Niaz Morshed
22101783
Section - 24

Task 1: This is a Merge Sort Algorithm. Just the basic Divide and Conquer Method was used here to sort it efficiently. To elaborate more, I recursively Divided the Given array into two parts until there were only one element left and during merging, I checked for the larger number and put it in the right position and continued until the list was readjourned.

Task 2: This one just used the idea of the Divide and Conquer method to find the largest number in a fast or efficient way.

Task 3: This was about counting the pair(s) of aliens that Stands next to each other in a way if the Indexing was denoted by I, the right next to the alien would be j = i+1. The first Condition is i < j and the other condition is The height of the j has to be less than the Height of the ith indexed (Hi > Hj). If so, we will count how many of this situation is possible and print that.

Task 4: We have to find a pair that makes the maximum amount of number possible meeting those conditions. The conditions are: 1 <= i < j <= N and A[i] + A[j]^2. So, we used merge sort type method to break the array into two and then 4 and so on recursively then continued to find the best possible answer from those subsections. Finally connecting the subarrays and finding the ultimate solution possible.

Task 5: This is just a basic Quick sort algorithms as shown in the question. By setting up a pivot (randomly would be the best thing to do but for this task last value was used. Comparing to the pivot point, if requirements were met, swapping those values on the left or the right accordingly.

Task 6: The more or less same logic from the previous one here, just keeping a counter of the lowest term and maintaining the order to find the Serial of the lowest value for the required question is done here extra.