Camil Demetrescu, Irene Finocchi, Giuseppe F. Italiano

Algoritmi e Strutture Dati

Capitolo 2
Modelli di calcolo e metodologie di analisi

Riepilogo algoritmi `Fibonacci`

	Numero di linee di codice	Occupazione di memoria
`fibonacci1`	$\Theta(1)$	$\Theta(1)$
`fibonacci2`	$\Theta(\Phi^n)$	$\Theta(\Phi^n)$*
`fibonacci3`	$\Theta(n)$	$\Theta(n)$
`fibonacci4`	$\Theta(n)$	$\Theta(1)$
`fibonacci5`	$\Theta(n)$	$\Theta(1)$
`fibonacci6`	$\Theta(\log n)$	$\Theta(\log n)$*

* per le variabili di lavoro delle chiamate ricorsive

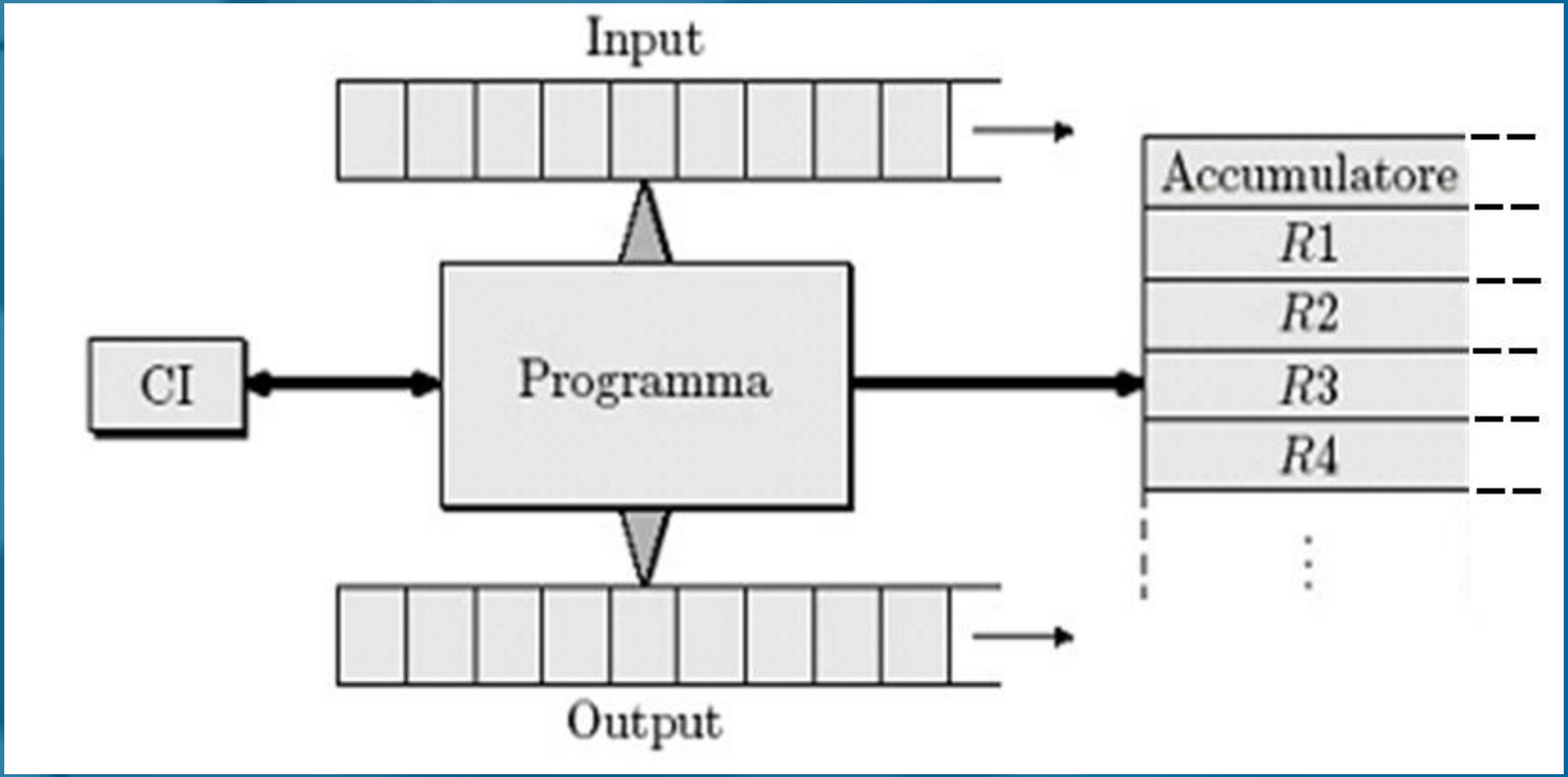
Modello di calcolo

- Contare il numero di linee dello pseudocodice di un algoritmo non è sufficiente a comprenderne la complessità temporale: quali operazioni reali si celano dietro lo pseudocodice???
- Per valutare la complessità di un algoritmo, bisogna quindi prima di tutto stabilire un modello di calcolo di riferimento su cui esso viene eseguito
- Un modello di calcolo è una macchina astratta che definisce l'insieme delle operazioni ammissibili ed eseguibili durante una computazione, e ne specifica i relativi costi (in termini di tempo e spazio utilizzato)
- Alcuni modelli di calcolo famosi: Macchina di Turing, Automi a Stati Finiti, Funzioni Ricorsive, Macchina a Registri, etc… (sono tutti modelli di calcolo equivalenti, cioè in grado di calcolare le stesse funzioni)

Il nostro modello di calcolo: la RAM

- La RAM (*Random Access Machine*) è un tipo particolare di macchina a registri (locazioni di memoria ad accesso diretto)
- La RAM è definita da:
  - un nastro di ingresso e uno di uscita, ove saranno scritti l'input e l'output, rispettivamente
  - una memoria ad accesso diretto strutturata come un array (di dimensione infinita) in cui ogni cella può contenere un valore intero arbitrariamente grande
