Assignment-4

Here **divide and conquer** is used to generate a segment Tree which stores the pair of minimum and maximum.
In the function make it is made by first traveling to the leaf node by **dividing** the array into sub-parts until the size is 1 and then generate it by **conquering** as it comes back using its min and max of its left and right child. The query function returns the minimum and maximum in the range l to r in O(logN) time.
To find the indices i and j we iterate through the array and fix each **i** in each iteration and **binary_search** to find **j**. in each step of binary search we are calculating the minimum and maximum in the range i to mid using the **query** function in O(logN) time. Thus, the overall time complexity is-
O(N) [for iteration] * **O(logN)** [for binary_search] * **O(logN)** [for query function using **conquer** on segment tree]

Recurrence Relation:-
For segtree make() function:-

T(n)=2*T(n/2) + 1

Time Complexity – O(N)

For query() function:-

T(n)=T(n/2)+1

Time Complexity – O(logN)

For binary_search():-

T(n)=T(n/2)+1

Time Complexity – O(logN)

Overall Time Complexity – $O(N\log_2 N)$
Space Complexity – O(N)