Introduction to Algorithms

Mong-Jen Kao (高孟駿)

Tuesday 10:10 – 12:00

Thursday 15:30 – 16:20

Asymptotic Notations

To describe the rate for which a function grows.

In a Nutshell…

up to some constant factor.

when n is <u>sufficiently large</u>.

$$f(n) = \begin{cases} \Theta(g(n)) \\ O(g(n)) \\ \Omega(g(n)) \\ o(g(n)) \\ \omega(g(n)) \end{cases}$$

… $f(n)$ grows

<u>exactly</u>

<u>at most</u> at the rate of $g(n)$.

<u>at least</u>

<u>strictly slower</u>

<u>strictly faster</u> than $g(n)$.

The Θ-Notation

- For a given function $g(n)$,
  define $\Theta\big(g(n)\big)$ to be the set of functions

  $$\{\ f(n)\ :\ \text{there exists } \textbf{positive constants}\ c_1, c_2, \text{and } n_0 \text{ such that}$$
  $$0 \le c_1 \cdot g(n) \le f(n) \le c_2 \cdot g(n) \quad \text{for all } n \ge n_0 .\ \}$$

- Intuitively, a function $f(n)$ is contained in $\Theta\big(g(n)\big)$
  if there exists $c_1, c_2 > 0$ such that
  - the value of $f(n)$ is ***always between*** $c_1 \cdot g(n)$ and $c_2 \cdot g(n)$
    when n is ***sufficiently large***.

The Θ-Notation

- Intuitively, a function $f(n)$ is contained in $\Theta(g(n))$ if there exists $c_1, c_2 > 0$ such that
  - the value of $f(n)$ is always between $c_1 \cdot g(n)$ and $c_2 \cdot g(n)$ when n is ***sufficiently large***.

- For example,
  - $(n^2 + n) \in \Theta(n^2)$
  - $(n^2 + n) \notin \Theta(n^3)$

This happens *only when* $f(n)$ and $g(n)$ grow ***roughly at the same speed*** ***when n is sufficiently large***.

Formal Proof

- $(n^2 + n) \in \Theta(n^2)$
  - Pick $c_1 = 1$, $c_2 = 2$, and $n_0 = 2$.
  - Then,

$$n^2 \leq n^2 + n \leq 2n^2 \quad \text{for all } n \geq n_0.$$

  - Hence, $(n^2 + n) \in \Theta(n^2)$.

Deriving the ***<u>Contrapositive Statement.</u>***

- $(n^2 + n) \notin \Theta(n^3)$
  - To prove this statement, we need to show that
    - <u>There exists no positive constants</u> c_1, c_2, n_0 such that

      $$c_1 \cdot n^3 \le n^2 + n \le c_2 \cdot n^3 \quad \text{for all } n \ge n_0.$$
  - This is equivalent of showing the following.
    - <u>For every positive constants</u> c_1, c_2, n_0,

      there always exists some $n \ge n_0$ such that

      $$n^2 + n < c_1 \cdot n^3 \quad \text{or} \quad c_2 \cdot n^3 < n^2 + n\,.$$

Formal Proof

- $\left(n^2 + n\right) \notin \Theta\left(n^3\right)$
  - Consider the following two cases.
    - For every $c_1 \leq 1$, we have

$$n^2 + n < c_1 \cdot n^3 \quad \text{for all} \quad n \geq \frac{2}{c_1}.$$

    - For every $c_1 > 1$, we have

$$n^2 + n < c_1 \cdot n^3 \quad \text{for all} \quad n \geq 2.$$

  - Hence, for every c_1, c_2, n_0,

    there exists some $n \geq n_0$ such that $n^2 + n < c_1 \cdot n^3$.

Deriving the Contrapositive Statement

- Consider the following statement.

P : “ **All of us** are ***potatoes***. “

- Then,

~(not) P : “ **~ (All of us)** are ***potatoes***. “

Which is equivalent to

~ P : “ **One of us** is ***~ (potatoes)***. “

The Θ-Notation

- For a given function $g(n)$,

  define $\Theta\big(g(n)\big)$ to be the set of functions

  $$\{ \ f(n) \ : \ \text{there exists } \textbf{positive constants } \ c_1, c_2, \text{and } n_0 \ \text{ such that}$$

  $$0 \ \leq \ c_1 \cdot g(n) \ \leq \ f(n) \ \leq \ c_2 \cdot g(n) \quad \text{for all } n \geq n_0 \, . \ \}$$

- We will write

  $$f(n) = \Theta\big(g(n)\big)$$

  to denote $f(n) \in \Theta\big(g(n)\big)$.