  - un programma finito di istruzioni elementari
  - un registro detto accumulatore (contiene gli operandi dell'istruzione corrente)
  - un registro detto contatore delle istruzioni (contiene l'indirizzo dell'istruzione successiva)
- La RAM è un'astrazione dell'architettura di von Neumann

La RAM



Criterio di costo uniforme

- Nel modello RAM a costi uniformi, le seguenti istruzioni elementari hanno un costo unitario
  - istruzione ingresso/uscita (lettura /scrittura del nastro di input/output)
  - operazione aritmetico (+,-,×,:,√, elevamento a potenza, etc.) /logica (test)
  - accesso/modifica del contenuto della memoria
- **Complessità temporale** tempo(I) misurata come numero di istruzioni elementari eseguite sull'istanza di input I
- **Complessità spaziale** spazio(I) misurata come numero massimo di celle di memoria della RAM occupate durante l'esecuzione sull'istanza di input I

Dimensione dell'input

- Misureremo le risorse di calcolo usate da un algoritmo (tempo di esecuzione / occupazione di memoria) in funzione della dimensione dell'istanza I in input
- Esistono due modalità di caratterizzazione della dimensione dell'input:
  - Quantità di memoria effettiva utilizzata per codificare l'input (ad esempio, numero di bit necessari per rappresentare un valore in input, come nel problema di Fibonacci)
  - Parametrizzazione della dimensione dell'input (ad esempio, numero di elementi di una sequenza da ordinare)
- In tutti i problemi che incontreremo in futuro, la dimensione dell'input sarà parametrizzata attraverso il numero n di elementi dell'istanza

Domanda di approfondimento

- Qual è la complessità temporale degli algoritmi `Fibonacci6`, `Fibonacci4` e `Fibonacci2` in funzione della rappresentazione dell'input?

