BLG335E - #1 Quicksort Report

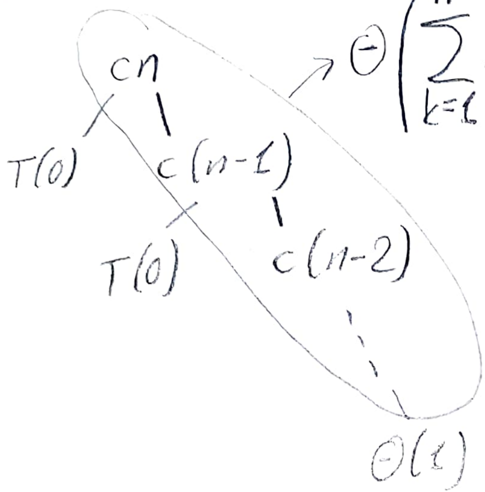
Ufuk Yarışan
070170206

1) Quicksort worstcase $\rightarrow T(n) = T(0) + T(n-1) + \Theta(n)$

Always pivot biggest or lowest one.

$$= \Theta(1) + T(n-1) + \Theta(n)$$
$$= T(n-1) + \Theta(n)$$

$\Theta(n) = cn$



$$\rightarrow \Theta\left(\sum_{k=1}^{n} k\right) = \Theta(n^2) \quad = \Theta(n^2)$$

$\Theta(n^2) = O(n^2)$ for worst case

Quicksort bestcase $\rightarrow T(n) = 2T(n/2) + \Theta(n)$

Always divide half

$$\underbrace{n^{\log_2 2} = n}_{\text{master theorem}} \qquad \Theta(n \cdot \log n) \text{ for best case.}$$

2) $X_k = \begin{cases} 1 & \text{if partition generates a } k : n-k-1 \text{ split} \\ 0 & \text{otherwise} \end{cases}$ for $k = 0, \dots, n-1$

$E[X_k] = Pr\{X_k = 1\} = 1/n$ equally likely

$$T(n) = \begin{cases} T(0) + T(n-1) + \Theta(n) & 0:n-1 \\ T(1) + T(n-2) + \Theta(n) & \\ \vdots & \vdots \\ T(n-1) + T(0) + \Theta(n) & \text{if } n-1:0 \end{cases}$$

$$= \sum_{k=0}^{n-1} X_k \left(T(k) + T(n-k-1) + \Theta(n)\right)$$

$$E[T(n)] = \frac{1}{n}\sum_{k=0}^{n-1} E[T(k)] + \frac{1}{n}\sum_{k=0}^{n-1} E[T(n-k-1)] + \frac{1}{n}\sum_{k=0}^{n-1}\Theta(n) = \frac{2}{n}\sum_{k=1}^{n-1} E[T(k)] + \Theta(n)$$

2 devam) $E[T(n)] = \frac{2}{n} \sum_{k=2}^{n-1} E[T(k)] + \Theta(n) \leqslant \frac{2}{n} \sum_{k=2}^{n-1} ak\lg k + \Theta(n)$

Randomized Quicksort $\rightarrow \Theta(n.\log n)$

$= \frac{2a}{n}\left(\frac{1}{2}n^2\lg n - \frac{1}{8}n^2\right) + \Theta(n)$

$= an\lg n - \left(\frac{an}{4} - \Theta(n)\right)$

$\leqslant an\lg n$

if $an/4$ dominates $\Theta(n)$

3) It can't be seen from the chart easily but

$\frac{T_R(1k)}{T_D(1k)} > \frac{T_R(10k)}{T_D(10k)}$ ..... and so on when n incerase they both merge to each other if this case contine to happen.

Chart is at the end.

4) For deterministic quicksort sorted tweets was the worst case. 100k and 1m tweets couldn't be caldllated. It was $O(n^2)$. However randomized quicksort has $\Theta(n.\log n)$. It sorted tweet performance was nearly same with unsorted data. It's chart at the end. I would chose randomized in each case because I assume that sorted or unsorted we don't have any information about the array.

