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

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

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

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

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

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

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

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

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

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

(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) $f(n) = 3n^2,\ g(n) = n^2$.

With $c = 3,\ f(n) \leq 3g(n)$ so $f(n) = O(g(n))$.

$f(n) = \Omega(n) \Leftrightarrow cg(n) \leq f(n)$ for sufficiently large n. For $c = 1$ the inequality holds, so $f(n) = \Omega(g(n))$ and $f(n) = \Theta(g(n))$.

(b) $f(n) = 2n^4 - 3n^2 + 7,\ g(n) = n^5$.

$n^5 \gg n^4$ so $f(n) = O(g(n))$ and $f(n) \neq \Omega(g(n))$.

(c) $f(n) = \log n,\ g(n) = \log n + \frac{1}{n}$.

$\lim_{n\to\infty} \frac{1}{n} = 0$, so as $n \to \infty$, $f(n) - g(n) = 0$. So no function dominates the other. Thus, $f(n) = \Theta(g(n))$.

(d) $f(n) = 2^{k \log n},\ g(n) = n^k$.

$$\begin{aligned} f(n) = O(g(n)) &\Leftrightarrow f(n) \leq cg(n) \\ &\rightarrow\ 2^{k \log n} \leq cn^k;\ \text{taking logarithms,} \\ &\rightarrow\ \log\left(2^{k \log n}\right) \leq \log\left(cn^k\right) = \log c + \log n^k \\ &\rightarrow\ k \log n \log 2 \leq \log c + k \log n. \end{aligned}$$

Ignoring constant terms and multiplicative constants, we are left with $\log n \leq \log n$, so $f(n) = \Theta(g(n))$.

(e) $f(n) = 2^n,\ g(n) = 2^{2n}$.

$2^n \leq c2^{2n}$ clearly holds for $c = 1$, so $f(n) = O(g(n))$.

$c2^{2n} \leq 2^n$? Well, $2^{2n} = 2^2 \cdot 2^n = 4 \cdot 2^n$, so $4c2^n \leq 2^n$ is satisfied with $c = 1/4$. So $f(n) = \Omega(g(n))$ and finally, $f(n) = \Theta(g(n))$.