Notazione asintotica

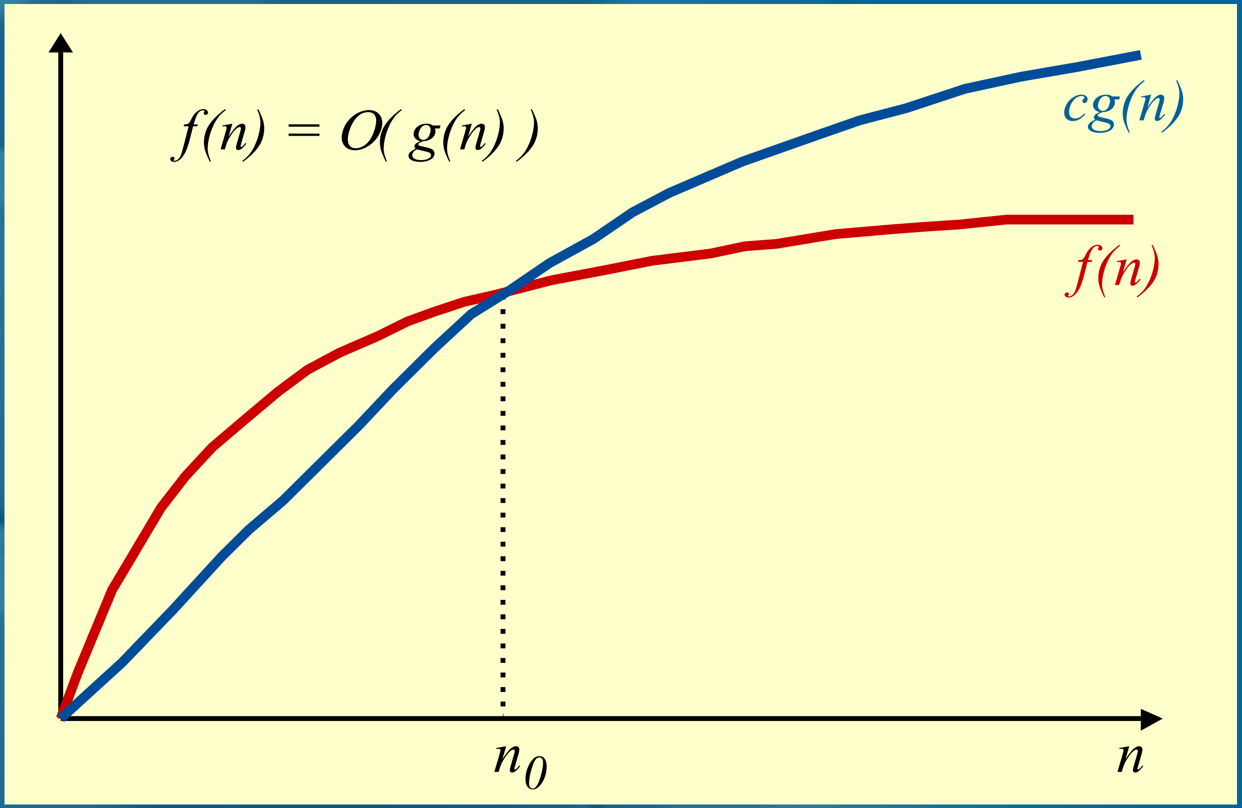
Notazione asintotica e operazioni dominanti

- Complessità temporale e spaziale saranno espresse in notazione asintotica rispetto alla dimensione dell'input
- La notazione asintotica è un'astrazione utile per descrivere l'ordine di grandezza di una funzione ignorando i dettagli non influenti, come costanti moltiplicative e termini di ordine inferiore
- Ai fini dell'analisi asintotica, sarà sufficiente considerare le cosiddette operazioni dominanti, ovvero quelle che nel caso peggiore vengono eseguite più spesso
- Queste si trovano annidate nei cicli più interni dello pseudocodice che descrive l'algoritmo
- NOTA: Nel prosieguo, ci concentreremo su funzioni di variabile intera non negativa a valori reali non negativi

$$f : n \in \aleph \rightarrow \Re \geq 0$$

Notazione asintotica O (‘o’ grande)

$f(n) = O(g(n))$ se $\exists$ due costanti $c>0$ e $n_0 \geq 0$ tali che $f(n) \leq c\, g(n)$ per ogni $n \geq n_0$



