

CSE780: Analysis of Algorithms – Homework 1 (Partial Solutions)

**Problem 1** on Asymptotic notations

*Answer for (a):* False. For example, let  $f(n) = 2n$ ,  $g(n) = n$ , and  $h(x) = 2^x$ . Easy to see that  $f(n) = \Theta(g(n))$ . However,  $h(f(n)) = 2^{2n} = 4^n$ , while  $h(g(n)) = 2^n = o(h(f(n)))$ . Hence  $h(f(n)) \neq O(h(g(n)))$ .

*Answer for (b):* False. (The intuition is that it is possible that there is no relation between  $f(n)$  and  $g(n)$ .) Consider  $f(n) = 2^{n/2}$  and  $g(n) = 2^{n \cdot \sin(n\pi/2)}$ . Easy to check that  $\frac{f(n)}{g(n)} = 2^{n(\frac{1}{2} - \sin(n\pi/2))}$  does not converge at all in the limit as  $n$  approaches  $+\infty$ . Hence  $f(n) \neq o(g(n))$  and  $g(n) \neq o(f(n))$ . Furthermore,  $\frac{f(n)}{g(n)}$  can neither be upper or lower bounded by a positive constant. It then follows that  $f(n) \neq O(g(n))$  and  $f(n) \neq \Omega(g(n))$ .

*Answer for (c):* True. First, it follows from

$$f(n) + g(n) \leq \max\{f(n), g(n)\} + \max\{f(n), g(n)\} = 2 \max\{f(n), g(n)\}$$

that  $f(n) + g(n) = O(\max\{f(n), g(n)\})$ . Next, note that  $f(n) + g(n)$  is larger than or equal to both  $f(n)$  and  $g(n)$  — this is because that  $f(n)$  and  $g(n)$  are positive functions for sufficiently large  $n$  (i.e.,  $f(n)$  and  $g(n)$  are asymptotic positive functions). Thus we have that  $f(n) + g(n) \geq \max\{f(n), g(n)\}$ , implying that  $f(n) + g(n) = \Omega(\max\{f(n), g(n)\})$ . Putting these two together proves that  $f(n) + g(n) = \Theta(\max\{f(n), g(n)\})$ .

**Problem 3** on ordering functions based on their asymptotic growth.

*Answer:*

$$\ln n = \Theta(\log(n^3)), \log(n^3) = o(\log(n!)), \log(n!) = \Theta(n \log n), n \log n = o(4^{\log n}), 4^{\log n} = o((\log n)!).$$

**Problem 4.(b).**  $T(n) = T(\alpha n) + T((1 - \alpha)n) + \beta n$ , where  $0 < \alpha < 1$  and  $\beta > 0$  are constants.

*Answer:* We claim that  $T(n) = \Theta(n \log n)$ . First, we show that  $T(n) = O(n \log n)$ . That is, we wish to show that there exists some constant  $c > 0$  and some integer  $n_0 > 0$  such that  $T(n) \leq cn \log n$  for all  $n \geq n_0$ . We prove the claim by substitution method.

$$\begin{aligned} T(n) &= T(\alpha n) + T((1 - \alpha)n) + \beta n \\ &\leq c\alpha n \log(\alpha n) + c(1 - \alpha)n \log((1 - \alpha)n) + \beta n \\ &= cn \log n + (c\alpha \log \alpha)n + [c(1 - \alpha) \log(1 - \alpha)]n + \beta n \\ &\leq cn \log n. \end{aligned}$$

The last inequality holds when  $c\alpha \log \alpha + c(1 - \alpha) \log(1 - \alpha) + \beta \leq 0$ , that is,

$$c \geq c_1 = \frac{\beta}{\alpha \log \frac{1}{\alpha} + (1 - \alpha) \log \frac{1}{1 - \alpha}} \geq 0.$$

On the other hand, as long as  $c \geq T(2)/2$ , we have that the claim holds for  $n_0 = 2$ . Hence the claim is true for any  $c \geq \max\{T(2)/2, c_1\}$ .

The proof for the lower bound  $T(n) = \Omega(n \log n)$  is similar.

**Problem 4.(c).**  $T(n) = T(n/3) + T(2n/3) + \log n$

*Answer:* We claim that  $T(n) = \Theta(n)$ . First, we show that  $T(n) = O(n)$  by the substitution method. Our goal is to show that there exists some constant  $c > 0$  and some integer  $n_0 > 0$  such that  $T(n) \leq cn - \sqrt{n}$  for all  $n \geq n_0$ . In particular:

$$\begin{aligned} T(n) &= cn/3 - \sqrt{n/3} + 2cn/3 - \sqrt{2n/3} + \log n \\ &= cn - \frac{1 + \sqrt{2}}{\sqrt{3}} + \log n \\ &\leq cn - \sqrt{n}. \end{aligned}$$

The last inequality as long as  $\frac{1+\sqrt{2}-\sqrt{3}}{\sqrt{3}}\sqrt{n} \geq \log n$  which will hold for  $n$  sufficiently large (e.g, for all  $n \geq n_0 = 2^{16}$ ). Furthermore, the base case holds for  $T(n_0)$  as long as  $cn_0 - \sqrt{n_0} \geq T(n_0/3) + T(2n_0/3) + \log n_0$ , that is, for any  $c \geq c_1 = (T(n_0/3) + T(2n_0/3) + \log n_0 + \sqrt{n_0})/n_0$ . (Note that  $T(n_0), T(n_0/3),$  and  $T(2n_0/3)$  are all constant since  $n_0$  is constant.) Thus by induction we ave that  $T(n) \leq cn - \sqrt{n} \leq cn$  for any  $n \geq n_0$  as long as  $c \geq c_1$ .It then follows that  $T(n) = O(n)$ .

