Ordine di grandezza delle funzioni

2

L'ordine di grandezza del tempo di esecuzione di un algoritmo, definito nel Capitolo 1, fornisce una semplice caratterizzazione dell'efficienza di un algoritmo e consente anche di confrontare le prestazioni di algoritmi alternativi. Quando la dimensione n dell'input diventa sufficientemente grande, il merge sort, con il suo tempo di esecuzione $\Theta(n \lg n)$ nel caso peggiore, risulta migliore dell'insertion sort, il cui tempo di esecuzione nel caso peggiore è $\Theta(n^2)$. Sebbene si possa talvolta determinare il tempo esatto di esecuzione di un algoritmo, come si è fatto per l'insertion sort nel Capitolo 1, l'estrema precisione non è di solito così importante da compensare lo sforzo del calcolo; infatti per input sufficientemente grandi, le costanti moltiplicative e i termini di ordine più basso dell'esatto tempo di esecuzione sono trascurabili rispetto agli effetti della dimensione stessa dell'input.

Quando si considerano input sufficientemente grandi da rendere rilevante solo l'ordine di grandezza del tempo di esecuzione, si sta studiando l'efficienza ***asintotica*** degli algoritmi. In tal caso ciò che interessa è come cresce il tempo di esecuzione di un algoritmo al crescere all'infinito della dimensione dell'input. Di solito, un algoritmo che sia asintoticamente più efficiente costituisce la scelta migliore per tutti gli input tranne quelli veramente piccoli.

Questo capitolo fornisce alcuni metodi standard per semplificare l'analisi degli algoritmi. Il prossimo paragrafo comincia con la definizione di alcuni tipi di "notazione asintotica", di cui si è già visto un esempio nella notazione Θ; quindi saranno presentate alcune convenzioni notazionali usate nel libro e infine sarà fornita una panoramica sul comportamento delle funzioni che di solito si presentano nell'analisi degli algoritmi.

2.1 Notazione asintotica

Le notazioni usate per descrivere il tempo asintotico di esecuzione di un algoritmo sono definite in termini di funzioni il cui dominio è l'insieme dei numeri naturali $\mathbf{N} = \{0, 1, 2, \ldots\}$. Queste notazioni sono comode per descrivere la funzione $T(n)$ del tempo di esecuzione nel caso peggiore, definita di solito solo su dimensioni intere dell'input. Talvolta è comunque conveniente *usare in modo improprio* la notazione asintotica. Per esempio la notazione è facilmente estesa al dominio dei numeri reali o, alternativamente, ristretta a un sottoinsieme dei numeri naturali. È importante però comprendere il significato preciso della notazione in modo che anche quando sia usata impropriamente, non sia usata male. Questo paragrafo definisce la notazione asintotica di base e introduce inoltre alcuni comuni usi impropri.

Notazione Θ

Nel Capitolo 1 si è determinato che il tempo di esecuzione nel caso peggiore dell'algoritmo insertion sort è $T(n) = \Theta(n^2)$. Questa notazione ha il seguente significato.
Data una funzione $g(n)$, si denota con $\Theta(g(n))$ l'*insieme di funzioni*

$$\Theta(g(n)) = \{ f(n): \text{ esistono tre costanti positive } c_1, c_2 \text{ e } n_0 \text{ tali che } 0 \le c_1\, g(n) \le f(n) \le c_2\, g(n) \text{ per ogni } n \ge n_0 \}.$$

Una funzione $f(n)$ appartiene all'insieme $\Theta(g(n))$ se esistono le costanti positive c_1 e c_2 tali che essa possa essere schiacciata tra $c_1 g(n)$ e $c_2\, g(n)$ per n sufficientemente grande. Malgrado $\Theta(g(n))$ sia un insieme, per indicare che $f(n)$ è membro di $\Theta(g(n))$ si scrive "$f(n) = \Theta(g(n))$", oppure si scrive "$f(n) \in \Theta(g(n))$". Questo uso improprio del segno di uguaglianza per denotare l'appartenenza insiemistica, può sembrare, almeno all'inizio, confusionario, ma si vedrà che ha i suoi vantaggi.

