Formal Complexiy Analysis

1 supplement1

Comparing two functions $f(n)$ and $g(n)$

$f(n) \in \Theta(g(n))$ ($f(n)$ is asymptotically equal to $g(n)$)

$$\text{if } 0 < \lim_{n\to\infty} \frac{f(n)}{g(n)} < \infty$$

Example: $f(n) = n^3 + \log(n)$, $g(n) = 1000n^3 + 2n^2$,

$\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0.001$

$f(n) \in O(g(n))$ ($f(n)$ is asymptotically less than or equal to

$g(n))$

$$\text{if } 0 \leq \lim_{n\to\infty} \frac{f(n)}{g(n)} < \infty$$

In previous example, $f(n) \in O(g(n))$ and $g(n) \in O(g(n))$.

$f(n) = logn$, $g(n) = n$.

Here, $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$ and $\lim_{n\to\infty} \frac{g(n)}{f(n)} = \infty$

So, $f(n) \in O(g(n))$, but $g(n) \notin O(f(n))$.

A more technical definition for O notation is that, $f(n) \in O(g(n))$, if there exists a nonnegative constant c and a number n_0, such that for all $n \geq n_0$, $f(n) \leq cg(n)$.

$f(n) \in o(g(n))$ ($f(n)$ is asymptotically strictly less than $g(n)$)

$$\text{if } \lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$$

In previous example, $f(n) \in o(g(n))$.

Refer to Cormen, Leiserson, Rivest, chapter 2 for details

We would often express complexity of an algorithm as $O(f(N))$ where N is the input size, and $f(N)$ is the function expressing the complexity .

2 Supplement 2

1. For every vertex $v \in V$, $d(v) = \infty$

2. d(s) = 0

3. $S \to \phi$

4. While $Q \neq \phi$,

   (a) $v \to$Extract-Min(Q)

   (b) $S \to S \cup \{v\}$

   (c) $Q \to V \setminus S$

   (d) for each vertex $u \in$Adj(v)

   $d(u) \to \min(d(u), d(v) + w(v, u)$

3 Supplement 3

1. For every vertex $v \in V$, $d_0(v) = \infty$

2. $d_0(s) = 0$

3. For $t = 1 \ldots |V| - 1$,

$$d_t(v) = \min_{u: v \in Adj(u)} (d_{t-1}(u) + w(u, v))$$