Saibkeswari Polarapu
UTA ID: 1002190695

HandsOn-6

3. Mathematically derive the average runtime complexity of the non-random pivot version of quicksort.

Let's first consider the array size of 'n'.
The recurrence relation for the quicksort is defined $T(n)$.

$T(n) \rightarrow$ average runtime of quicksort of size 'n'.

For the average case,
Pivot divides the array into two partitions equally.

Then the recurrence relation for the average scenario is

$$T(n) = O(n) + 2 \cdot T\left(\frac{n}{2}\right)$$

here,
$2T(n/2)$ represents average time for two recursive calls of size $n/2$.

from above equation,

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

$$T(n) = O(n) + 2\left(O\left(\frac{n}{2}\right) + 2 \cdot T\left(\frac{n}{4}\right)\right)$$

$$T(n) = O(n) + 2 \cdot O\left(\frac{n}{2}\right) + 4 \cdot T\left(\frac{n}{4}\right))$$

$$T(n) = O(n) + O(n) + 4(O(n/4)) + 8 \cdot T(n/8)$$

$$\vdots$$

$$T(n) = K \cdot O\left(\frac{n}{2^K}\right) + 2^K \cdot T\left(\frac{n}{2^K}\right)$$

here let's assume k iterations

array size will be 1 when reaches base case

$$T(n) = \log_2 n \cdot O(1) + n \cdot T(1)$$

here $T(1) \rightarrow$ constant

$\therefore$ Therefore, the runtime complexity of quicksort for average case using the non-random pivot is $O(n\log n)$.

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