Legame con il concetto di limite

Se g(n) è definitivamente diversa da 0 per $n \to \infty$ (praticamente, tutti i casi di nostro interesse), avremo che

$$f(n) = O(g(n)) \quad \Leftrightarrow \quad \limsup_{n\to\infty} \frac{f(n)}{g(n)} < \infty$$

ovvero f(n)=O(g(n)) se e solo se f(n) è un infinito di ordine non superiore a g(n)

Un caso notevole: i polinomi

Sia $f(n) = a_d\, n^d + a_{d-1}\, n^{d-1} + \ldots + a_0$ un polinomio di grado d (con $a_d>0$); dimostriamo che $f(n)=O(n^d)$

Se scegliamo $c = a_d + |a_{d-1}| + \ldots + |a_0| \Rightarrow$

$c\, n^d = a_d\, n^d + |a_{d-1}|\, n^d + \ldots + |a_0|\, n^d \geq a_d\, n^d + |a_{d-1}|\, n^{d-1} + \ldots + |a_0| \geq a_d\, n^d + a_{d-1}\, n^{d-1} + \ldots + a_0 = f(n)$

$\Rightarrow f(n) \leq c\, n^d \;\; \forall\, n \geq 0$, e quindi posso scegliere $n_0=0$

Alternativamente:

$$\limsup_{n\to\infty} \frac{f(n)}{g(n)} = \frac{a_d n^d + \ldots + a_0}{n^d} = a_d < \infty \Rightarrow f(n) = O(g(n))$$

Camil Demetrescu, Irene Finocchi, Giuseppe F. Italiano

Esempi

- Sia $f(n) = 2n^2 + 3n$; applichiamo la dimostrazione appena fatta per mostrare che $f(n)=O(n^2)$:

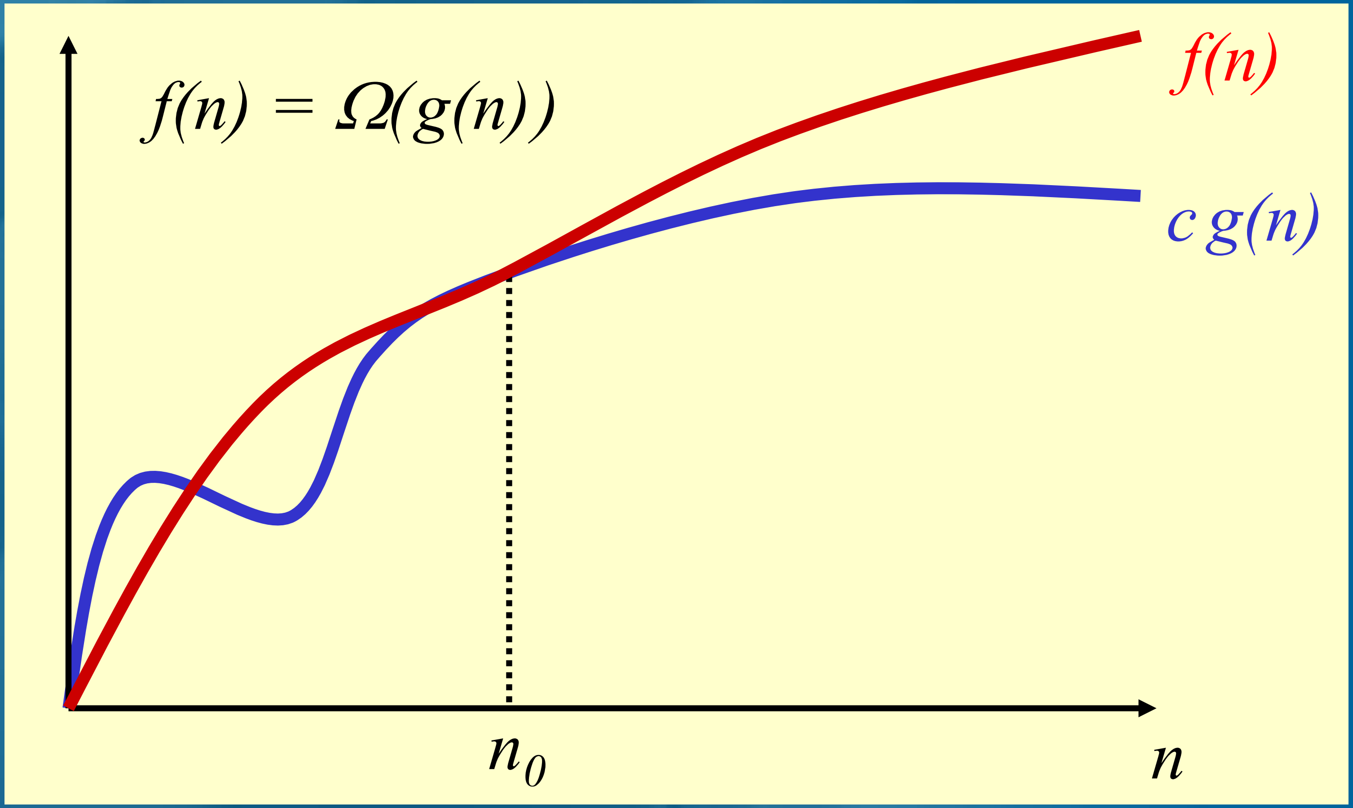
  scegliendo $c = a_2 + |a_1| + |a_0| = 2+3+0=5$

