EECS 340: Assignment 2

Shaochen (Henry) ZHONG, `sxz517`
Zhitao (Robert) CHEN, `zxc325`

Due and submitted on 02/03/2020
EECS 340, Dr. Koyutrk

Problem 1

(a) $max\{f(n), g(n)\} = \Theta(f(n) + g(n))$

Since it is known that $f(n) \geq 0$, $g(n) \geq 0$, and $c > 0$; we must have:

$$
\begin{aligned}
f(n) &\leq f(n) + g(n) \\
g(n) &\leq f(n) + g(n) \\
\Rightarrow \max(f(n), g(n)) \in O(f(n) + g(n)) \quad &\text{for } \begin{cases} c = 1 \\ \forall n_0 \in \mathbb{R} \end{cases}
\end{aligned}
\tag{1}
$$

Since it is also known that $f(n) + g(n) \leq 2 \cdot \max(f(n), g(n))$, we may therefore infer:

$$
\max(f(n), g(n)) \in \Omega(f(n) + g(n)) \quad \text{for } \begin{cases} c = \frac{1}{2} \\ \forall n_0 \in \mathbb{R} \end{cases}
\tag{2}
$$

Since both the O- and Ω-notation are established, we may therefore conclude:

$$
\max(f(n), g(n)) \in \Theta(f(n) + g(n))
\tag{3}
$$

(b1) $f(n) + d = O(f(n))$.

For the seek of disambiguation, we rewrite the questioned equation as $f(n) + d = O(f(n))$ by d replacing c for $d > 0$.

According to the defination of O-notation, we have:

$$
\begin{aligned}
f(n) &= O(f(n)) \\
\exists c, n_0 > 0 \;\; \text{s.t.} \;\; 0 \leq f(n) &\leq cf(n) \quad \text{for } n \geq n_0 && (4) \\
\exists n' > 0 \;\; \text{s.t.} \;\; f(n) &\geq f(n') \quad \text{for } n, n' \geq n_0 && (5) \\
\Rightarrow 0 \leq f(n) + d &\leq cf(n) + d \quad \text{for } n \geq n_0 && (6)
\end{aligned}
$$

Due to *Equation 5*, we may rewrite *Euqation 6* as:

$$0 \leq f(n) + d \leq (c + \frac{d}{f(n)})f(n) \tag{7}$$

$$\Rightarrow 0 \leq f(n) + d \;\; \leq \;\; c'f(n) \quad \text{for} \begin{cases} c' = c + \frac{d}{f(n')} \\ n, n' \geq n_0 \end{cases} \tag{8}$$

Based *Euqation 8*, we may conclude $f(n) + d = O(f(n))$ via direct proof.

(b2) If $f(n) \geq 1$, then $f(n) + c = O(f(n))$.

Please refer to proof at **b1** as it provides a broader proof base on $f(n)$ regardless $f(n) \geq 1$ or not.

(c1) If $f(n) = O(g(n)), \log(f(n)) \geq 0$ and $\log(g(n)) \geq 0$, then $log(f(n)) = O(\log(g(n))$.

According to the defination of O-notation, we must have:

$$\exists c, n_0 > 0 \;\; \text{s.t.} \quad f(n) \leq cg(n) \quad \text{for } n \geq n_0 \tag{9}$$

Since it is given that $\log(f(n)), \log(g(n)) \geq 0$, thus we must have:

$$
\begin{aligned}
\log(f(n)) &\leq \log(c(g(n))) \quad \text{for } n \geq n_0 \\
\Rightarrow \log(f(n)) &\leq \log c + \log(g(n)) \quad \text{for } n \geq n_0 && (10)
\end{aligned}
$$

As c, n_0 are constants, there must be a constant c' s.t.

Case 1 Assume $\log(g(n_0)) \neq 0$:

$$
\begin{align}
c' \geq \frac{\log c}{\log(g(n_0))} + 1 \tag{11} \\
\Rightarrow (c'-1)\log(g(n)) \;\geq\; (c'-1)\log(g(n_0)) \;\geq\; \log c \quad \text{for } n \geq n_0 \tag{12} \\
\exists c, n_0 > 0 \;\; \text{s.t.} \nonumber \\
\log(f(n)) \;\leq\; \log c + \log(g(n)) \;\leq\; (c'-1)\log(g(n)) + \log(g(n)) \quad \text{for } n \geq n_0 \tag{13} \\
\Rightarrow \log(f(n)) \leq c' \log(g(n)) \tag{14}
\end{align}
$$

Thus we may conclude $log(f(n)) = O(\log(g(n))$ for this case.

