QuickSort

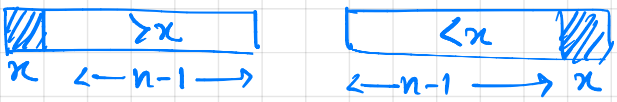
Pseudo Code

Quicksort (A[p,...,q])

    if p == q ← stop
    r ← Partition(A[p,...,q])
    Recursive Call
        Quicksort (A[p,...r-1])
        Quicksort (A[r+1,...,q])

Time Complexity Analysis

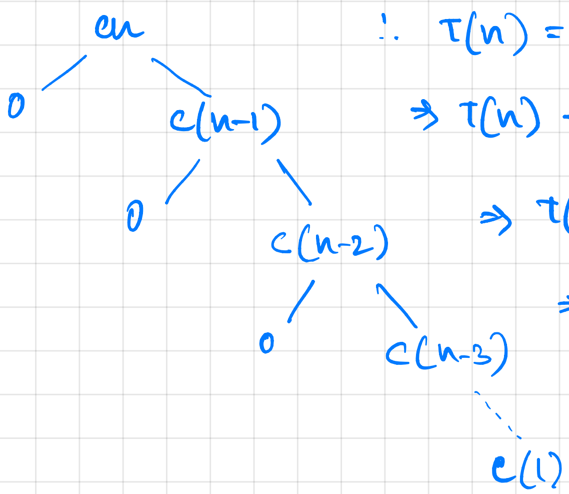
Worst Case :



At every step, the pivot is at the either ends and the splitting occurs in n-1 : 1 ratio.

$\therefore\ T(n) = T(n-1) + T(1) + \theta(n)$ Partitioning time complexity

$\Rightarrow T(n) = T(n-1) + \theta(n)$ [Assuming $T(1) = 0$]



$\therefore\ T(n) = cn + c(n-1) + c(n-2) + \cdots + c$

$\Rightarrow T(n) = c\left(n + (n-1) + (n-2) + \cdots + 2 + 1 \right)$

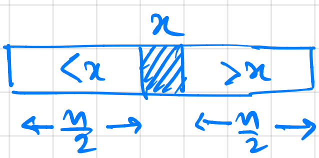
$\Rightarrow T(n) = c \cdot \frac{n(n-1)}{2}$

$\Rightarrow T(n) = c\left(\frac{n^2}{2} - \frac{n}{2}\right) = \frac{cn^2}{2} - \frac{cn}{2}$

$\Rightarrow T(n) = cn^2 - \left(\frac{cn^2}{2} + \frac{cn}{2}\right)$

$\Rightarrow T(n) = O(n^2)$ when $\frac{cn^2}{2} > \frac{cn}{2}$

Best Case :

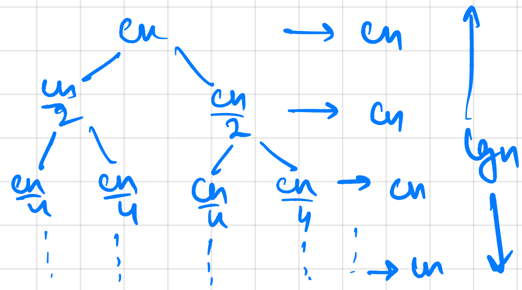


Always the partitioning occurs in this way at every stage.

