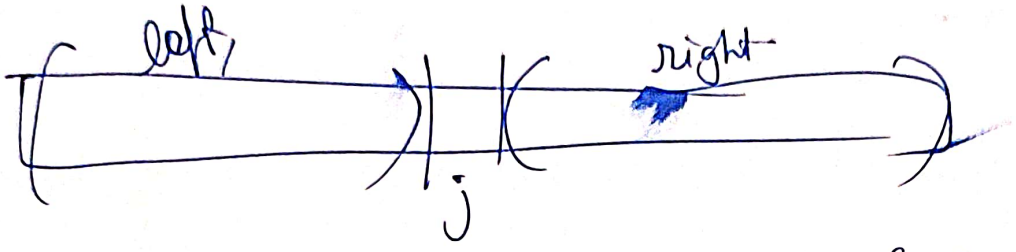
As am Looping on both (i and j).

and then doing bit tree lookup.

which is (log N) operation.

$(N^2 \log N)$

In the improved approach, change your perspective to j's point



For jth element you have to find 4 things,

leftsmaller, leftLarger from j

rightsmaller, rightLarger from j

for rat[i] < rat[j] < rat[k]

[leftsmaller, j, rightLarger.]

for rat[i] > rat[j] > rat[k]

~~rightLarger~~ i

[leftLarger, j, rightsmaller]

Just maintain and manipulate 2 BIT trees.

Left & Right.