Case 2 Assume $\log(g(n_0)) = 0$:

Since $\log(g(n_0)) = 0$, we shall infer that $g(n_0) = 1$. We may arbitrarily pick some constants c, c' where:

$$
\begin{align}
\log c \leq 0 \nonumber \\
\log c \leq (c'-1) \cdot 0 \nonumber \\
\log c \leq (c'-1) \cdot \log(g(n_0)) \tag{15}
\end{align}
$$

Since $n \geq n_0$ by defination, and known that $g(n) \geq 1$ due to $\log(g(n)) \geq 0$; therefore there must be $g(n) \geq g(n_0)$. Putting this into the context of *Equation 10*, we may have:

$$
\begin{align}
\log(f(n)) \leq \log c + \log(g(n)) \quad \text{for } n \geq n_0 \nonumber \\
\Rightarrow \log(f(n)) \leq (c'-1) \cdot \log(g(n_0)) + \log(g(n)) \nonumber \\
\Rightarrow \log(f(n)) \leq (c'-1) \cdot \log(g(n)) + \log(g(n)) \nonumber \\
\Rightarrow \log(f(n)) \leq c' \log(g(n)) \tag{16}
\end{align}
$$

Thus we may conclude $log(f(n)) = O(\log(g(n))$ for this case.

Since both cases reach to the conclusion of $log(f(n)) = O(\log(g(n))$, we have proven the statement to be valid.

(c2) If $f(n) = O(g(n)), \log(f(n)) \geq 0$ and $\log(g(n)) \geq 1$, then $log(f(n)) = O(\log(g(n))$.

According to the defination of O-notation, we must have:

$$
\exists c, n_0 > 0 \;\; \text{s.t.} \quad f(n) \leq cg(n) \quad \text{for } n \geq n_0 \tag{17}
$$

Since it is given that $\log(f(n)) \geq 0$, $\log(g(n)) \geq 1$, thus we must have:

$$\log(f(n)) \leq \log(c(g(n))) \quad \text{for } n \geq n_0$$
$$\Rightarrow \log(f(n)) \leq \log c + \log(g(n)) \quad \text{for } n \geq n_0 \tag{18}$$

As c, n_0 are constants, there must be a constant c' s.t.

$$c' \geq \frac{\log c}{\log(g(n_0))} + 1 \tag{19}$$
$$\Rightarrow (c'-1)\log(g(n)) \;\geq\; (c'-1)\log(g(n_0)) \;\geq\; \log c \quad \text{for } n \geq n_0 \tag{20}$$
$$\exists c, n_0 > 0 \;\text{ s.t.}$$
$$\log(f(n)) \;\leq\; \log c + \log(g(n)) \;\leq\; (c'-1)\log(g(n)) + \log(g(n)) \quad \text{for } n \geq n_0 \tag{21}$$
$$\Rightarrow \log(f(n)) \leq c' \log(g(n)) \tag{22}$$

Thus we may conclude $log(f(n)) = O(\log(g(n))$, the statement is therefore proven to be valid.

(d1) $f(2n) = \Theta(f(n))$

Since it is known that $f(n) \geq 0$, $g(n) \geq 0$, and $c > 0$; we must have a constant $c' > 0$ which satisfy:

Case 1 Assume $f(n) \neq 0$.

$$c' \geq \frac{f(2n)}{f(n)} \tag{23}$$
$$\Rightarrow f(2n) \leq c'(f(n)) \tag{24}$$

Thus we may conclude $f(2n) = O(f(n))$.
Similarly, we may also have a constant $c'' > 0$ which satisfy:

$$c'' \leq \frac{f(2n)}{f(n)} \tag{25}$$
$$\Rightarrow f(2n) \geq c''(f(n)) \tag{26}$$

Thus we may conclude $f(2n) = \Omega(f(n))$.

