MC215: MATHEMATICAL REASONING AND DISCRETE STRUCTURES

- **Friday, 11/14/08**
  - Today:
    - **More on order of growth**
- **READING**: 4.3
- **EXERCISES**:
  - pp. 207-210: 1-15, and also put all these functions into increasing Θ-order, giving each class a simple Θ-label

Big-Oh, Big-Omega (Ω), Big-Theta (Θ)

- **f(n) = O(g(n)): ∃ C > 0 s.t. |f(n)| ≤ C|g(n)|,** except for finitely many n > 0.
  - Idea: Growth rate of f ≤ Growth rate of g
  - It's a ***partial order*** on Θ-classes of functions with domain $\mathbb{Z}^+$
- **f(n) = Ω(g(n)): ∃ C > 0 s.t. |f(n)| ≥ C|g(n)|,** except for finitely many n > 0.
  - Idea: Growth rate of f ≥ Growth rate of g
  - It's a ***partial order*** on Θ-classes of functions with domain $\mathbb{Z}^+$
- **f(n) = Θ(g(n)): f(n) = O(g(n))** ***and g(n) = O(f(n))***
  - Idea: Growth rate of f ≈ Growth rate of g
  - It's an ***equivalence relation*** on functions with domain $\mathbb{Z}^+$

Little-o and Little-omega (ω)

- These describe "strictly less than" and "strictly greater than" for growth
- **f(n) = o(g(n)):** ***<u>For all</u>*** **C > 0, |f(n)| ≤ C|g(n)|,** except for finitely many n > 0.
  - "For all" means no matter how ***small*** a constant C we use, eventually |f(n)| ≤ C|g(n)|,
  - Idea: Growth rate of f **<** Growth rate of g
- **f(n) = ω(g(n)):** ***<u>For all</u>*** **C > 0, |f(n)| ≥ C|g(n)|,** except for finitely many n > 0.
  - "For all" means no matter how ***small*** a constant C we use, eventually |f(n)| ≥ C|g(n)|,
  - Idea: Growth rate of f **<** Growth rate of g

Growth rates via ratios

- If we can say that, as n gets large ($n \to \infty$):
- $|f(n)/g(n)| \leq C$, then $f(n) = O(g(n))$
- $|f(n)/g(n)| \geq C$, then $f(n) = \Omega(g(n))$
- $C_1 \leq |f(n)/g(n)| \leq C_2$, then $f(n) = \Theta(g(n))$
- $|f(n)/g(n)| \to 0$, then $f(n) = o(g(n))$
- $|f(n)/g(n)| \to \infty$, then $f(n) = \omega(g(n))$

Some important growth classes

- **Power functions:** Sums of n^p, $p \ge 0$, *e.g.* $3n^2 - 2n$, $n^{1/2} + n^{3/2}$
  - *Same order* IFF *highest power is same*
- **Logarithmic functions:** $f(n) = \log_b n$, $b > 1$
  - All the *same order* for any $b > 1$
  - All *lower order* than *any* power function
- **Exponential functions:** $f(n) = a^n$, $a > 1$
  - All *higher order* than any power function
  - $a > b \Rightarrow a^n$ has *higher order* than b^n
- ***Proofs??***

Examples:

- Put in increasing Θ-order, giving each class a simple Θ-label:
  - $f_1(n) = n^2 + n + 1$
  - $f_2(n) = 5 \log_2 n$
  - $f_3(n) = \log_3 n$
  - $f_4(n) = 3^n$
  - $f_5(n) = 2^n$
  - $f_6(n) = 35$
  - $f_7(n) = n + 2$
  - $f_8(n) = (n + 2)^2$
- And how about:
  - $g_1(n) = n \log_3 n$
  - $g_2(n) = 2^n + 3^n$
  - $g_3(n) = 10 n \log_5 n + n$
  - $g_4(n) = n!$
  - $g_5(n) = n^n$
- If $f(n) = O(g(n))$,
  - What is $\Theta(f(n) + g(n))$?