

Rearrange_Array_Problem explanation

March 22, 2022

0.1 Problem Statement

In this problem we need to rearrange array elements so as to form two number such that their sum is maximum. No inbuilt python sorting function is to be used and the time complexity should be $O(n \log n)$.

0.1.1 Design considerations

Since we have a set constraint for time complexity I decided to use quicksort algorithm. After sorting the input array I then alternately create the two numbers starting from the largest element.

0.1.2 Space and Time Complexity

Since quicksort is an in-place sorting algorithm and no extra data structure is created we can accurately posit that the space complexity is $O(1)$.

For the time complexity we have $O(n \log n)$ and $O(n)$ for creating the numbers. Since $O(n \log n)$ is larger than $O(n)$. Then we can say that the time complexity is $O(n \log n)$.