CS345 Homework 2

Parker Gabel

February 7, 2019

1 It is in $O(n^2)$ and $\Omega(n)$

Let $n_0 = 1$ and $c = 1$.

1.1 Notice for all $n \geq n_0$, if n is odd then $f(n) = n \leq n^2$ and if n is even then $f(n) = n^2 = n^2$ so $f(n) \in O(n^2)$.

1.2 Notice for all $n > n_0$, if n is odd then $f(n) = n = n$ and if n is even then $f(n) = n^2 > n$.

2 Yes. Since big theta is an equvalence relation, it is reflexive. For every function f(n), $f(n) \in \Theta(f(n))$

3 This is reflexive, symmetric, and transitive

3.1 Claim: $f(n) \in \Theta(f(n))$

3.1.1 Subclaim: $f(n) \in O(f(n))$

Let n_0 be any positive integer and c be 1. Notice that for all $n \geq n_0$, $f(n) \leq f(n)$ (in fact it is strictly equal). So $f(n) \in O(f(n))$.

3.1.2 Subclaim: $f(n) \in \Omega(f(n))$

Let n_0 be any positive integer and c be 1. Notice that for all $n > n_0$, $f(n) \geq f(n)$ (in fact it is strictly equal). So $f(n) \in \Omega(f(n))$.

S ince f(n) is in both $O(f(n))$ and $\Omega(f(n))$, it is $\Theta(f(n))$.

3.2 Claim: If $f(n) \in \Theta(g(n))$ then $g(n) \in \Theta(f(n))$

Assume $f(n) \in \Theta(g(n))$. So $f(n) \in O(g(n))$ and $f(n) \in \Omega(g(n))$. So there is some n_0 and c such that for all $n \geq n_0$ $c_1\ g(n) \leq f(n) \leq c_2\ g(n)$ for some c_1 and c_2. So $f(n) \leq c_2\ g(n) = 1/c_2\ f(n) \leq g(n)$. So $c_1\ g(n) \leq f(n) = g(n) \leq 1/c_1\ f(n)$. So $1/c_2\ f(n) \leq g(n) \leq 1/c_1\ f(n)$. So $g(n) \in \Theta(f(n))$

3.3 Claim: If $f(n) \in \Theta(g(n))$ and $g(n) \in \Theta(h(n))$ then $f(n) \in \Theta(h(n))$.

Assume $f(n) \in \Theta(g(n))$ and $g(n) \in \Theta(h(n))$. So there exists a,b,c,d > 0 and $n > 0$ such that $ag(n) \leq f(n) \leq bg(n)$ and $ch(n) \leq g(n) \leq dh(n)$. So $f(n) \geq ag(n) \geq a(ch(n)) = ac(h(n))$ and $f(n) \leq bg(n) \leq b(dh(n)) = bd(h(n))$. So $ac(h(n)) \leq f(n) \leq bd(h(n))$. So $f(n) \in \Theta(h(n))$.

4 2, $\log_3(n)$, $\log_2(n)$, $n^{2/3}$, 20n, $4n^2$, 3^n, n!

5 100n, 10n, n, 2^{100n}

6

6.1 lim $\log(n^2)/(\log(n) + 5) = 2$. Since this is a constant $f(n) = \Theta(n)$

6.2 lim $(n\log(n) + n)/\log(n) = \infty$. So f(n) grows faster so $f(n) \in \Omega(g(n))$

7

7.1 $\Theta(n^2)$. The loop must run n*n times.

7.2 $\Theta(n \log(n))$. The outer loop runs n times and the inner loop runs log(n) times

7.3 $\Theta(n \log(n))$. The outer loop runs log(n) times and the inner loop runs n times

7.4 $\Theta(n^2 \log(n))$. The outer loop is run n times and the inner loop costs n log(n).

7.5 $\Theta(n^2)$. For each time the outer loop is run, the inner loop happens a random amount of times but it is quranteed to run i times for each value of i from 1 to n.

8

```
/** @return The position of an element in sorted array A
with value k. If k is not in A, return A.length. */
static int binary(int[] A, int k) {
    if (A[0] > k) {
return ERROR
    }
    int l = -1;
    int r = A.length; // l and r are beyond array bounds
    while (l+1 != r) { // Stop when l and r meet
 int i = (l+r)/2; // Check middle of remaining subarray
 if (k < A[i]) r = i; // In left half
 if (k == A[i]) return i; // Found it
 if (k > A[i] && k < A[i + 1]) return i + 1; // k not in array
 else: l = i // In right half
     }
     return A.length; // Search value not in A
  }
```

9

9.1 n > DE / (P + E), E = 1 and P = 4 and D = 30. So n > 30 / 5 = 6. The break even point is six. When n is less than 6 then the linked list requires less space.

9.2 n > DE / (P + E), E = 32 and P = 4, and D = 40. So n > 32*40 / 36 = 1280 / 36 = 320 / 9 = 35.55. So the break even point is 35. When n is less than 35 then the linked list requires less space.

10

```
E x;
for(int i = 0; i < Q.length(); i++) {
```

```
    x = Q.dequeue();
    S.push(x);
}
 for(int i = 0; i < S.length(); i++) {
    x = S.pop();
    Q.enqueue(x);
}
```