5) Dual pivot → Best case $T(n) = 3T(n/3) + \Theta(n)$

master theorem

$n^{\log_3 3} = n \rightarrow \Theta(n.\log n)$

Worst case → $T(n) = T(1) + T(n-2) + T(1) + \Theta(n)$

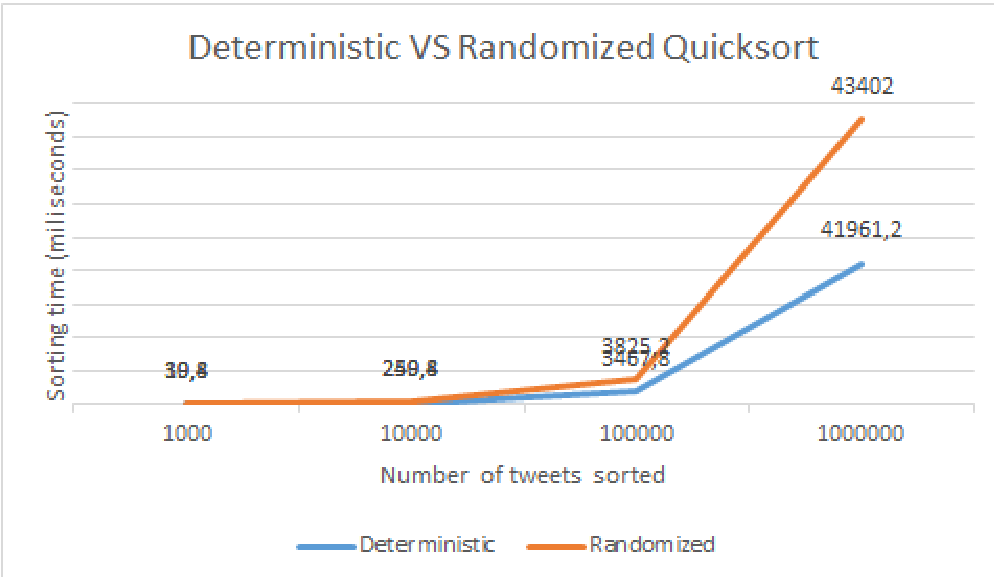
$= O(n^2) \leftarrow \sum_{k=3}^{\left(\frac{n-1}{2}\right)+2} (k-2) \Big\}$ tree

Asimptomatic bounds are same but dual pivot is ussualy faster because of memory reaching is slower than processing.

		time					
	deterministic	1	2	3	4	5	**average**
N	1k	18	19	20	19	23	19,8
	10k	250	235	250	235	234	240,8
	100k	3452	3467	3468	3484	3468	3467,8
	1m	42027	41902	42037	41850	41990	41961,2

		time					
	randomized	1	2	3	4	5	**average**
N	1k	30	29	30	31	32	30,4
	10k	250	250	265	262	270	259,4
	100k	3843	3795	3811	3837	3840	3825,2
	1m	43178	43948	43240	43396	43248	43402

	1k	10k	100k	1m
deterministic	19,8	240,8	3467,8	41961,2
randomized	30,4	259,4	3825,2	43402
r/d	1,535354	1,077243	1,103062	1,034336



		time					
	deterministic	1	2	3	4	5	**average**
N	1k	771	763	771	773	778	771,2
	10k	79395					79395
	100k	NA	NA	NA	NA	NA	NA
	1m	NA	NA	NA	NA	NA	NA

	1k	**10k**	**100k**	**1m**
deterministic	771	79395	NA	NA
randomized	20	267	3346	46973
r/d	0,02594	0,003363	NA	NA

		time					
	randomized	1	2	3	4	5	average
N	1k	20	19	20	18	23	20
	10k	267					267
	100k	3346					3346
	1m	46973					46973