Chapter 1

Introduction to algorithm design

n/a

Chapter 2

Algorithm analysis

Notes

The dominance pecking order:

$$n! \gg c^n \gg n^3 \gg n^2 \gg n^{1+\epsilon} \gg n \log n \gg n \gg \sqrt{n} \gg$$
$$\log^2 n \gg \log n \gg \log n / \log\log n \gg \log\log n \gg \alpha(n) \gg 1$$

Solutions

2-10

(a) $f(n) = (n^2 - n)/2,\ g(n) = 6n.$

Is $f(n) = O(g(n))$? If so, there is c such that $f(n) \leq cg(n)$ for sufficiently large n.

$$\frac{1}{2}\left(n^2 - n\right) \leq 6n \ \rightarrow \ n^2 - n \leq 12n \ \rightarrow \ n(n-1) \leq 12n$$

Suppose there is such a c, then

$$n(n-1) \leq 12cn \ \rightarrow \ n - 1 \leq 12c$$

Clearly we can always find n such that this inequality won't hold, so $f(n) \neq O(g(n))$.

Is $g(n) = O(f(n))$? If so, there is c such that $g(n) \leq cf(n)$ for sufficiently large n.

$$6n \leq \frac{1}{2}\left(n^2 - n\right) \ \rightarrow \ 12n \leq n^2 - n = n(n-1) \ \rightarrow \ 12 \leq n - 1 \ \rightarrow \ 13 \leq n.$$

So with $c = 1$ the inequality will hold for $n_0 \geq 13$, and $g(n) = O(f(n))$.

(b) $f(n) = n + 2\sqrt{n},\ g(n) = n^2$.

$f(n) = O(g(n)) \Leftrightarrow f(n) \le cg(n)$ for sufficiently large n.

$$n + 2\sqrt{n} \le cn^2, \text{ with } c = 1,$$
$$n + 2\sqrt{n} \le 2n \text{ for } n > 4,$$
$$2n \le n^2 \text{ so } f(n) = O(g(n)).$$

$g(n) = O(f(n)) \Leftrightarrow g(n) \le cf(n)$ for sufficiently large n. But this asks to find c such that $n^2 \le c\,(n + 2\sqrt{n})$; since ultimately $n^2 \gg n$, $g(n) \ne O(f(n))$.

(c) $f(n) = n \log n,\ g(n) = n\sqrt{n}$.

$$f(n) = O(g(n)) \Leftrightarrow n \log n \le cn\sqrt{n}, \text{ with } c = 1,$$
$$\to \ \log n \le \sqrt{n/2},$$

since $\sqrt{n} \gg \log n,\ f(n) = O(g(n))$.

By the same argument, $g(n) \ne O(f(n))$.

(d) $f(n) = n + \log n,\ g(n) = \sqrt{n} \ \to \ n + \log n \le c\sqrt{n}$, and since $n \gg \sqrt{n}$, any constant factor will be dominated by the linear term, so $f(n) \ne O(g(n))$. Conversely and by the same argument, $g(n) = O(f(n))$.

(e) $f(n) = 2\left(\log n\right)^2,\ g(n) = \log n + 1$. Note that $2\left(\log n\right)^2 = 2\log^2 n$, and $\log^2 n \gg \log n$, so $g(n) = O(f(n))$ and $f(n) \ne O(g(n))$.

(f) $f(n) = 4n\log n + n,\ g(n) = \left(n^2 - n\right)/2$. We know that $n \log n \gg n$, so we can consider just this term from $f(n)$. But ultimately the quadratic term in $g(n)$ dominates so $f(n) = O(g(n))$.

2-11

(a) HOLA