Data Structures and Algorithm Analysis– Note 1

Feng-Hao Liu

In this note, we summarize some concepts in our lectures.

1 Asymptotic Notation

Definition 1.1 (Big O) *Let $f(n), g(n)$ be two non-negative functions. We say that $f(n)$ is $O(g(n))$ if, there exists constants c, n_0 such that for all $n \geq n_0$, $f(n) \leq c \cdot g(n)$. This is typically denoted as $f(n) = O(g(n))$.*

Example. $3n = O(n)$, $2n^2 + 3n - 6 = O(n^2)$.

Definition 1.2 (Big Ω) *Let $f(n), g(n)$ be two non-negative functions. We say that $f(n)$ is $\Omega(g(n))$ if, there exists constants c, n_0 such that for all $n \geq n_0$, $f(n) \geq c \cdot g(n)$. This is typically denoted as $f(n) = \Omega(g(n))$.*

Example. $0.5n = \Omega(n)$, $2n^2 + 3n - 6 = \Omega(n^2)$.

Definition 1.3 (Big Θ) *Let $f(n), g(n)$ be two non-negative functions. We say that $f(n)$ is $\Theta(g(n))$ if, $f(n)$ is both $O(g(n))$ and $\Omega(g(n))$.*

Example. $n = \Theta(n)$, $2n^2 + 3n - 6 = \Theta(n^2)$.

Definition 1.4 (Little o) *Let $f(n), g(n)$ be two non-negative functions. We say that $f(n)$ is $o(g(n))$ if, for every constant $c > 0$, there exists an integer N_c such that $f(n) \leq c \cdot g(n)$ for all $n \geq N_c$.*

Example. $n = o(n^2)$, $\log n = o(n)$.

Definition 1.5 (Little ω) *Let $f(n), g(n)$ be two non-negative functions. We say that $f(n)$ is $\omega(g(n))$ if, for every constant $c > 0$, there exists an integer N_c such that $f(n) \geq c \cdot g(n)$ for all $n \geq N_c$.*

Example. $n = \omega(\log^n)$, $n^2 = \omega(n)$.

Theorem 1.6 *Let $f(n), g(n)$ be two non-negative functions. $f(n) = o(g(n))$ implies that $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$. Similarly $f(n) = \omega(g(n))$ implies that $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$.*