- This means that $f(n)$ and $g(n)$ are ***roughly at the same order***.

The O-Notation (Big-O)

- For a given function $g(n)$,
  define $O\big(g(n)\big)$ to be the set of functions

$$\{\ f(n)\ :\ \text{there exists } \textbf{positive constants}\ c \text{ and } n_0 \text{ such that}$$

$$0 \le f(n) \le c \cdot g(n) \quad \text{for all } n \ge n_0\,.\ \}$$

- Intuitively, a function $f(n)$ is contained in $O\big(g(n)\big)$
  if there exists $c > 0$ such that
  - the value of $f(n)$ is ***<u>always upper-bounded by</u>*** $c \cdot g(n)$
    when n is ***sufficiently large***.

The O-Notation (Big-O)

- Similarly, we write $f(n) = O\big(g(n)\big)$ if $f(n) \in O\big(g(n)\big)$.
- For example,
  - $\left(n^2 + n\right) = O\left(n^2\right)$
  - $\left(n^2 + n\right) = O\left(n^3\right)$
  - $\left(n^2 + n\right) \neq O(n \log n)$

The Ω-Notation (Big-Omega)

- For a given function $g(n)$,
  define $\Omega\big(g(n)\big)$ to be the set of functions

$$\{\ f(n)\ :\ \text{there exists } \textbf{positive constants}\ c \text{ and } n_0 \text{ such that}$$
$$0 \le c \cdot g(n) \le f(n) \quad \text{for all } n \ge n_0\ .\ \}$$

- Intuitively, a function $f(n)$ is contained in $\Omega\big(g(n)\big)$
  if there exists $c > 0$ such that
  - the value of $f(n)$ is ***<u>always lower-bounded by</u>*** $c \cdot g(n)$
    when n is ***sufficiently large***.

The Ω-Notation (Big-Omega)

- Similarly, we write $f(n) = \Omega(g(n))$ if $f(n) \in \Omega(g(n))$.
- For example,
  - $(n^2 + n) = \Omega(n^2)$
  - $(n^2 + n) = \Omega(n \log n)$
  - $(n^2 + n) \neq \Omega(n^3)$

Try to prove them yourself!

Some Simple Facts

- $f(n) = \Theta(g(n))$

  if and only if $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$.

- $f(n) = \Theta(g(n))$ if and only if

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \Theta(1).$$

- (Transitivity)

  If $f(n) = O(g(n))$ and $g(n) = O(h(n))$, then $f(n) = O(h(n))$.

$f(n) = O(g(n))$ **and** $f(n) \neq \Omega(g(n))$

The o-Notation (Small-O)

- The notation $f(n) = o(g(n))$ is used to denote the situation that the ***asymptotic growth rate*** of $f(n)$ is ***strictly slower*** than that of $g(n)$.

- For a given function $g(n)$,
  define $o(g(n))$ to be the set of functions

$$\{\ f(n)\ :\ \text{for every positive constant } c > 0, \text{ there always exists a constant } n_c > 0 \text{ such that } 0 \le f(n) < c \cdot g(n) \quad \text{for all } n \ge n_0 .\ \}$$

The ω-Notation (Small-Omega)

- The notation $f(n) = \omega(g(n))$ is used to denote the situation that the ***asymptotic growth rate*** of $f(n)$ is ***strictly faster*** than that of $g(n)$.

- For a given function $g(n)$,
  define $\omega(g(n))$ to be the set of functions

$$\{\ f(n) \ :\ \text{for every positive constant } c > 0, \text{ there always exists a constant } n_c > 0 \text{ such that } 0 \leq c \cdot g(n) < f(n) \ \text{ for all } n \geq n_0 .\ \}$$

$$\boldsymbol{f(n) = \Omega(g(n))} \ \textbf{and} \ \boldsymbol{f(n) \neq O(g(n))}$$

The o- and ω-Notations

- For example,
  - $(n^2 + n) = o(n^3)$
  - $(n^2 + n) = \omega(n \log n)$
  - $(n^2 + n) \neq o(n^2)$

Try to prove them yourself!

Some Simple Facts

- $f(n) = o\big(g(n)\big)$

  if and only if $g(n) = \omega\big(f(n)\big)$.

- $f(n) = o\big(g(n)\big)$ if and only if

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

- (Transitivity)

  If $f(n) = o\big(g(n)\big)$ and $g(n) = o\big(h(n)\big)$, then $f(n) = o\big(h(n)\big)$.