ADSA Workshop 1 (Solutions)

August 8, 2022

Question 1.0. True or False?

- $n^2 + 10^6 n = \mathcal{O}(n^2)$.
- $n \log(n) = \mathcal{O}(n)$.
- $n \log(n) = \Omega(n)$.
- $\log(n) = o(n)$

Solution. True,

$$\lim_{x \to \infty} \frac{x^2 + 10^6 x}{x^2} = 1 + 10^6 \lim_{x \to \infty} \frac{1}{x} = 1$$

$\square$

Solution. False,

$$\lim_{x \to \infty} \frac{x \log(x)}{x} = \infty$$

$\square$

Solution. True,

$$\lim_{x \to \infty} \frac{x \log(x)}{x} = \infty$$

$\square$

Solution. True,

$$\lim_{x \to \infty} \frac{\log(x)}{x} = \lim_{x \to \infty} \frac{(1/x)}{1} = 0$$

$\square$

Question 1.1. Prove lemma 6 of Mehlhorn and Sanders

$cf(n) = \Theta(f(n))$ for any positive constant c
$f(n) + g(n) = \Omega(f(n))$
If $g(n) = \mathcal{O}(f(n))$ then $f(n) + g(n) = \mathcal{O}(f(n))$
$\mathcal{O}(f(n)) \cdot \mathcal{O}(g(n)) = \mathcal{O}(f(n) \cdot g(n))$

Solution. Recall that

$$\Theta(f(n)) = \mathcal{O}(f(n)) \cap \Omega(f(n))$$

Then clearly $\forall n \in \mathbb{N}$

$$c \cdot f(n) \leq c \cdot f(n) \leq (1+c)f(n)$$

Thus, $c \cdot f(n) = \mathcal{O}(f(n))$ and $c \cdot f(n) = \Omega(f(n))$. □

Solution. Since

$$\forall n \in \mathbb{N} f(n), g(n) \geq 0$$

Then

$$f(n) + g(n) \geq f(n)$$

So by definition of Big-O, $f(n) + g(n) = \Omega(f(n))$ □

Solution. Suppose that

$$g(n) = \mathcal{O}(f(n))$$

Then there exists some $n_0 \in \mathbb{N}$ and $c > 0$, such that, $\forall n \geq n_0$

$$g(n) \leq c \cdot f(n)$$

So,

$$f(n) + g(n) \leq (1+c)f(n)$$

Hence, $f(n) + g(n) = \mathcal{O}(f(n))$. □

Solution. Recall that

$$\mathcal{O}(f(n)) \cdot \mathcal{O}(g(n)) = \{h(n) \cdot k(n) \mid h \in \mathcal{O}(f(n)) \text{ and } \mathcal{O}(g(n))\}$$

There is some abuse of notation here, what we actually need to show is that

$$\mathcal{O}(f(n)) \cdot \mathcal{O}(g(n)) \subseteq \mathcal{O}(f(n) \cdot g(n))$$

Suppose $h(n) \cdot k(n) \in \mathcal{O}(f(n)) \cdot \mathcal{O}(g(n))$. Then there are some $n_0, n_1 \in \mathbb{N}$ and $c_0, c_1 > 0$ such that

$$\forall n \geq n_0, h(n) \leq c_0 f(n) \text{ and } \forall n \geq n_1, k(n) \leq c_1 g(n)$$

So then if we let $N = max\{n_0, n_1\}$

$$\forall n \geq N, h(n) \cdot k(n) \leq (c_0 \cdot c_1)(f(n) \cdot g(n)).$$

Hence, $h(n)k(n) = \mathcal{O}(f(n)g(n))$. □

Question 2. Suppose that $f(n) = \Theta(g(n))$ and $g(n) = \Theta(h(n))$. Show that $h(n) = \Theta(f(n))$.

Solution. By definition of Θ for some $n_0 \in \mathbb{N}$ and $c_1, c_2 > 0$, we have

$$c_1 g(n) \leq f(n) \leq c_2 g(n)$$

We also have some $n_1 \in \mathbb{N}$ and $c_1', c_2' > 0$ such that

$$c_1' h(n) \leq g(n) \leq c_2' h(n)$$

Let $N = max\{n_0, n_1\}$ using the above inequalities we have $\forall n \geq N$.

$$c_1' c_1 h(n) \leq c_1 g(n) \leq f(n)$$

Which implies that

$$h(n) \leq 1/(c_1 c_1') f(n)$$

Likewise we have that,

$$(1/c_2) f(n) \leq g(n) \leq c_2' h(n)$$

Thus,

$$(1/c_2 c_2') f(n) \leq h(n)$$

Therefore, h(n) $= \Theta(f(n))$ □

Question 3. Suppose that $f(n) = \mathcal{O}(g(n))$ and $g(n) = \mathcal{O}(h(n))$. Show that $h(n) = \Omega(f(n))$.