La figura 2.1(a) fornisce una rappresentazione intuitiva delle funzioni $f(n)$ e $g(n)$, dove $f(n) = \Theta(g(n))$. Per ogni valore di n alla destra di n_0, il valore di $f(n)$ coincide o si trova sopra $c_1 g(n)$ e coincide o si trova sotto $c_2 g(n)$. In altre parole, per ogni $n \ge n_0$, la funzione $f(n)$ è uguale a $g(n)$ a meno di un fattore costante. Si dice che $g(n)$ è un ***limite asintotico stretto*** per $f(n)$.

La definizione di $\Theta(g(n))$ richiede che ogni membro $f(n)$ di $\Theta(g(n))$ sia ***asintoticamente non negativo***, cioè che $f(n)$ sia non negativa tutte le volte che n è sufficientemente grande. Di conseguenza, la stessa funzione $g(n)$ deve essere asintoticamente non negativa, altrimenti l'insieme $\Theta(g(n))$ è vuoto. Si assumerà perciò che ogni funzione usata dentro la notazione Θ sia asintoticamente non negativa; questa ipotesi vale anche per le altre notazioni asintotiche definite in questo capitolo. Si dice ***asintoticamente positiva*** una funzione che sia strettamente positiva per tutti gli n sufficientemente grandi.

Nel Capitolo 1 si è introdotta una nozione informale della notazione Θ che consisteva nel tralasciare i termini di grado più basso ed ignorare il coefficiente principale del termine di ordine più grande. Si può brevemente giustificare questa soluzione intuitiva usando la

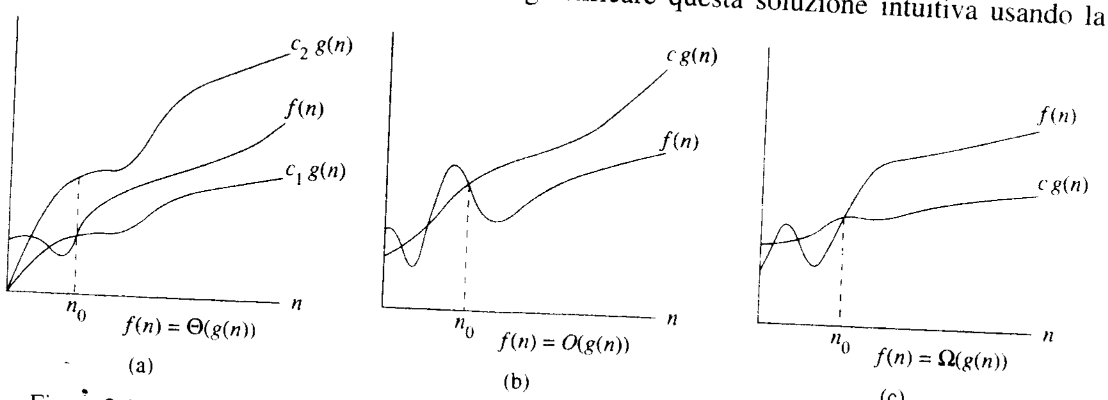

Figura 2.1 *Esempi grafici delle notazioni Θ, O, e Ω. In ogni parte della figura, il valore di n_0 mostrato è il minimo valore possibile; qualunque valore più grande va bene.* **(a)** *La notazione Θ limita una funzione rispetto a fattori costanti. Si scrive $f(n) = \Theta(g(n))$ se esistono delle costanti positive n_0, c_1 e c_2 tali che alla destra di n_0 il valore di $f(n)$ si trova sempre tra $c_1 g(n)$ e $c_2 g(n)$ inclusi.* **(b)** *La notazione O fornisce un limite superiore per una funzione rispetto a fattori costanti. Si scrive $f(n) = O(g(n))$ se esistono delle costanti positive n_0 e c tali che alla destra di n_0 il valore di $f(n)$ è sempre sotto o coincide con $cg(n)$.* **(c)** *La notazione Ω fornisce un limite inferiore per una funzione rispetto a fattori costanti. Si scrive $f(n) = \Omega(g(n))$ se esistono delle costanti positive n_0 e c tali che alla destra di n_0 il valore di $f(n)$ è sempre sopra o coincide con $cg(n)$.*

definizione formale per mostrare che $\frac{1}{2}n^2 - 3n = \Theta(n^2)$. Per far questo si devono determinare le costanti positive c_1, c_2 ed n_0 tali che