$\therefore\ T(n) = 2T\left(\frac{n}{2}\right) + \theta(n)$

$\Rightarrow T(n) = \theta(n\lg n)$

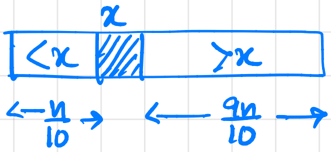
using Master's Theorem



$\therefore\ T(n) = cn\lg n$

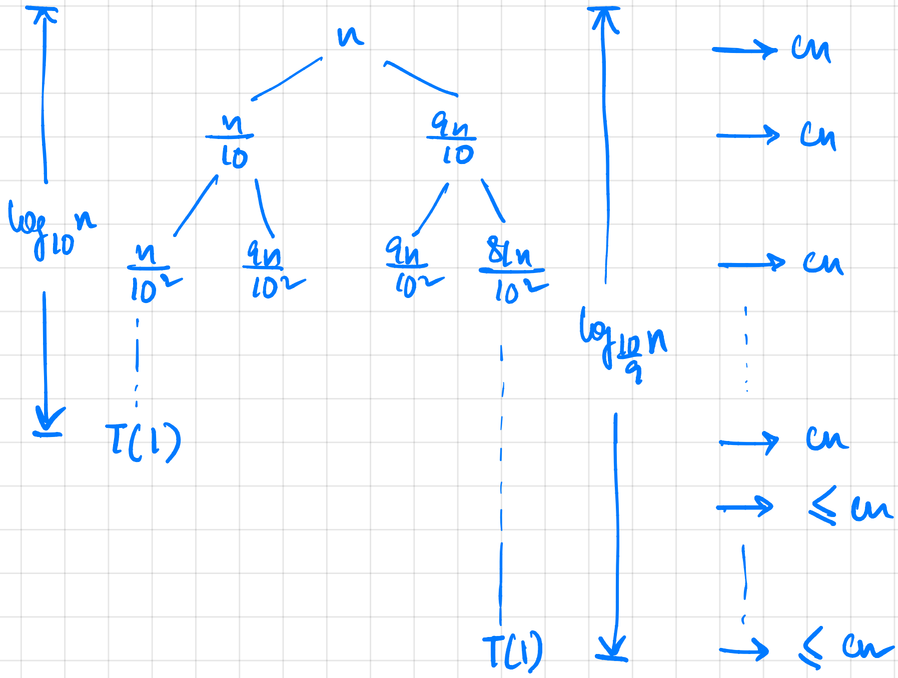
$\Rightarrow T(n) = \theta(n\lg n)$

Almost Best Case :



Suppose the partitioning occurs at this way for every step.

$$T(n) = T\left(\frac{n}{10}\right) + T\left(\frac{9n}{10}\right) + \theta(n)$$



One part is getting divided at a very fast rate. thus it will reach $T(1)$ faster at a small height as shown.

The part getting divided slowly will reach $T(1)$ after a long time. Hence the height of the tree will be longer as shown.

Considering the longer tree we obtain that,

$$T(n) \leq cn \log_{\frac{10}{9}} n \leq cn \lg n.$$

$$\Rightarrow T(n) = \theta(n \lg n)$$

In depth analysis of Worst Case Complexity

Suppose, $T(n) = T(k) + T(n-k) + \theta(n)$ —— (1), where

'k' is the random variable dependent for the partitioning.

Lets assume $T(n) = O(n^2) \Rightarrow T(n) \leq cn^2$ for $c>0$ and $n \geq n_0$. —— (2)

Using eq^n (2) in eq^n (1),

$$T(n) \leq \underbrace{\boxed{ck^2 + c(n-k)^2}}_{q} + \theta(n)$$

we want the max value of this to obtain the worst case time complexity

Let, $x = ck^2 + c(n-k)^2$

$$\Rightarrow \frac{dx}{dk} = 2ck + 2c(n-k)(-1)$$

$$\Rightarrow \frac{dx}{dk} = 2ck - 2cn + 2ck = 4ck - 2cn$$

$$\Rightarrow \frac{d}{dk}\left(\frac{dx}{dk}\right) = 4c > 0 \quad [\because c \text{ is a positive constant}]$$

$\therefore$ $4ck - 2cn = 0 \Rightarrow k = \frac{n}{2}$ this mean $k_{min} = \frac{n}{2}$.

$\Rightarrow k_{max} = 1$ or $n-1$.

So, $1 + (n-1)^2 = 1 + n^2 - 2n + 1 = n^2 - 2n + 2$.

$$T(n) \leq c(n^2 - 2n + 2) + \theta(n)$$

$$\Rightarrow T(n) \leq cn^2 - (2cn - 2c - \theta(n))$$

$\Rightarrow T(n) = \theta(n^2)$ when $2cn - 2c - \theta(n) > 0 \Rightarrow 2cn - 2c > \theta(n)$.

