Name: Kaushick Suresh

~~To:~~

3) Mathematically derive the average runtime complexity of non-random Pivot
Since each element of the partitioning operations is compared with the pivot the recurrence relation for quick Sort, where n is the array size, the partitioning process takes O(n) Times.

$$T(n) = T(K) + T(n-K-1) + O(n)$$

In average case, the array is divided into two sub-arrays of almost equal size by the pivot.

So on average $K \approx n/2$, leading to the recurrence relation

$$T(n) = 2T(n/2) + O(n)$$

The Master theorem or the recursion tree approach can be used to solve this recurrence.

$$T(n) = aT(n/b) + O(n^d)$$

Now, we calculate the critical exponent:

$$\log_b a = \log_2 2 = 1$$

Since $d = \log_b a$, the Time complexity is $O(n^d \log n)$, which simplifies to

$$T(n) = O(n \log n)$$

Thus, the average cost time complexity of quick sort $O(n \log n)$