es. 15 ft

Lezione 14

Confronti di crescita

Problema: date due successioni $\{a_n\}$, $\{b_n\}$ t.c.

$$\lim_{n \to +\infty} a_n = \lim_{n \to +\infty} b_n = +\infty$$

è possibile capire quale delle due tende a infinito "più velocemente"?

Consideriamo ad esempio la seguente tabella:

n	$\log_2 n$	n^2	$3n^2+n$	n^3	2^n
2	1	4	14	8	4
10	$3{,}322$	10^2	$3{,}1 \cdot 10^2$	10^3	$> 10^3$
10^2	$6{,}644$	10^4	$3{,}01 \cdot 10^4$	10^6	$\gg 10^{25}$
10^3	$9{,}966$	10^6	$3{,}001 \cdot 10^6$	10^9	$\gg 10^{250}$
10^4	$13{,}288$	10^8	$3{,}0001 \cdot 10^8$	10^{12}	$\ggg 10^{2500}$

Per confrontare la crescita di $\{a_n\}$ e $\{b_n\}$ si deve considerare il rapporto $\frac{a_n}{b_n}$ (che è ben definito almeno se n è sufficientemente grande, poiché $b_n \neq 0$ definitivamente). Si ha che:

- se $\lim_{n \to +\infty} \frac{a_n}{b_n} = 0$, $\{a_n\}$ tende a $+\infty$ "più lentamente" di $\{b_n\}$
- se $\lim_{n \to +\infty} \frac{a_n}{b_n} = +\infty$, $\{a_n\}$ tende a $+\infty$ "più velocemente" di $\{b_n\}$
- se $\lim_{n \to +\infty} \frac{a_n}{b_n} = \ell \in (0, +\infty)$, $\{a_n\}$ e $\{b_n\}$ tendono a $+\infty$ "con uguale velocità"

Scala infiniti

"con uguale velocità"

Si può dimostrare che

$$\boxed{\begin{array}{lll} 1) & \lim_{n\to+\infty} \dfrac{\log_a n}{n^k} = 0 & \forall a>1,\ \forall k>0 \\ 2) & \lim_{n\to+\infty} \dfrac{n^k}{n^{\bar{J}}} = 0 & \forall \bar{J}>k>0 \\ 3) & \lim_{n\to+\infty} \dfrac{n^{\bar{J}}}{q^n} = 0 & \forall \bar{J}>0,\ \forall q>1 \end{array}}$$

"Gerarchia degli infiniti"

Per "sintetizzare" informazioni sulla crescita è utile introdurre i cosiddetti simboli di Landau

Def Date due successioni $\{a_n\}, \{b_n\}$ t.c. $b_n \neq 0$ definitiv. si dice che:

- $\{a_n\}$ è "o-piccolo" di $\{b_n\}$, e si scrive $\boxed{a_n = o(b_n)}$, se

$$\lim_{n\to+\infty} \frac{a_n}{b_n} = 0$$

- $\{a_n\}$ è "o-grande" di $\{b_n\}$, e si scrive $\boxed{a_n = O(b_n)}$, se esiste $C>0$ t.c.

$$\left|\frac{a_n}{b_n}\right| \leq C \qquad \text{definitivamente per } n\to+\infty$$

- $\{a_n\}$ è "theta-grande" di $\{b_n\}$, e si scrive $\boxed{a_n = \Theta(b_n)}$ se esistono $c, C > 0$ t.c.

$$c \leq \left|\frac{a_n}{b_n}\right| \leq C \qquad \text{definitivamente per } n\to+\infty$$

- $\{a_n\}$ è equivalente a $\{b_n\}$, e si scrive $\boxed{a_n \sim b_n}$ se

- $\{a_n\}$ è equivalente a $\{b_n\}$, e si scrive $\boxed{a_n \sim b_n}$ se

$$\lim_{n\to+\infty} \frac{a_n}{b_n} = 1$$

Oss

Osservazioni

1) $a_n = o(b_n)$ vuol quindi dire che $\{a_n\}$ cresce più lentamente di $\{b_n\}$; la gerarchia degli infiniti si può dunque scrivere come

$$\log_a n = o(n^k) \qquad a>1,\ k>0$$

$$n^k = o(n^\delta) \qquad 0<k<\delta$$

$$n^\delta = o(q^n) \qquad \delta>0,\ q>1$$

Si noti inoltre che

$$\lim_{n\to+\infty} \frac{a_n}{b_n} = +\infty \iff b_n = o(a_n)$$

2) $a_n = O(b_n)$ vuol invece dire che $\{a_n\}$ cresce "mai più velocemente" di $\{b_n\}$; quindi

$$\underbrace{a_n = o(b_n)}_{\forall \varepsilon>0\ \left|\frac{a_n}{b_n}\right| < \varepsilon \text{ definitiv.}} \Rightarrow a_n = O(b_n)$$

ma non vale il viceversa. Ad esempio

$$n^2 = O(n^2+1) \qquad \text{poiché } 0 \le \frac{n^2}{n^2+1} \le 1$$

$$n^2 \text{ non è } o(n^2+1) \qquad \text{poiché } \lim_{n\to+\infty} \frac{n^2}{n^2+1} = 1 \neq 0$$

Osserviamo inoltre che, se $\exists \lim_{n\to+\infty} \left|\frac{a_n}{b_n}\right|$, allora

Osserviamo inoltre che, se $\exists \lim_{n\to+\infty} \left|\frac{a_n}{b_n}\right|$, allora

$$a_n = O(b_n) \iff \lim_{n\to+\infty} \left|\frac{a_n}{b_n}\right| \in [0,+\infty)$$

