Notes 9-13

1 Asymptotic analysis continued

We consider $T(n)$ the worst-case running time of an algorithm. We typically omit clarifying worst-case, as we always consider worst-case runtimes in elementary algorithm analysis.

The big picture: to analyze runtimes we want tools like $\leq$, $\geq$, and $=$, but which ignore constants and small input sizes:

- $T(n) = \mathcal{O}(g(n))$ is an asymptotic upper bound (think $\leq$)
- $T(n) = \Omega(g(n))$ is an asymptotic lower bound (think $\geq$)
- $T(n) = \Theta(g(n))$ is a "tight" asymptotic bound, i.e. a matching asymptotic upper and lower bound (think $=$)

We defined $\mathcal{O}(g(n))$ last time. We will define $\Omega(g(n))$ in a very similar manner, except that it will act as a lower bound instead of an upper bound:

$T(n) = \Omega(g(n))$ means $\exists c, n_0 > 0$ such that $\forall n \geq n_0$, $T(n) \geq c \cdot g(n)$.

Then, towards a notion of equal asymptotic runtimes, we define $T(n) = \Theta(g(n))$ means $T(n) = \mathcal{O}(g(n))$ and $T(n) = \Omega(g(n))$. This is similar to the notion that $a \leq b$ and $a \geq b$ implies $a = b$ for numbers.

Example of proving that something is NOT $\mathcal{O}$ of some function: Claim: $6n^3 \notin \mathcal{O}(n^2)$. Proof: Suppose $6n^3 = \mathcal{O}(n^2)$. That means $\exists c, n_0 > 0$ such that $\forall n \geq n_0$, $6n^3 \leq c \cdot n^2$.

$$\begin{aligned} 6n^3 &\leq c \cdot n^2 \\ 6n &\leq c \\ n &\leq \frac{c}{6} \end{aligned}$$

Therefore this is only true for $n \leq \frac{c}{6}$, not for all $n \geq n_0$, therefore $6n^3 \neq \mathcal{O}(n^2)$.

Example of proving that something is $\Omega(g(n))$: Claim: if $p, q, r > 0$, then $pn^2 + qn + r = \Omega(n^2)$. Proof: Let $c = p$, $n_0 = 1$. Then $pn^2 + qn + r \geq pn^2 = cn^2$ for all $n \geq n_0$.

Further, since last time we proved that $pn^2 + qn + r = \mathcal{O}(n^2)$, it is both upper and lower bounded asymptotically by n^2, so it is $\Theta(n^2)$.

For the following section, note that since our time functions are always parameterized by n, we will omit then n (e.g. we write $T(n)$ as T). These claims are same for Ω notation.

- Transitivity: $f \in \mathcal{O}(g)$ and $g \in \mathcal{O}(h)$, then $f\mathcal{O}(h)$ (for numbers, we know this as: $a \leq b$ and $b \leq c$ then $a \leq c$).
- Transpose symmetry: $f \in \mathcal{O}(g) \Leftrightarrow g \in \Omega(f)$ (for numbers, we know this as: $a \leq b \Leftrightarrow b \geq a$).
- $f \in \mathcal{O}(h_1)$ and $g \in \mathcal{O}(h_2)$ then $f \cdot g \in \mathcal{O}(h_1 \cdot h_2)$ (for numbers, we know this as $a \leq b$ and $c \leq d \Rightarrow ac \leq bd$).
- $f \in \mathcal{O}(h_1)$ and $g \in \mathcal{O}(h_2)$ then $f + g \in \mathcal{O}(h_1 + h_2)$ and $f + g \in \mathcal{O}(\max(h1, h2))$ (for numbers, we know this as $a + b$ and $c + d \Rightarrow a + c \leq b + d$) and $a + c \leq 2 \cdot \max(b, d)$.

  One word of caution: it's possible that two functions f and g do not bound each other at all: i.e., it is possible that $f \notin \mathcal{O}(g)$ and $g \notin \mathcal{O}(f)$. This is different than numbers since it must be that either $a \leq b$ or $b \leq a$. For example, $f(n) = n^2$ if n is even, $f(n) = n$ if n is odd, then $f(n) \notin \mathcal{O}(n)$ and $f(n) \notin \Omega(n)$.

  Another tool for proving asymptotic bounds is the Limit Theorem. To use this, compute $\lim_{n\to\infty} \frac{f(n)}{g(n)}$.

  - If $\lim_{n\to\infty} \frac{f(n)}{g(n)} < \infty$ then $f(n) \in \mathcal{O}(g(n))$.
  - If $\lim_{n\to\infty} \frac{f(n)}{g(n)} > 0$ then $f(n) \in \Omega(g(n))$.
  - If $\lim_{n\to\infty} \frac{f(n)}{g(n)} = c$, then both the above inequalities are true (> 0, and $< \infty$) and so $f(n) \in \Theta(g(n))$.

Claim: $\log(n^2) \in \Theta(\log n + 5)$. Proof: $\lim_{n\to\infty} \frac{f(n)}{g(n)} =^1 \lim_{n\to\infty} \frac{\frac{2}{n}}{\frac{1}{n}} = 2$.

[1] Recall L'Hopital's rule: If f and g are differentiable such that $\lim_{n\to\infty} f(n) = \lim_{n\to\infty} = 0$ or ∞ then $\lim_{n\to\infty} \frac{f'(n)}{g'(n)} = \lim_{n\to\infty} \frac{f(n)}{g(n)}$.