CSE 241 Algorithms and Data Structures **9/15/2011**

Key Facts on Asymptotic Notation

1 Definitions

Let $f(n)$ and $g(n)$ be non-negative functions on the set of positive integers.

- **Defn**: $f(n) = O(g(n))$ iff there exist positive constants c and n_0 such that for every $n \geq n_0$,
$$f(n) \leq cg(n).$$
- **Defn**: $f(n) = \Omega(g(n))$ iff there exist positive constants c and n_0 such that for every $n \geq n_0$,
$$f(n) \geq cg(n).$$
- **Defn**: $f(n) = \Theta(g(n))$ iff $f(n) = O(g(n))$ **and** $f(n) = \Omega(g(n))$, or equivalently, iff there are positive constants c_1, c_2, and n_0 such that for every $n \geq n_0$,
$$c_1 g(n) \leq f(n) \leq c_2 g(n).$$

2 The Limit Test

Let $f(n)$ and $g(n)$ be functions as above.

- If
$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0,$$
then $f(n) = O(g(n))$, but $f(n) \neq \Omega(g(n))$. We say that "$f(n)$ grows asymptotically slower than $g(n)$" and write "$f(n) = o(g(n))$."
- If
$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty,$$
then $f(n) = \Omega(g(n))$, but $f(n) \neq O(g(n))$. We say that "$f(n)$ grows asymptotically faster than $g(n)$" and write "$f(n) = \omega(g(n))$."
- If
$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = c$$
for some constant $c > 0$, then $f(n) = \Theta(g(n))$. We say that "$f(n)$ and $g(n)$ grow asymptotically at the same rate."

3 Conversion between O and Ω

Fact: $f(n) = O(g(n))$ iff $g(n) = \Omega(f(n))$. It follows that $f(n) = \Theta(g(n))$ iff $g(n) = \Theta(f(n))$.