classfellow

Date: 09/07/2024

Page No.

Algorithms lecture-1
Homeworks

Q1) Prove that $\sqrt{(n+1)^3} = \Omega(n\sqrt{n})$. Hint $\sqrt{ab} = \sqrt{a}\sqrt{b}$, for any nonnegative real numbers a and b.

A1) According to formal definition of Big-Omega, $\sqrt{(n+1)^3} = \Omega(n\sqrt{n})$ if and only if there exists positive constants c and n_0 such that $\sqrt{(n+1)^3} \geq c \cdot n\sqrt{n}$ for all $n \geq n_0$

So,
$$\sqrt{(n+1)^3} \geq c \cdot n\sqrt{n}$$
$$((n+1)^3)^{1/2} \geq n(n)^{1/2}$$
$$(n+1)^{3/2} \geq n^{3/2}$$
$$\left(n\left(1+\frac{1}{n}\right)\right)^{3/2} \geq n^{3/2}$$
$$n^{3/2}\left(1+\frac{1}{n}\right)^{3/2} \geq n^{3/2} \Rightarrow \left(1+\frac{1}{n}\right)^{3/2} \geq 1$$

If n increases, $n^{3/2}(1+\frac{1}{n})^{3/2}$ automatically increases. If we choose $c=\frac{1}{2}$,
$$(n+1)^{3/2} \geq \frac{1}{2}(n^{3/2})$$
Therefore, there exist $c=\frac{1}{2}$, such that $n \geq n_0$, such that $\sqrt{(n+1)^3} \geq c \cdot n\sqrt{n}$ for all $n \geq n_0$.

Therefore,
$$\sqrt{(n+1)^3} = \Omega(n\sqrt{n}) \Rightarrow \sqrt{(n+1)^3} = (n+1)^{3/2} \geq \frac{1}{2} n\sqrt{n}$$

Q2) Prove that if $f_1(n) = O(g_1(n))$ and $f_2(n) = O(g_2(n))$, then $f_1(n) f_2(n) = O(g_1(n) g_2(n))$.

A2) By formal definition of Big-Oh,

$f_1(n) = O(g_1(n))$, there exist positive constants c_1 and n_1, such that $n \geq n_1$ — ① $f_1(n) \leq c_1(g_1(n))$

$f_2(n) = O(g_2(n))$, there exist positive constants c_2 and n_2, such that $n \geq n_2$ — ② $f_2(n) \leq c_2(g_2(n))$.

Multiply ① and ②,

$$f_1(n) f_2(n) \leq c_1(g_1(n)\ c_2(g_2(n))$$

let $c_1 \times c_2 = c$, then

$$f_1(n) f_2(n) \leq c(g_1(n) g_2(n)) \quad — ③$$

Eq ③ becomes formal definition of Big Oh,

let $f(n) = f_1(n) f_2(n)$, $g_n = g_1(n) \times g_2(n)$.

There exist positive constants c and n_0 for max(n_1 and n_2) such that $n \geq n_0$.

$$f_1(n) f_2(n) \leq c(g_1(n) g_2(n)).$$

$$f_1(n) f_2(n) = O(g_1(n) g_2(n))$$

Hence proved.

Date: / / 20

Page No.

classfellow

Q3) Prove that if $f(n) = O(g(n))$ and $g(n) = O(h(n))$, then $f(n) = O(h(n))$.

A3) By formal definition of Big-Oh, $f(n) \le O(g(n))$, there exist positive constant c_1 and n_1 such that for $n \ge n_1$. — ①

Similarly, $g(n) \le c_2(h(n))$. — ②

From ① and ②,

$$f(n) \le c_1(g(n))$$
$$g(n) \le c_2(h(n)).$$

Substitute $cg(n)$ into $f(n)$, it becomes

$$f(n) \le (c_1 c_2) \cdot h(n)$$

let $c = c_1 c_2$,

$$f(n) \le c(h(n))$$

By definition of Big-Oh,

$$f(n) = O(h(n)).$$

There exist constant c and $n_0 = \max(n_1, n_2)$, such that $n \ge n_0$.