$$c_1 n^2 \leq \frac{1}{2}n^2 - 3n \leq c_2 n^2$$

per ogni $n \geq n_0$. Dividendo per n^2 si ha

$$c_1 \leq \frac{1}{2} - \frac{3}{n} \leq c_2 .$$

La disuguaglianza della parte destra può essere resa valida per qualunque valore di $n \geq 1$, scegliendo $c_2 \geq 1/2$. Analogamente, la disuguaglianza della parte destra può essere resa valida per qualunque valore di $n \geq 7$, scegliendo $c_1 \leq 1/14$. Quindi, scegliendo $c_1 = 1/14$, $c_2 = 1/2$ ed $n_0 = 7$, si può verificare che $\frac{1}{2}n^2 - 3n = \Theta(n^2)$. Certamente esistono altre scelte per le costanti, ma la cosa importante è che qualche scelta esista. Si noti che queste costanti dipendono dalla funzione $\frac{1}{2}n^2 - 3n$; una diversa funzione appartenente a $\Theta(n^2)$ di solito richiederà costanti diverse.

Si può usare la definizione formale anche per verificare che $6n^3 \neq \Theta(n^2)$. Si supponga per assurdo che esistano c_2 ed n_0 tali che $6n^3 \leq c_2 n^2$ per ogni $n \geq n_0$. Allora $n \leq c_2/6$, che non può valere in alcun modo per n comunque grande, dato che c_2 è costante.

Intuitivamente, i termini di ordine più basso di una funzione asintoticamente positiva possono essere ignorati nella determinazione del limite asintotico stretto perché essi sono trascurabili per n grande. Una frazione piccola del termine di ordine più alto è sufficiente a dominare i termini di ordine più basso.

Quindi l'assegnamento a c_1 di un valore leggermente più piccolo del coefficiente del termine di ordine più alto e a c_2 di un valore leggermente più grande, consente che le disuguaglianze nella definizione della notazione Θ siano soddisfatte. Il coefficiente del termine più alto può analogamente essere ignorato, in quanto esso cambia c_1 e c_2 solo per un fattore costante uguale al coefficiente.

Come esempio si consideri una generica funzione quadratica $f(n) = an^2 + bn + c$, dove a, b e c sono costanti ed $a > 0$. Tralasciando i termini di ordine più basso ed ignorando la costante si ha $f(n) = \Theta(n^2)$. Formalmente, per mostrare la stessa cosa, si prendono le costanti $c_1 = a/4$, $c_2 = 7a/4$ ed $n_0 = 2\max((|b|/a), \sqrt{(|c|/a)})$. Il lettore può verificare che $c_1 n^2 \leq an^2 + bn + c \leq c_2 n^2$ per ogni $n \geq n_0$. In generale per qualunque polinomio $p(n) = \sum_{i=0}^{d} a_i n^i$, dove le a_i sono costanti e $a_d > 0$, si ha $p(n) = \Theta(n^d)$ (si veda il Problema 2-1).

Dato che qualunque costante è un polinomio di grado 0, si può esprimere qualunque funzione costante come $\Theta(n^0)$ o $\Theta(1)$. Quest'ultima notazione, poiché non è chiaro quale variabile stia tendendo all' infinito, è un uso improprio, comunque di minore importanza[1]. Si userà spesso la notazione $\Theta(1)$ per indicare o una costante o una funzione costante rispetto a qualche variabile.

Notazione O

La notazione Θ limita asintoticamente una funzione da sopra e da sotto. Quando si ha solo un ***limite asintotico superiore***, si usa la notazione O. Per una funzione data $g(n)$, si denota con $O(g(n))$ l'insieme di funzioni

[1] Il problema reale è che là nostra notazione ordinaria per le funzioni non fa distinzione tra funzioni e valori. Nel λ-calcolo, i parametri delle funzioni sono chiaramente specificati: la funzione n^2 potrebbe essere scritta come $\lambda n.n^2$ o anche $\lambda r.r^2$. D'altra parte adottando una notazione più rigorosa si potrebbe complicare la manipolazione algebrica, e così si preferisce tollerare quest'uso improprio.

$O(g(n)) = \{f(n)$: esistono c ed n_0 tali che $0 \le f(n) \le cg(n)$ per ogni $n \ge n_0\}$.

La notazione O si usa per dare un limite superiore ad una funzione a meno di un fattore costante. La figura 2.1 (b) mostra il concetto intuitivo che sta dietro la notazione O; per tutti i valori di n alla destra di n_0 il valore della funzione $f(n)$ coincide o si trova sotto $g(n)$.