3) $a_n = \Theta(b_n)$ vuol invece dire che $\{a_n\}$ cresce "non più velocemente" e "non più lentamente" di $\{b_n\}$ (cioè $\{a_n\}$ e $\{b_n\}$ hanno lo stesso tipo di crescita)

Quindi:

$$a_n = \Theta(b_n) \Rightarrow \begin{cases} a_n = O(b_n) \\ a_n \text{ non è } o(b_n) \end{cases}$$

Ad esempio

$$n^2 = O(n^3) \quad \text{ma} \quad n^2 \text{ non è } \Theta(n^3)$$

(infatti: $n^2 = o(n^3)$)

In particolare, se $\exists \lim_{n\to+\infty} \left|\frac{a_n}{b_n}\right|$, allora

$$a_n = \Theta(b_n) \iff \lim_{n\to+\infty} \left|\frac{a_n}{b_n}\right| \in (0,+\infty)$$

4) $a_n \sim b_n$ vuol invece dire che non solo $\{a_n\}$ e $\{b_n\}$ tendono a $+\infty$ con uguale velocità, ma sono "sempre più simili" (equivalenti) per $n \to +\infty$

Ad esempio

$$n^2 \sim n^2+1 \qquad \text{poiché} \quad \lim_{n\to+\infty} \frac{n^2}{n^2+1} = 1$$

$$n^2 \not\sim 2n^2+1 \qquad \text{poiché} \quad \lim_{n\to+\infty} \frac{n^2}{2n^2+1} = \frac{1}{2} \neq 1$$

$\left(\text{quindi } n^2 = O(2n^2+1) \right)$

In generale, se

$$P(n) = a_0 n^k + a_1 n^{k-1} + \ldots + a_{k-1} n + a_n$$

con $k \geq 1$, $a_0, \ldots, a_k \in \mathbb{R}$ e $a_0 \neq 0$, si ha

$$P(n) \sim a_0 n^k$$

infatti:

$$\frac{P(n)}{a_0 n^k} = \frac{a_0 n^k \left(1 + \frac{a_1}{a_0} \overbrace{\frac{1}{n}}^{\to 0} + \ldots + \frac{a_{k-1}}{a_0} \overbrace{\frac{1}{n^{k-1}}}^{\to 0} + \frac{a_n}{a_0} \overbrace{\frac{1}{n^k}}^{\to 0}\right)}{a_0 n^k} \to 1$$

Similmente, se $P(n)$ è come sopra e

$$Q(n) = b_0 n^j + b_1 n^{j-1} + \ldots + b_j$$

e $b_0 \neq 0$ allora

$$\frac{P(n)}{Q(n)} \sim \frac{a_0}{b_0} \frac{n^k}{n^j} = \frac{a_0}{b_0} n^{k-j}$$

Osservazione : i simboli di Landau possono essere introdotti anche per funzioni $f, g : [a, +\infty) \to \mathbb{R}$ t.c. $g(x) \neq 0$ definitiv per $x \to +\infty$

Esempi:

1) $x^2 = O(x^3)$ per $x \to +\infty$

2) $x^2 = o(x^3)$ per $x \to +\infty$

2) $x^2 = o(x^3)$ per $x \to +\infty$

3) $\dfrac{x^2+3x}{5x^3+1} \sim \dfrac{1}{5x}$ per $x \to +\infty$

es.

4) $\dfrac{x^3+\boxed{\sin x}}{x^2+1} \sim x$ per $x \to +\infty$

è limitato

4') $\dfrac{x^3+\boxed{\log x}}{x^2+1} \sim x$

cresce meno di x^3

Un'applicazione: complessità di un algoritmo

Complessità temporale = tempo impiegato dall'algoritmo

Possiamo supporre che sia proporzionale al numero di "operazioni" eseguite, che chiamiamo

$Q_{\boxed{n}}$ ove n = numero di componenti del vettore di input

Si vuole stimare la crescita di Q_n per $n \to +\infty$

Esempio: algoritmo di ricerca di un numero dato (es $n = 42$) in una stringa ordinata di lunghezza n

1) Ricerca esaustiva: scorro il vettore in modo sequenziale confrontando ogni elemento con 42

Caso peggiore: 42 non c'è ed è più grande di tutti

Caso peggiore: 42 non c'è ed è più grande di tutti i valori di input

$n = 3$

2	4	6	7	15	21	28	31	35

3 confronti $\to$ 42 non c'è

Quindi, nel caso peggiore, $Q_n = n$

La complessità dell'algoritmo è dunque $O(n)$

2) Ricerca binaria:

2	4	6	7	15	21	28	31	35

Confronto 42 con elemento centrale (15)

~~2~~	~~4~~	~~6~~	~~7~~	~~15~~	21	28	31	35

" " " " " (31)

~~2~~	~~4~~	~~6~~	~~7~~	~~15~~	~~21~~	~~28~~	~~31~~	35

" " " " " (35)

3 confronti $\to$ 42 non c'è

Cosa ho fatto?

$$n \to n/2 \to n/2^2 \to \dots \to n/2^m$$

Nel caso peggiore l'algoritmo si conclude quando

$$n/2^m \le 1$$

cioè $\quad m \ge \log_2 n$

La complessità dell'algoritmo è dunque $O(\log_2 n)$

Conclusione: se n è grande, l'algoritmo di ricerca binaria è più efficiente