CS 385 Hw 1b: Analysis of Algorithms

Name: Harris Spahic Date: 9/19/21 (Pledge: "I pledge my honor I have abided by the Stevens Honor System")

1. Upper bound of $f(n) = n^4 + 10n^2 + 5 \rightarrow f(n) \in O(n^4)$

Proof: $n^4 + 10n^2 + 5 \leq n^4 \cdot c$ Let $c = 2$

$n^4 + 10n^2 + 5 \leq n^4 \cdot 2$

$\rightarrow 10n^2 + 5 \leq n^4 \rightarrow$ if $n = 3$

$\rightarrow 10n^2 + 5 \leq 81$ FALSE

$10(16) + 5 \leq 256 \rightarrow$ if $n = 4$ $\boxed{C = 2, n = 4}$

$160 + 5 \leq 256 \rightarrow$ TRUE

2. Asymptotic bound of $f(n) = 3n^3 - 2n \rightarrow f(n) \in \Theta(n^3)$

Proof: $O(n^3)$

$3n^3 - 2n \leq n^3 \cdot c_1$ Let $c_1 = 3$

$\rightarrow 3n^3 - 2n \leq 3n^3 \quad \forall n_0$

$\Omega(n^3)$

$n^3 \cdot c_2 \leq 3n^3 - 2n$ Let $c_2 = 2$

$n^3 \cdot 2 \leq 3n^3 - 2n$

$\rightarrow 0 \leq n^3 - 2n$

$\rightarrow 2 \leq n^2$ $n_0 = 2$

$\rightarrow \boxed{C_1 = 3, C_2 = 2, n_0 = 2}$

3. Is $3n-4 \in \Omega(n^2)$? No

Proof: Assume $3n-4 \in \Omega(n^2)$

$\rightarrow \exists c, n_0$ s.t $\forall n \geq n_0 \mid n \in \mathbb{Z}_+ \rightarrow 3n-4 \geq n^2 \cdot c$

But notice $3n-4 \geq n^2$

$\rightarrow \frac{3n}{n^2} - \frac{4}{n^2} \geq 1$ as $\lim_{n \to \infty}$

$\rightarrow 0 \geq 1$ which is a contradiction,

Thus $3n-4 \notin \Omega(n^2)$ ■

4. Increasing

$O(1)$	1
$O(\lg n)$	2
$O(n)$	3
$O(n\lg n)$	4
$O(n^2)$	5
$O(n^2\lg n)$	6
$O(n^3)$	7
$O(2^n)$	8
$O(n!)$	9
$O(n^n)$	10

5. $f(n) = n$, $t = 1s \rightarrow 10^3$ calculations

b. * $f(n) = n\lg n$, $t = 3600s$

$(3.6\times10^6) = n\lg(n) \rightarrow \boxed{n = 204094}$

c. $f(n) = n^2$, $t = 3.6\times10^6$ ms

$\sqrt{(3.6\times10^6)} = n \rightarrow \boxed{n = 1897}$

d. $f(n) = n^3$, $t = 3.6\times10^6 \times 24 = 8.64\times10^7$ ms

$(8.64)^{1/3} = n \rightarrow \boxed{n = 442}$

e. $f(n) = n!$, $t = 6.0\times10^4$ ms

$6\times10^4 = n! \rightarrow \boxed{n = 8}$

Can you solve 5b numerically?

6. $4n^3 \leq 64n\lg 2$ → $n=1$ $1 \leq 0$ ✗

$n^2 \leq 16\lg n$ → $n=2$ $4 \leq 16$ ✓

$\boxed{\text{Thus } \forall n \geq 1 \mid n \in \mathbb{Z}_+ \quad 4n^3 \leq 64\lg n}$

6b. Step 1) If Alg 1 is "beating" Alg 2, it runs for less time.

Thus find n s.t $4n^3 \leq 64 \lg n$

Step 2) Simplify. $n^2 \leq 16 \lg n$

Step 3) Start with $n = 1$, & increment until true.

$n^2 \leq 16 \lg(n)$ $n=1$ $\rightarrow$ $1 \leq 0$ x

$n^2 \leq 16 \lg(n)$ $n=2$ $\rightarrow$ $4 \leq 16$ ✓

$\rightarrow \{\forall\ n \geq 1 \mid n \in \mathbb{Z}_+\}$ for $4n^3 \leq 64 \lg n$

7. setting count = 0 → c_1

loop 1 → $(n/2)$

loop 2 → $\lg n$

count++ → c_2

return → c_1

→ $\boxed{\text{total time} = 2c_1 + \frac{n}{2} \lg n\, c_2 \in \Theta(\frac{n}{2} \lg n)}$

7b. st count = 0 → c_1

loop 1 → $n^{1/3}$

count ++ → c_2

return → c_1

→ $\boxed{\text{total time} = 2c_1 + n^{1/3} c_2 \in \Theta(n^{1/3})}$

7c. st count = 0 → c_1

loop 1 → n

loop 2 → n

loop 3 → n

count ++ → c_2

return → c_1

$\boxed{\text{total time} = 2c_1 + n^3 c_2 \in \Theta(n^3)}$

7d. count = 0 → c_1

loop 1 → n

loop 2 → 1

c++ → c_2

break → c_2

return → c_1

$$\boxed{\text{total time} = 2c_1 + c_2 n \in \Theta(n)}$$

7e. count = 0 → c_1

loop 1 → n

c++ → c_2

loop 2 → n

c++ → c_2

return → c_1

$$\boxed{\text{total time} = 2c_1 + 2c_2 n \in \Theta(n)}$$