Per indicare che una funzione $f(n)$ è membro di $O(g(n))$, si scrive $f(n) = O(g(n))$; si noti che $f(n) = \Theta(g(n))$ implica che $f(n) = O(g(n))$ in quanto la notazione Θ è una notazione più forte della notazione O; per dirlo con la teoria degli insiemi $\Theta(g(n)) \subseteq O(g(n))$. Di conseguenza la dimostrazione che qualunque funzione quadratica $an^2 + bn + c$, dove $a > 0$, è in $\Theta(n^2)$ implica anche che ogni funzione quadratica è in $O(n^2)$. Vale la pena notare che qualunque funzione *lineare* $an + b$ è in $O(n^2)$, il che è facilmente verificabile prendendo $c = a + |b|$ ed $n_0 = 1$.

Qualche lettore che avesse già visto la notazione O può trovare strano che si possa scrivere, per esempio, $n = O(n^2)$. In letteratura, la notazione O è usata talvolta per descrivere informalmente limiti asintoticamente stretti, cioè quelli che sono stati qui definiti usando la notazione Θ.

In questo libro, comunque, quando si scrive $f(n) = O(g(n))$, si sta semplicemente affermando che qualche funzione $g(n)$ è un limite asintotico di $f(n)$ senza affermare quanto il limite superiore sia stretto; la distinzione tra limiti asintotici superiori e limiti asintoticamente stretti è ormai diventata standard nella letteratura riguardante gli algoritmi.

Usando la notazione O si può spesso descrivere il tempo di esecuzione di un algoritmo semplicemente analizzando la struttura complessiva dell'algoritmo; per esempio, per la struttura ciclica doppiamente annidata dell'algoritmo di insertion sort del Capitolo 1, si ha un limite superiore $O(n^2)$ del tempo di esecuzione nel caso peggiore: il costo del ciclo più interno è limitato superiormente da $O(1)$ (costante), gli indici i e j valgono entrambi al più n, e il ciclo più interno è eseguito al più una volta per ognuna delle n^2 coppie di valori di i e j.

Poiché la notazione O descrive un limite superiore, quando la si usa per limitare il tempo di esecuzione di un algoritmo nel caso peggiore, per implicazione si limita anche il tempo di esecuzione dell' algoritmo su input arbitrari. Per cui il limite $O(n^2)$ del tempo di esecuzione di insertion sort nel caso peggiore si applica anche al tempo di esecuzione su qualunque input.

Viceversa, il limite $\Theta(n^2)$ del tempo di esecuzione di insertion sort nel caso peggiore non implica un limite $\Theta(n^2)$ al tempo di esecuzione su qualunque input. Per esempio nel Capitolo 1 si è visto che quando l'input è già ordinato, l'insertion sort viene eseguito in tempo $\Theta(n)$.

Tecnicamente, è un uso improprio dire che il tempo di esecuzione di insertion sort è $O(n^2)$, in quanto, per un dato n, il tempo di esecuzione reale dipende dal particolare input di dimensione n. Cioè, il tempo di esecuzione non è in realtà una funzione solo di n. Ciò che si intende quando si dice "il tempo di esecuzione è $O(n^2)$" è che il tempo di esecuzione nel caso peggiore (che è una funzione di n) è $O(n^2)$ o equivalentemente, qualunque sia il particolare input di dimensione n scelto per ogni valore di n, il tempo di esecuzione su quell'insieme di input è $O(n^2)$.

Notazione Ω

Così come la notazione O fornisce un limite asintotico superiore ad una funzione, la notazione Ω fornisce un ***limite asintotico inferiore***. Per una funzione data $g(n)$, si denota con $\Omega(g(n))$ l'insieme di funzioni

$\Omega(g(n)) = \{f(n)$: esistono c ed n_0 tali che $0 \le cg(n) \le f(n)$ per ogni $n \ge n_0\}$.

Il concetto intuitivo che sta dietro la notazione Ω è mostrato nella figura 2.1(c); per tutti i valori di n alla destra di n_0 il valore della funzione $f(n)$ coincide o si trova sopra $cg(n)$.

Dalle definizioni delle notazioni asintotiche che si sono viste fino adesso, è semplice dimostrare il seguente importante teorema (si veda l'Esercizio 2.1-5).

