Analysis of Algorithm Homework-1

Elif Arıkan - 150180010

November 25, 2021

Q1

Time taken by QuickSort, in general, can be written as following.
$T(n) = T(|L|) + T(|R|) + \Theta(n)$.
The T(L) includes elements smaller than pivot, T(R) includes elements bigger than pivot. Two terms are for two recursive calls, the last term is for the partition process.

Worst Case of Deterministic Quick Sort

The worst case occurs when the partition process always picks greatest or smallest element as pivot.
$T(n) = T(0) + T(n-1) + \Theta(n)$ which is equivalent to $T(n) = T(n-1) + \Theta(n)$.
We solve this equation with substitution method. And obtain $\Theta(n^2)$.

Best Case of Deterministic Quick Sort

The best case occurs when the partition process always picks the middle element as pivot. Following is recurrence for best case.
$T(n) = 2T(n/2) + \Theta(n)$.
The solution of above equation is $\Theta(nlogn)$.

Q2

$E[|L|] = E[|R|] = \frac{n-1}{2}$
The expected number of items on each side of the pivot is half of the things. Whether or not a,b are compared is a random variable, that depends on the choice of pivots. Let's say $X_{a,b} = \left\{ \text{1 is ever compared ,0 is never compared.} \right.$

$$\sum_{a=1}^{n} \sum_{b=a+1}^{n} X_{a,b} \text{ is the number of comparisons total during the algorithm .}$$

$$E\left[\sum_{a=1}^{n} \sum_{b=a+1}^{n} X_{a,b}\right] = \sum_{a=1}^{n} \sum_{b=a+1}^{n} E[X_{a,b}]$$

$E[X_{a,b}] = P(X_{a,b} = 1) * 1 + P(X_{a,b} = 0) * 0 = P(X_{a,b} = 1)$.
So we need to figure out P($X_{a,b} = 1$)(probability that a and b are ever compared). P($X_{a,b} = 1$) $= \frac{2}{b-a+1}$

Expected number of comparison is $\sum_{a=1}^{n} \sum_{b=a+1}^{n} \frac{2}{b-a+1}$

We get that this is less than 2nln(n). It's mean that $\Theta(nlogn)$.

Q3

When we examine the data in the graph (figure 2), our quick sort algorithm fits nlogn, as we have calculated before.

Q4

While implicating the deterministic partition, we provide the worst case because we take the last element as the pivot every time. It's like sorting an array sorted from largest to smallest, and it is fixed to n^2 as we can see in the graph(figure 3 and figure 4).
Randomized partition again fits to nlogn.

Q5

Dual pivot quick sort is a little bit faster than the original single pivot quick sort. But still, the worst case will remain $\Theta(n^2)$ when the array is already sorted in an increasing or decreasing order.

	1000	10000	100000	1000000
Deterministic	11,917	155,174	1931,88	23060,1
Randomized	9,677	122,202	1554,59	19078,4

Figure 1: table of deterministic and randomized sorting runtime(ms)

