19201089
Sec: 9

Explanation

Task 1

I have used the divide and conquer approach to solve the problem. The 'countPair' function performs merge sort using left and right subarrays found by dividing the array into subarrays. It uses the 'countPairMain' function to check if left array's element is greater than right and increments the count variable if so. It appends the greater element into a 'merged' list. The count ~~variable is~~ in total is obtained by adding the counts from the left and right subarrays and adding the count from the 'countPairMain' function. Sorting helps to find all kinds of possible pairs.

Task 2

The 'findMax' function applies ~~merge sort~~ divide & conquer to the input array 'arr', dividing it into subarrays recursively. ~~The merging takes place in the~~ 'findMaxMain' function ~~where it~~ compares elements from the left and right subarrays and calculates the expression left[i] + (right[j])^2. It updates the max value by taking the maximum of this max value and the calculated expression. Finally, it returns the original array and the maximum value for the expression.

Task 3

The 'quickSort' function recursively divides the input array 'a' into smaller subarrays using the 'partition' function. The partition function selects the last element (can be any element though) as 'x' as pivot and partitions the array in such a way that the elements smaller than x are at its left while the greater elements are at the right. The pivot element is then placed in its proper position. Then it returns the index of the pivot element. Recursively, this happens till the whole array is sorted.

Task 4

I have used a partitioning technique to rearrange the elements of an array and find the kth smallest element. It iteratively partitions the array and adjusts the p and r values for the next partition. If the pivot index == k-1, return the pivot index. If its smaller than k-1, set the p = pivot Index + 1 as its in the right side. Otherwise, set p = pivot index - 1 as the result is in the left side. This process was repeated for every query in the input file. If no such kth element exists, it returns None.