Teorema 2.1

Per ogni coppia di funzioni $f(n)$ e $g(n)$, $f(n) = \Theta(g(n))$ se e solo se $f(n) = O(g(n))$ e $f(n) = \Omega(g(n))$.

Come esempio di applicazione di questo teorema, la dimostrazione che $an^2 + bn + c = \Theta(n^2)$ per qualunque costante a, b e c, con $a > 0$, implica immediatamente che $an^2 + bn + c = \Omega(n^2)$ e $an^2 + bn + c = O(n^2)$. In pratica, piuttosto che usare il Teorema 2.1 per ottenere limiti asintotici superiori e inferiori dai limiti asintoticamente stretti, come è stato fatto per quest'esempio, generalmente lo si usa per dimostrare limiti asintoticamente stretti partendo da limiti asintotici superiori ed inferiori.

Dato che la notazione Ω descrive un limite inferiore, quando la si usa per limitare il tempo di esecuzione nel caso migliore di un algoritmo, per implicazione si limita anche il tempo di esecuzione dell'algoritmo su input arbitrari. Per esempio il tempo di esecuzione nel caso migliore dell'algoritmo insertion sort è $\Omega(n)$ che implica che il tempo di esecuzione dell'insertion sort è $\Omega(n)$.

Il tempo di esecuzione dell'insertion sort è perciò compreso tra $\Omega(n)$ e $O(n^2)$, in quanto cade ovunque tra una funzione lineare di n ed una funzione quadratica di n. Inoltre, questi limiti sono il più possibile asintoticamente stretti: per esempio, il tempo di esecuzione dell'insertion sort non è $\Omega(n^2)$ in quanto l'insertion sort viene eseguito in tempo $O(n)$ quando l'input è gia ordinato. Non è contraddittorio, tuttavia, dire che il tempo di esecuzione dell'insertion sort nel *caso peggiore* è $\Omega(n^2)$, in quanto esiste un input per il quale l'algoritmo impiega un tempo $\Omega(n^2)$. Quando si dice che il *tempo di esecuzione* (senza ulteriore specificazione) di un algoritmo è $\Omega(g(n))$, si intende che, *a prescindere da quale particolare input di dimensione n sia scelto per ciascun valore di n*, il tempo di esecuzione su quell'insieme di input è almeno una costante moltiplicata per $g(n)$, con n sufficientemente grande.

Notazione asintotica nelle equazioni

Si è già visto come la notazione asintotica possa essere usata all'interno di formule matematiche; per esempio, nell'introdurre la notazione O, si è scritto "$n = O(n^2)$". Si potrebbe anche scrivere $2n^2 + 3n + 1 = 2n^2 + \Theta(n)$. Come interpretare questo tipo di formule?

Quando la notazione asintotica si trova da sola sul lato destro di un'equazione, come in $n = O(n^2)$, il segno uguale, come si è già detto, indica appartenenza: $n \in O(n^2)$. In generale, però, quando la notazione asintotica appare in una formula, la si interpreta come una qualche anonima funzione che non interessa specificare. Per esempio, la formula $2n^2 + 3n + 1 = 2n^2 + \Theta(n)$ significa che $2n^2 + 3n + 1 = 2n^2 + f(n)$, dove $f(n)$ è una funzione nell'insieme $\Theta(n)$. In questo caso, $f(n) = 3n + 1$, che è proprio in $\Theta(n)$.

Usando la notazione asintotica in questo modo si possono eliminare da un'equazione confusioni e dettagli inessenziali. Per esempio nel Capitolo 1 il tempo di esecuzione del merge sort nel caso peggiore è stato espresso con la ricorrenza

$T(n) = 2T(n/2) + \Theta(n)$.

Se si è interessati solo al comportamento asintotico di $T(n)$, non è di alcuna importanza specificare esattamente tutti i termini di ordine più basso; è sottinteso che siano tutti inclusi nella funzione anonima denotata dal termine $\Theta(n)$.

Il numero di funzioni anonime in un'espressione è sottinteso che sia uguale al numero di volte che la notazione asintotica appare. Per esempio, nell'espressione

$$\sum_{i=1}^{n} O(i) ,$$

vi è solo una singola funzione anonima (una funzione di i). Questa espressione quindi *non è* la stessa cosa di $O(1) + O(2) + \ldots + O(n)$, che in realtà non ha una chiara interpretazione.

