Algorithm 2020 Spring: Assignment Week 1

Due on Monday, February 24, 2020

李胜锐 2017012066

Question 1

Prove $2n + \Theta(n^2) = \Theta(n^2)$

Suppose $f(x) = \Theta(n^2)$

According to the definition of $\Theta(g(n))$, we can find $c_1, c_2, n_0 \in R^+$, that $\forall n \geq n_0, 0 \leq c_1 n^2 \leq f(n) \leq c_2 n^2$

Now, let $c_1' = c_1, c_2' = c_2 + 1$ and $n_0' = max\{n_0, 3\}$

Therefore, for $\forall n \geq n_0'$, we can get $n > 3$ and $n^2 > 2n$

So $\forall n \geq n_0', 0 \leq c_1' n^2 \leq f(n) + 2n \leq c_2 n^2 + n^2 = c_2' n^2$

And $2n + \Theta(n^2) = \Theta(n^2)$

Question 2

Prove $\Theta(g(n)) \cap o(g(n)) = \emptyset$

Suppose $f(n) \in \Theta(g(n)) \cap o(g(n))$

So $f(n) \in \Theta(g(n))$ and $f(n) \in o(g(n))$

According to the definition: we can find $c_1, c_2, n_0 \in R^+$, that $\forall n \geq n_0, 0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$; and for any $c > 0, \exists n_1 > 0, s.t. \forall n \geq n_1, 0 \leq f(n) < cg(n)$

Now we first find c_1 and c_2, and let $c = c_1 - 1$

And let $n_2 = max\{n_0, n_1\}$. Thus for $\forall n > n_2$, we have $n > n_0$ and $n > n_1$

Therefore $f(n) < cg(n)$ and $c_1 g(n) \leq f(n) \leq c_2 g(n)$

So $c_1 g(n) < cg(n)$, and $c_1 < c$, which contradicts the assumption

So $f(n)$ dosen't exists, and $\Theta(g(n)) \cap o(g(n)) = \emptyset$

Question 3

Prove $\Theta(g(n)) \cup o(g(n)) \neq O(g(n))$

We let $f(n) = (1 + \sin(\frac{\pi}{2} n))n^2$, and $g(n) = n^2$

Let $c = 3, n_0 = 1$, and $\forall n \geq n_0, 0 \leq f(n) \leq c \cdot g(n)$

So $f(n) \in O(g(n))$

$\exists c_0 = 1, \forall n_1 > 0, \exists n > n_1$, that $\sin(\frac{\pi}{2} n) < 0$, thus $(1 + \sin(\frac{\pi}{2} n))n^2 < c \cdot n^2$

So $f(n) \notin o(g(n))$

$\forall c_1, n_2 \in R^+, \exists n => n_2$ that $n = 4m + 3, m \in N^*, \sin(\frac{\pi}{2} n))n^2 = 0 < c_1 n^2$

So $f(n) \notin \Theta(g(n))$

Question 3 continued on next page. . .

Thus $f(n) \notin \Theta(g(n)) \cup o(g(n))$
And $O(g(n)) \not\subset \Theta(g(n)) \cup o(g(n))$, which means $O(g(n)) \neq \Theta(g(n)) \cup o(g(n))$

Question 4

Prove $\max(f(n), g(n)) = \Theta(f(n) + g(n))$

First we know that $\exists n_0, \forall n > n_0, f(n) > 0, g(n) > 0$
For any $n > n_0, \max(f(n), g(n)) < f(n) + g(n)$
let $c_1 = 0, c_2 = 1$
Thus $\exists n_0 > 0, \forall n > n_0, 0 \cdot f(n) + g(n) < \max(f(n), g(n)) < 1 \cdot f(n) + g(n)$
So $\max(f(n), g(n)) = \Theta(f(n) + g(n))$

Question 5

根据渐进增长率排序

(a) 排序并划分等价类

(每行代表一个等价类)
$g_1 : 2^{2^{n+1}}$
$g_2 : 2^{2^n}$
$g_3 : (n+1)!$
$g_4 : n!$
$g_5 : e^n$
$g_6 : n \cdot 2^n$
$g_7 : 2^n$
$g_8 : (\frac{3}{2})^n$
$g_9 - g_{10} : (\lg n)^{\lg n}; n^{\lg \lg n}$
$g_{11} : (\lg n)!$
$g_{12} : n^3$
$g_{13} - g_{14} : n^2; 4^{\lg n}$
$g_{15} : (\sqrt{2})^{\lg n}$
$g_{16} - g_{17} : n \lg n; \lg(n!)$
$g_{18} - g_{19} : n; 2^{\lg n}$
$g_{20} : 2^{\sqrt{2 \lg n}}$

Question 5 continued on next page. . .

$g_{21} : \lg^2 n$

$g_{22} : \ln n$

$g_{23} : \sqrt{\lg n}$

$g_{24} : \ln \ln n$

$g_{25} : 2^{\lg^* n}$

$g_{26} - g_{27} : \lg^* n; \lg^*(\lg n)$

$g_{28} : \lg(\lg^* n)$

$g_{29} - g_{30} : 1; n^{1/\lg n}$

(b) 给出 $f(n)$ 的一个例子，使其既不不是 $O(g_i(n))$ 也不是 $\Omega(g_i(n))$

$(1 + \sin(\frac{\pi}{2}n))2^{2^{n+2}}$