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

First, picking a pivot and computing it with each element of an array which takes time complexity of $\Theta(n)$.

The pivot partition of the array to subarray of sizes k and n-k-1. we assume that the pivot divides the array into somewhere equal halves making the recurrence relation to

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

for average case we expect the value of k roughly n/2,

$T(n) = T(n/2) + T(n/2) + \Theta(n)$

$T(n) = 2T(n/2) + \Theta(n)$

Using Master Theorem,

$T(n) = aT(n/b) + \Theta(n^d)$

$a=2 \quad b=2 \quad d=1$

Since, this is the case of $a = b^d$
we have,

$\Theta(n^d lg(n))$

So,
$T(n) = \Theta(n^1 lg(n))$

$T(n) = \Theta(nlogn)$