Hand-on-6

Swarani Bampally.
1002208330.

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

A) In non-random pivot choice of quicksort the pivot element can be first index element or last index element. So let array length be 'n'.

After one partition, the array can be divided into two subarrays of sizes 'm' and (n−m)

$\because \quad T(n) = T(n-m) + T(m)$

$$T(n) = \frac{1}{n}\left[\sum_{i=1}^{n-1} T(i) + \sum_{i=1}^{n-1} T(n-i)\right]$$

$$T(n) = \frac{2}{n}\left[\sum_{i=1}^{n-1} T(i)\right] \qquad \left[\because \sum_{i=1}^{n-1} T(i) = \sum_{i=1}^{n-1} T(n-i)\right]$$

$$n \times T(n) = 2\left[\sum_{i=1}^{n-1} T(i)\right]$$

subtitute $n = n-1$ in above equation.

$$(n-1)T(n-1) = 2\left[\sum_{i=1}^{n-2} T(i)\right]$$

By substracting above two equations

$$n(T(n)) - (n-1)(T(n-1)) = 2T(n-1) + n^2(K) - (n-1)^2(K)$$

where K is a constant.

$$n(T(n)) = T(n-1)(2+n-1) + K + 2nK - K$$

$$n(T(n)) = (n+1)T(n-1) + 2nK$$

divide both sides by $n(n-1)$

So we get

$$\frac{T(n)}{n+1} = \frac{T(n-1)}{n+2} * \frac{K}{n+1} \Rightarrow \frac{T(n-1)}{n} = \frac{T(n-2)}{n-1} + \frac{2K}{n} \quad [\because n = n-1]$$

$$\frac{T(n)}{n+1} = \frac{T(n-2)}{n-1} + \frac{2k}{n+1} + \frac{2k}{n}$$

The final equation will be as by replacing n by (n-2) in the equation

$$\frac{T(n)}{n+1} = \frac{T(1)}{2} + 2k\left(\frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n-1} + \frac{1}{n} + \frac{1}{n+1}\right)$$

$$T(n) = 2k \log_2 n(n+1)$$

$$\therefore T(n) = \log_2(n*(n+1))$$

∴ The average runtime complexity of non-random pivot version of quicksort is ~~log~~ $O(n \log n)$.