In depth analysis of Average Case Complexity

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

When the pivot is the first element / last element,

$$T(n) = T(1) + T(n-1) + \theta(n).$$

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

Assume, $i-1 = j \Rightarrow i = j+1$.

At $i = 2 \longrightarrow j = 1$

At $i = n-1 \longrightarrow j = n-2$

$$T(n) = \frac{1}{n}\left[2T(n-1) + \sum_{j=1}^{n-2}\{T(j) + T(n-j-1)\}\right] + \theta(n)$$

$$\Rightarrow T(n) = \frac{1}{n}\left[2T(n-1) + 2\sum_{k=1}^{n-2} T(k)\right] + \theta(n) \quad [\text{From the Select Algo}]$$

$$\Rightarrow T(n) = \frac{2}{n}\left[T(n-1) + \sum_{k=1}^{n-2} T(k)\right] + \Theta(n)$$

$$\Rightarrow T(n) = \frac{2}{n}\sum_{k=1}^{n-1} T(k) + \Theta(n). \longrightarrow (1)$$

Assume $T(n) = \Theta(n\lg n) \Rightarrow T(n) \le an\lg n + b$ where $a, b > 0$ for $n \geq n_0$ $\longrightarrow (2)$.

Putting eqn. (2) in eqn. (1), we obtain,

$$T(n) \le \frac{2}{n}\sum_{k=1}^{n-1}(ak\lg k + b) + \Theta(n)$$

$$\Rightarrow T(n) \le \frac{2a}{n}\sum_{k=1}^{n-1} k\lg k + \frac{2b(n-1)}{n} + \Theta(n)$$

Consider this part seperately.

$$\sum_{k=1}^{n-1} k\lg k = \sum_{k=1}^{\lceil \frac{n}{2}\rceil - 1} k\lg k + \sum_{k=\lceil \frac{n}{2}\rceil}^{n-1} k\lg k.$$

$\lg k$ is bounded by $\lg(\frac{n}{2})$ or $(\lg n - 1)$

$\lg k$ is bounded by $\lg n$.

$$\Rightarrow \sum_{k=1}^{n-1} k\lg k \le (\lg n - 1)\sum_{k=1}^{\lceil \frac{n}{2}\rceil - 1} k + \lg n \sum_{k=\lceil \frac{n}{2}\rceil}^{n-1} k$$

$$\Rightarrow \sum_{k=1}^{n-1} k\lg k \le \lg n \sum_{k=1}^{n-1} k - \sum_{k=1}^{\lceil \frac{n}{2}\rceil - 1} k$$

$$\Rightarrow \sum_{k=1}^{n-1} k\lg k \le \lg n \frac{n(n-1)}{2} - \frac{n/2(n/2 - 1)}{2}$$

$$\Rightarrow \sum_{k=1}^{n-1} k\lg k \le \frac{n^2}{2}\lg n - \frac{n^2}{8} \quad \text{for} \quad \frac{n}{2}\lg n - \frac{n}{4} > 0.$$

$$T(n) \le \frac{2a}{n}\left[\frac{n^2}{2}\lg n - \frac{n^2}{8}\right] + \frac{2b(n-1)}{n} + \Theta(n)$$

$$\Rightarrow T(n) \le \frac{2a}{n}\left(\frac{n^2}{2}\lg n\right) - \frac{2a}{n}\left(\frac{n^2}{8}\right) + \frac{2b(n-1)}{n} + \Theta(n)$$

$$\Rightarrow T(n) \le an\lg n - \frac{an}{4} + 2b + \theta(n)$$

$$\Rightarrow T(n) \le (an\lg n + b) - \left(\frac{an}{4} - b - \theta(n)\right)$$

$$\Rightarrow T(n) = \theta(n\lg n) \quad \text{when} \quad \frac{an}{4} - b \ge \theta(n).$$