Drew Wates

CS472

1. The outer loop runs n times

   The middle loop runs i times for each iteration of the outer loop where i takes values from 1 to n.

   The innermost loop runs i times for each iteration of the middle loop

   i = 1, k runs from 1 to 2 (j to i+j) = 2 iterations.

   i = 2, k runs from 2 to 5 = 4 iterations.

   i = 3, k runs from 3 to 7 = 5 iterations

   Total number of iterations: (n*(n+1))/2

   The time complexity is $O(n^2)$ due to the innermost loops having i iterations, and i goesup to n in the middle loop.

2. Prove $f(n) \in O(n^3)$

   $f(n) = n^2 + 3n^3 <= n^3 + 3n^3 = 4n^3$

   Let $c_1 = 4$.

   For n >= 1, we have $f(n) <= 4n^3$.

   $f(n) \in O(n^3)$

   Now WMS $f(n) \in \Omega(n^3)$

   $f(n) = n^2 + 3n^3 >= n^3$

   Let $c_2 = 1$.

   For n >= 1, we get $f(n) >= n^3$.

   $f(n) \in \Omega(n^3)$

   $f(n) \in O(n^3)$ and $f(n) \in \Omega(n^3)$

   $f(n) \in \Theta(n^3)$

3. Let $a_n = a$ for all n

$a_{n+1} = a$

$0 <= c_1 * a <= a$ for all $n >= n_0$

$a <= c_2 * a$ for all $n >= n_0$

$c_1 = c_2 = 1$ and $n_0 = 1$

$0 <= a <= a$ for all $n >= 1$

$a <= a$ for all $n >= 1$

$a_{n+1} = a$ is in $\Theta(a_n)$ when a is constant

The specific properties of the sequence and relationship between consecutive terms need to be considered when determining Θ relationships. So this result does not generalize arbitrary non-constant sequences.

4. The order of this algorithm in the worst case is $O(n^2)$. The recursion uses a matrix with the size of (n - 1) x (n - 1), then (n - 2) x (n - 2). This keeps going until the base case is reached with n = 1.
5. This is a graph that shows the average run times of the ‘sorted()’ and ‘quicksort’ algorithms along with the function f(n) = n * log(n). As can be seen by this graph, the quicksort algorithm stays constant no matter the data set size. f(n) = n * log(n) is shown on the graph to stay constant from the 10^1 data set size to the 10^2 size. When the data set size increases to 10^3, so will the average run time. Finally, when the data set size approaches 10^4, the run time skyrockets dramatically as it goes from 10000 to 60000 all the way up to over 120000.