The other direction that  $T(n) = \Omega(n)$  can be shown similarly and is simpler, e.g, by arguing that  $T(n) \geq cn$  using substitution method.

**Problem 4.(e).**  $T(n) = \max\{2T(n/2) + 3n, T(n/5) + T(4n/5) + n\}$

*Answer:* We claim that  $T(n) = \Theta(n \log n)$ . First, we show that  $T(n) = O(n \log n)$ . That is, we wish to show that there exists some constant  $c > 0$  and some integer  $n_0 > 0$  such that  $T(n) \leq cn \log n$  for all  $n \geq n_0$ . We prove the claim by substitution method.

$$\begin{aligned} T(n) &= \max\{2T(n/2) + 3n, T(n/5) + T(4n/5) + n\} \\ &\leq \max\{cn \log(n/2) + 3n, \frac{cn}{5} \log(n/5) + \frac{4cn}{5} \log(4n/5) + n\} \\ &= \max\{cn \log n + (3 - c)n, cn \log n + (1 - (8/5 + \log 5)c)n\} \\ &\leq cn \log n. \end{aligned}$$

The last inequality holds as long as  $\max\{3 - c, 1 - (8/5 + \log 5)c\} \leq 0$ . That is,  $c \geq 3$ . On the other hand, the claim holds for  $n_0 = 2$  when  $c \geq T(2)/2$ . As such,  $T(n) \leq cn \log n$  for all  $n \geq 2$  as long as  $c \geq \max\{3, T(2)/2\}$ .

The proof for the lower bound  $T(n) = \Omega(n \log n)$  is similar.

**Problem 4.(f).**  $T(n) = 3T(n/3 + 10) + n/2$

*Answer:* We claim that  $T(n) = \Theta(n \log n)$ . First, we show that  $T(n) = O(n \log n)$ . That is, we wish to show that there exists some constant  $c > 0$  and some integer  $n_0 > 0$  such that  $T(n) < cn \log n$  for all  $n \geq n_0$ . We prove the claim by substitution method.

$$\begin{aligned} T(n) &= 3T(n/3 + 10) + n/2 \\ &\leq cn \log(n/3 + 10) + 30c \log(n/3 + 10) + n/2 \\ &\leq cn \log(n/3) + cn \log 2 + 30c \log n + n/2 & (*1) \\ &\leq cn \log n - cn \log 3 + cn + n & (*2) \\ &\leq cn \log n \end{aligned}$$

The lines (\*1) and (\*2) hold for all  $n \geq n_0$  where  $n_0$  is large enough (i.e,  $n_0$  satisfies that  $n_0/3+10 \leq 2n_0/3$  and  $30 \log n_0 \leq n_0/2$ ). The last inequality holds when  $c \geq 1/\log(3/2)$ . On the other hand, the claim holds for  $n_0$  when  $c \geq \frac{T(n_0)}{n_0 \log n_0}$ . Hence  $T(n) \leq cn \log n$  for all  $n \geq n_0$  as long as  $c \geq \max\{\frac{1}{\log(3/2)}, \frac{T(n_0)}{n_0 \log n_0}\}$ .

The proof for the lower bound  $T(n) = \Omega(n \log n)$  is similar.

**Problem 4.(g).**  $T(n) = 2T(n/2) + n/\log n$

*Answer:* (Note: We can show that  $T(n) = \Theta(n \log \log n)$  by substitution method. For example, to show  $T(n) = O(n \log \log n)$ , we can argue that there exists  $n_0, c > 0$  such that  $T(n) \leq cn \log \log n - n$  for  $n \geq n_0$ . Below, however, I will use the expansion method. )

$$\begin{aligned} T(n) &= 2T(n/2) + n/\log n \\ &= 2[2T(n/4) + (n/2)/\log(n/2)] + n/\log n \\ &= 4T(n/4) + n/\log(n/2) + n/\log n = \dots \\ &= 2^k T(n/2^k) + n/\log(n/2^{k-1}) + n/\log(n/2^{k-2}) + \dots + n/\log(n/2^0) \\ &= 2^k T(n/2^k) + n \sum_{i=0}^{k-1} \frac{1}{\log(n/2^i)} \end{aligned}$$

We stop the expansion when  $n/2^k = 1$ , implying that  $k = \log n$ . We have that

$$\begin{aligned} T(n) &= 2^k T(n/2^k) + n \sum_{i=0}^{k-1} \frac{1}{\log(n/2^i)} \\ &= n + n \sum_{i=0}^{k-1} \frac{1}{\log(2^k/2^i)} \\ &= n + n \sum_{h=1}^k \frac{1}{\log 2^h} \\ &= n + n \sum_{h=1}^k \frac{1}{h} = n + n\Theta(\ln k) \\ &= \Theta(n \log \log n) \end{aligned}$$

**Problem 4.(h).**  $T(n) = T(n-1) + 1/n$

*Answer:*

$$\begin{aligned} T(n) &= T(n-1) + 1/n = T(n-2) + 1/(n-1) + 1/n = \dots \\ &= T(1) + \sum_{i=2}^n 1/i = \Theta(\log n) \end{aligned}$$