In alcuni casi, la notazione asintotica appare sul lato sinistro di un'equazione, come in

$$2n^2 + \Theta(n) = \Theta(n^2) .$$

Si interpretano equazioni di questo tipo usando la seguente regola: *indipendentemente da come sono scelte le funzioni anonime nella parte sinistra del segno uguale, vi è un modo per scegliere le funzioni anonime sulla destra del segno uguale per rendere valida l'equazione.* Di conseguenza il significato dell'esempio è che per *qualunque* funzione $f(n) \in \Theta(n)$ vi è una *qualche* funzione $g(n) \in \Theta(n^2)$ tale che $2n^2 + f(n) = g(n)$ per ogni n. In altre parole, il lato destro di un'equazione fornisce un livello di dettaglio più grossolano del lato sinistro.

Varie relazioni di questo tipo possono essere utilizzate insieme a catena come in

$$\begin{aligned} 2n^2 + 3n + 1 &= 2n^2 + \Theta(n) \\ &= \Theta(n^2) . \end{aligned}$$

Si può interpretare ogni equazione separatamente attraverso la regola descritta sopra. La prima equazione dice che vi è *qualche* funzione $f(n) \in \Theta(n)$ tale che $2n^2 + 3n + 1 = 2n^2 + f(n)$ per ogni n. La seconda equazione dice che per *qualsiasi* funzione $g(n) \in \Theta(n)$ (come la $f(n)$ già menzionata) vi è *qualche* funzione $h(n) \in \Theta(n^2)$ tale che $2n^2 + g(n) = h(n)$ per ogni n. Si noti che questa interpretazione implica che $2n^2 + 3n + 1 = \Theta(n^2)$, il che è ciò che la catena di equazioni intuitivamente fornisce.

Notazione *o*

Il limite asintotico superiore fornito dalla notazione O può essere asintoticamente stretto oppure no; il limite $2n^2 = O(n^2)$ è asintoticamente stretto, mentre il limite $2n = O(n^2)$ non lo è. Si usa la notazione o per denotare un limite superiore che non è asintoticamente stretto. Si definisce formalmente $o(g(n))$ ("o piccolo di g di n") l'insieme

$o(g(n)) = \{f(n):$ per qualunque costante positiva $c > 0$, esiste una costante $n_0 > 0$ tale che $0 \le f(n) < cg(n)$ per ogni $n \ge n_0\}$.

Per esempio, $2n = o(n^2)$, ma $2n^2 \neq o(n^2)$.

Le definizioni della notazione O e della notazione o sono simili; la principale differenza è che in $f(n) = O(g(n))$, il limite $0 \le f(n) \le cg(n)$ vale per *qualche* costante c, mentre in $f(n) = o(g(n))$, il limite $0 \le f(n) < cg(n)$ vale per *tutte* le costanti $c > 0$. Il concetto intuitivo nella notazione o è che la funzione $f(n)$ diventa trascurabile rispetto a $g(n)$ per n che tende all'infinito; cioè,

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0 . \qquad (2.1)$$

Alcuni autori usano questo limite come definizione della notazione o; la definizione di questo libro limita le funzioni anonime ad essere asintoticamente non negative.

Notazione ω

Per analogia, la notazione ω sta alla notazione Ω come la notazione o sta alla notazione O. Si usa la notazione ω per denotare un limite inferiore non asintoticamente stretto. Un modo per definirla è

$f(n) \in \omega(g(n))$ se e solo se $g(n) \in o(f(n))$.

Formalmente si definisce $\omega(g(n))$ ("omega piccolo di g di n") come l'insieme

$\omega(g(n)) = \{f(n)$: per qualunque costante positiva $c > 0$, esiste una costante $n_0 > 0$ tale che $0 \le cg(n) < f(n)$ per ogni $n \ge n_0\}$.

Per esempio, $n^2/2 = \omega(n)$, ma $n^2/2 \neq \omega(n^2)$. La relazione $f(n) = \omega(g(n))$ implica che

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

se il limite esiste; cioè $f(n)$ diventa arbitrariamente grande rispetto a $g(n)$ per n che tende all'infinito.

Confronto tra funzioni

Molte delle proprietà relazionali dei numeri reali sono valide anche per i confronti tra notazioni asintotiche. Nel seguito si assumerà che $f(n)$ e $g(n)$ siano asintoticamente positive.