Since both the O- and Ω-notation are established, we may therefore conclude $f(2n) = \Theta(f(n))$ in this case.

Case 1 Assume $f(n) = 0$ for n.

Thus we must have two constants k_1, k_2 which satisfy:

$$f(2n) = 0 \tag{27}$$
$$k_1(0) \leq f(2n) \leq k_2 f(n) \Rightarrow k_1(0) \leq 0 \leq k_2(0),\ \forall k \in \mathbb{R}^+ \tag{28}$$

Thus we may conclude $f(2n) = \Theta(f(n))$.

Since both cases reach to the conclusion of $f(2n) = \Theta(f(n))$, we have proven the statement to be valid.

(d2) If $f(n) = O(n^c)$, then $f(2n) = O(n^c)$

As It is known from **d1** that $f(2n) = \Theta(f(n))$, which implies $f(2n) = O(f(n))$. Also it is given that $f(n) = O(n^c)$. Together, we shall infer $f(2n) = O(n^c)$ due to the transitivity property of Θ- and O-notations.

(d3) If $f(n) = \Theta(n^c)$, then $f(2n) = \Theta(f(n))$

Please refer to proof at **d3** as it provides a broader proof base on $f(n)$ regardless $f(n) = \Theta(n^c)$ or not.

Problem 2

Answer

$$\frac{1}{n^a} \ll \frac{1}{n^\epsilon} \ll \log_{\frac{1}{\epsilon}}(n) \ \ (\text{when } 0 < \epsilon < \frac{1}{2}) \ll \log(n^\epsilon) \equiv \log(bn) \equiv \log(n^a) \tag{29}$$
$$\equiv \log(n^b) \leq \log_{\frac{1}{\epsilon}}(n) \ \ (\text{when } \epsilon \geq \frac{1}{2}) \ll (\log n)^a \ll n^\epsilon \tag{30}$$
$$\ll a^{\log_a(n)} \equiv \epsilon n \equiv \frac{n}{a} \ll n^a \equiv (n+b)^a \tag{31}$$
$$\ll (n+a)^b \ll n^{a+b} \ll \epsilon^n \ll a^{\epsilon n} \equiv a^n \ll b^n \tag{32}$$

Justification of $\log(n^\epsilon) \equiv \log(bn) \equiv \log(n^a) \equiv \log(n^b)$

The above equations can be rewrite as:

$$\log(n^\epsilon) = \epsilon \log(n) \tag{33}$$
$$\log(n^a) = a \log(n) \tag{34}$$
$$\log(bn) = \log(b) + \log(n) \tag{35}$$
$$\log(n^b) = b \log(n) \tag{36}$$

Where all of them can be generalized as $\Theta(\log(n))$, as it is known that $n^\epsilon \leq bn \leq n^a$ due to the decending of *power(s)*.

Justification of $\log_{\frac{1}{\epsilon}}(n)$ **(when** $0 < \epsilon < \frac{1}{2}$**)** $\equiv \log(n^b) \leq \log_{\frac{1}{\epsilon}}(n)$ **(when** $\epsilon \geq \frac{1}{2}$**)**

It is known that when $\epsilon < \frac{1}{2} \Rightarrow \frac{1}{\epsilon} \geq 2$; when $\epsilon \geq \frac{1}{2} \Rightarrow \frac{1}{\epsilon} \leq 2$. Since it is observable that $\log_x(n) \gg \log_{x'}(n)$ for $x < x'$, such (in)equaility is valid.

Justification of $a^{\log_a(n)} \equiv \epsilon n \equiv \frac{n}{a}$

The above equations can be rewrite as:

$$a^{\log_a(n)} = n^1 = \Theta(n) \tag{37}$$

$$\epsilon n = \Theta(n) \tag{38}$$

$$fracna = \frac{1}{a}n = \Theta(n) \tag{39}$$

Thus, the above equality is justified.

Justification of $a^{\epsilon n} \equiv a^n$

We may rewrite $a^{\epsilon n}$ as the following, since there must be a c for $c \in \mathbb{R}^+$ which satisfy the equality.

$$a^\epsilon \cdot a^n = c \cdot a^n = \Theta(a^n) \tag{40}$$

Since it is known that $a^n = \Theta(a^n)$, these two expressions are considered equivalent.