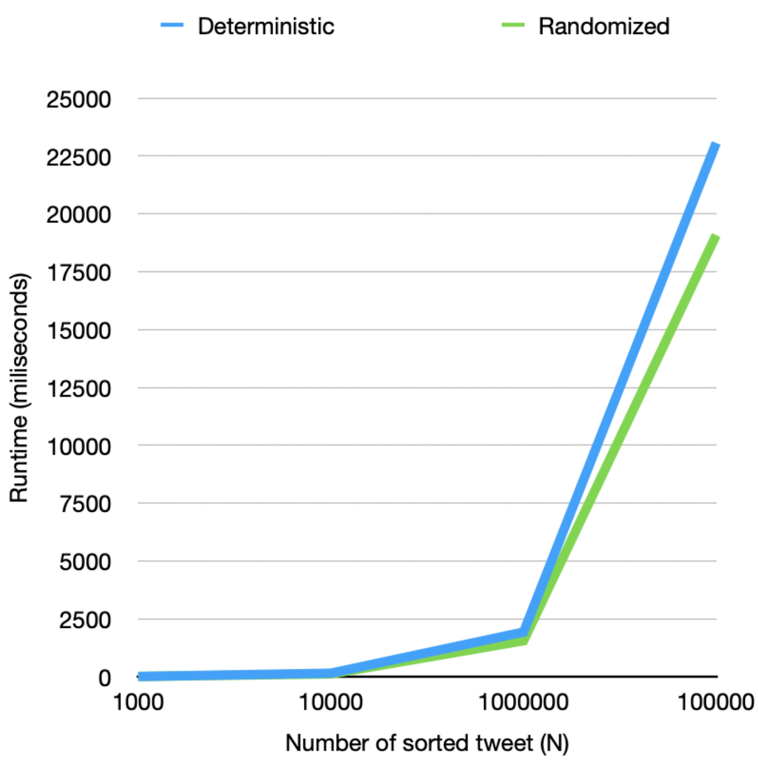
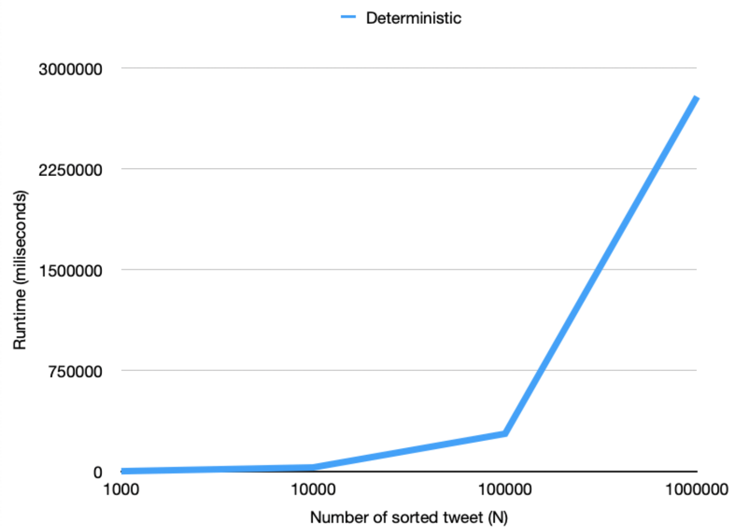


Figure 2: graph of deterministic and randomized sorting runtime(ms)



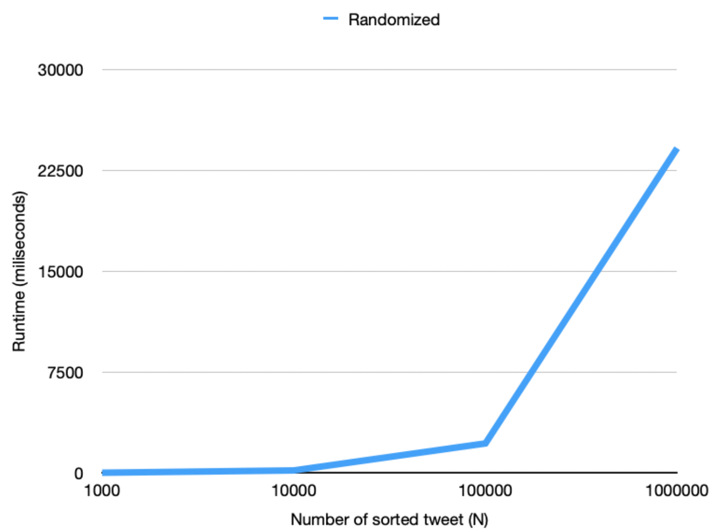


Figure 3: graph of deterministic and randomized sorting runtime(ms)

	1000	10000	100000	1000000
Deterministic	280,662	28637,7	279453,2	2786598,553
Randomized	13,152	183,889	2192,81	24136,5

Figure 4: table of deterministic and randomized sorting runtime(ms)