Transitività:

$f(n) = \Theta(g(n))$ e $g(n) = \Theta(h(n))$ implicano $f(n) = \Theta(h(n))$,
$f(n) = O(g(n))$ e $g(n) = O(h(n))$ implicano $f(n) = O(h(n))$,
$f(n) = \Omega(g(n))$ e $g(n) = \Omega(h(n))$ implicano $f(n) = \Omega(h(n))$,
$f(n) = o(g(n))$ e $g(n) = o(h(n))$ implicano $f(n) = o(h(n))$,
$f(n) = \omega(g(n))$ e $g(n) = \omega(h(n))$ implicano $f(n) = \omega(h(n))$.

Riflessività:

$f(n) = \Theta(f(n))$,
$f(n) = O(f(n))$,
$f(n) = \Omega(f(n))$.

Simmetria:

$f(n) = \Theta(g(n))$ se e solo se $g(n) = \Theta(f(n))$.

Simmetria trasposta:

$f(n) = O(g(n))$ se e solo se $g(n) = \Omega(f(n))$,
$f(n) = o(g(n))$ se e solo se $g(n) = \omega(f(n))$.

Poiché queste proprietà valgono per le notazioni asintotiche, si può tracciare la seguente analogia tra il confronto asintotico di due funzioni f e g ed il confronto di due numeri reali a e b:

$$
\begin{aligned}
f(n) &= O(g(n)) &\approx\ & a \leq b, \\
f(n) &= \Omega(g(n)) &\approx\ & a \geq b, \\
f(n) &= \Theta(g(n)) &\approx\ & a = b, \\
f(n) &= o(g(n)) &\approx\ & a < b, \\
f(n) &= \omega(g(n)) &\approx\ & a > b.
\end{aligned}
$$

Una proprietà dei numeri reali che però non può essere trasferita alla notazione asintotica è la seguente:

Tricotomia: Per ogni coppia di numeri reali a e b, deve valere esattamente una delle seguenti espressioni: $a < b$, $a = b$, $a > b$.

Sebbene qualunque coppia di numeri possa essere confrontata, non tutte le funzioni sono asintoticamente confrontabili. Cioè, per una coppia di *funzioni* $f(n)$ e $g(n)$, può essere che non valga né $f(n) = O(g(n))$ né $f(n) = \Omega(g(n))$. Per esempio le funzioni n e $n^{1+\sin n}$, usando le notazioni asintotiche, non possono essere confrontate in quanto il valore dell'esponente di $n^{1+\sin n}$ oscilla tra 0 e 2, assumendo tutti i valori dell'intervallo.

Esercizi

2.1-1 Siano $f(n)$ e $g(n)$ funzioni asintoticamente non negative. Usando la definizione di base della notazione Θ, dimostrare che $\max(f(n), g(n)) = \Theta(f(n) + g(n))$.

2.1-2 Mostrare che per ogni coppia di costanti reali a e b, dove $b > 0$,

$$(n + a)^b = \Theta(n^b) \ . \tag{2.2}$$

2.1-3 Spiegare perché l'affermazione: "Il tempo di esecuzione dell'algoritmo A è almeno $O(n^2)$" è priva di significato.

2.1-4 È vero che $2^{n+1} = O(2^n)$? È vero che $2^{2n} = O(2^n)$?

2.1-5 Dimostrare il Teorema 2.1.

2.1-6 Dimostrare che il tempo di esecuzione di un algoritmo è $\Theta(g(n))$ se e solo se il suo tempo di esecuzione nel caso peggiore è $O(g(n))$ e il suo tempo di esecuzione nel caso migliore è $\Omega(g(n))$.

2.1-7 Dimostrare che $o(g(n)) \cap \omega(g(n))$ è l'insieme vuoto.

2.1-8 Si può estendere la notazione al caso di due parametri n ed m che possono tendere indipendentemente all'infinito con tassi di crescita diversi. Per una data funzione $g(n, m)$, si denota con $O(g(n, m))$ l'insieme di funzioni:

$$O(g(n, m)) = \{f(n, m): \text{esistono tre costanti positive } c, n_0 \text{ e } m_0 \text{ tali che } 0 \leq f(n, m) \leq cg(n, m) \text{ per ogni } n \geq n_0 \text{ e } m \geq m_0\}.$$

Dare le corrispondenti definizioni per $\Omega(g(n, m))$ e $\Theta(g(n, m))$.