Solution. This is very similar to the questions above. In fact, it follows from it. By definition of Ω we have an $n_0, n_1 \in \mathbb{N}$ and $c_0, c_1 > 0$ such that.

$$f(n) \leq c_0 g(n) \; \forall n \geq n_0$$
$$g(n) \leq c_1 h(n) \; \forall n \geq n_1$$

Again define $N = max\{n_0, n_1\}$. Then using the above we have $\forall n \geq N$.

$$f(n) \leq c_0 g(n) \leq c_0 c_1 h(n) \tag{1}$$

Which implies that,

$$h(n) \geq (1/c_0 c_1) f(n)$$

Therefore, $h(n) = \Omega(f(n))$. □

Question 4. Is it true that a $\Theta(n^2)$ algorithm always takes longer to run that a $\Theta(\log(n))$ algorithm?

Solution. Not in general. When we actually implement an algorithm there are many factors that can impact the run-time of an algorithm. Different computer architectures will impact how an algorithm actually runs. □

Question 5. These are all nice functions (except $f(n) = n!$ and $g(n) = (n+1)!$) so we can apply the limit theorems. Note that most of these require the use of L'Hôpital's rule.

Solution. $f(n) = \sqrt{n}$ and $g(n) = \log(n)$.

$$\lim_{x\to\infty} \frac{\sqrt{x}}{\log(x)} = \lim_{x\to\infty} \frac{1}{2}\sqrt{x} = \infty$$

So $f = \Omega(g)$. □

Solution. $f(n) = 1$ and $g(n) = 2$. Clearly, $f = \Theta(g)$. □

Solution. $f(n) = 1000 \cdot 2^n$ and $g(n) = 3^n$.

$$\lim_{x\to\infty} \frac{1000 \cdot 2^x}{3^x} = 1000 \cdot \lim_{x\to\infty} \left(\frac{2}{3}\right)^x = 0$$

So $f = \mathcal{O}(g)$. □

Solution. $f(n) = 4^{n+4}$and $g(n) = 2^{2n+2}$.

$$\lim_{x\to\infty} \frac{4^{x+4}}{2^{2x+2}} = \lim_{x\to\infty} 2^6 = 64$$

So $f = \Theta(g)$ □

Solution. $f(n) = 5n\log(n)$ and $g(n) = n\log(5n)$.

$$\lim_{x\to\infty} \frac{5x\log(x)}{x\log(5x)} = 5 \lim_{x\to\infty} \frac{(1/x)}{(1/x)} = 5$$

So $f = \Theta(g)$. □

Solution. $f(n) = n!$ and $g(n) = (n+1)!$. Note that,

$$\forall n \in \mathbb{N}, n! < (n+1)!$$

So $f = \mathcal{O}(g)$. However $f \neq \Omega(g)$. As if that were true there would be some $c > 0$ and an n_0 such that for all $n \geq n_0$.

$$n! \geq c(n+1)!$$

Which implies that

$$n \leq \frac{1}{c} - 1.$$

However, this would imply that the natural numbers have some upper bound. The natural numbers are unbounded so this a contradiction. Therefore, $f \neq \Omega(g)$.

□

Question 6. Prove that $n^k = o(c^n)$ for any $k \in \mathbb{Z}$ and $c > 1$.

Solution. Recall that

$$o(c^n) = \{f(n) \mid \forall m > 0, \exists n_0 \in \mathbb{N} \text{ s.t. } \forall n \geq n_0, f(n) < m \cdot c^n\}$$

So we need to find an n_0 such that for all $m > 0$ and $n \geq n_0$ we have

$$n^k < m \cdot c^n.$$

Take the log of both sides.

$$\log(n^k) < \log(m \cdot c^n).$$

Then

$$k \log(n) < \log(m) + n \log(c)$$

Rearranging for n

$$n > (k \log(n) - \log(m)) / \log(c)$$

So given $m > 0$ we must find an n_0 such that the above inequality holds for all $n \geq n_0$. A possible argument as to why the above is true is as follows. Define:

$$\begin{aligned} g(x) &= (k \log(x) - \log(m)) / \log(c) \\ f(x) &= x \\ h(x) &= f(x) - g(x) \end{aligned}$$

Differentiate $h(x)$.

$$h'(x) = 1 - \frac{k}{\log(c)} \frac{1}{x}$$

Then $\forall x > \frac{k}{\log(c)}$ $h'(x) > 0$. Therefore, $h(x)$ is strictly increasing on $[\frac{k}{\log(c)}, \infty)$. Thus there must be some point $x_0 > \frac{k}{\log(c)}$ such that $h(x_0) = 0$. But, since $h(x)$ is strictly increasing there must be some natural number $n_0 \geq x_0$ such that $\forall n \geq n_0$ $h(n) > 0$. Therefore, $f(n) > g(n)$. (Note that this argument is for $k \geq 0$ for $k < 0$, the same argument applies but the interval is $(0, \infty)$ due to the vertical asymptote of $h'(x)$ at $x = 0$).

□