CS 170 Spring 2017 − Discussion 1

Raymond Chan

Asymptotic Analysis

When looking at function behavior, we want to thinkg about how it behaves when the input gets significantly large. We use the following notations O ("Big-Oh"), Ω ("Big-Omega"), and Θ ("Big-Theta"). These refer to function sets rather than runtime specifically.

- $O(\cdot)$
  This is considered an "upper bound". $f(n) \in O(g(n))$ means that the function $f(n)$ belongs in the set of functions that are upper bounded by $g(n)$ when n gets significantly large.
  Mahtematically, the can be refered to as $f(n) \leq c \cdot g(n)$ for some constant c.
  If $f(n) \in O(n^2)$, it also means that $f(n) \in O(n^3)$, $f(n) \in O(2^n)$, and $f(n) \in O(\cdot)$ of any function that upper bounds n^2.

- $\Omega(\cdot)$
  This is a "lower bound". $f(n) \in \Omega(g(n))$ means that $f(n)$ is lower bounded by $g(n)$. For significantly large n, $f(n) \geq c \cdot g(n)$ for some constant c. Like $O(\cdot)$, if $f(n) \in \Omega(n^2)$, then $f(n) \in \Omega(n)$, $f(n) \in \Omega(\log n)$, and any function that lower bounds n^2.

- $\Theta(\cdot)$
  This is a "tight bound". $f(n) \in \Theta(g(n)) \Leftrightarrow f(n) \in \Omega(g(n))$ and $f(n) \in O(g(n))$. This mathematical expression is $c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$ where c_1 and c_2 are constants and $c_1 \leq c_2$.

Here are some rules when dealing with asymptotics

- Remove multiplicative constants and lower order terms.
  e.g. $O(2n^4 + n^2 + n \log n) = O(n^4)$.

- Any exponential dominates any polynomial.
  e.g. $n^2 \in O(2^n)$.

- Any polynomial dominates any logarithm.
  e.g. $n^2 \in \Omega(\log_3 n)$.