  avremo che $2n^2 + 3n \leq 5n^2$ per ogni $n \geq n_0=0$.
- $f(n) = O(n^3)$ ($c=1$, $n_0=3$, oppure $c=2$, $n_0=2$)
- $f(n) = O(2^n)$ ($c=1$, $n_0=7$, oppure $c=2$, $n_0=6$)
- Invece, $f(n) \neq O(n)$

Camil Demetrescu, Irene Finocchi, Giuseppe F. Italiano

Notazione asintotica Ω (omega grande)

$f(n) = \Omega(g(n))$ se $\exists$ due costanti $c>0$ e $n_0 \geq 0$ tali che $f(n) \geq c\, g(n)$ per ogni $n \geq n_0$



Legame con il concetto di limite

Se g(n) è definitivamente diversa da 0 per $n \rightarrow \infty$ (praticamente, tutti i casi di nostro interesse), avremo che

$$f(n) = \Omega(g(n)) \quad \Leftrightarrow \quad \limsup_{n\to\infty} \frac{f(n)}{g(n)} > 0$$

ovvero $f(n)=\Omega(g(n))$ se e solo se f(n) è un infinito di ordine non inferiore a g(n)

Un caso notevole: i polinomi

Sia $f(n) = a_d\, n^d + a_{d-1}\, n^{d-1} + \ldots + a_0$ un polinomio di grado d (con $a_d>0$); dimostriamo che $f(n)=\Omega(n^d)$

Infatti: $f(n)/n^d = a_d + a_{d-1}\, n^{-1} + \ldots + a_0\, n^{-d} \quad \Rightarrow$

$\exists\ n_0: \forall\ n \geq n_0 \qquad a_d - |a_{d-1}| n^{-1} - \ldots - |a_0|\, n^{-d} > 0$

Se scegliamo $c = a_d - |a_{d-1}|\, n_0^{-1} - \ldots - |a_0|\, n_0^{-d} \Rightarrow$

$c\, n^d = a_d\, n^d - |a_{d-1}|\, n^d\, n_0^{-1} - \ldots - |a_0|\, n^d\, n_0^{-d}$

e poiché per $n \geq n_0$ si ha $n^d\, n_0^{-k} \geq n^d\, n^{-k} \geq n^{d-k}$

$c\, n^d \leq a_d\, n^d - |a_{d-1}|\, n^{d-1} - \ldots - |a_0| \leq a_d\, n^d + a_{d-1}\, n^{d-1} + \ldots + a_0 = f(n)$

$\Rightarrow \quad \forall\ n \geq n_0 \quad c\, n^d \leq f(n)$

Alternativamente:

$$\limsup_{n\to\infty} \frac{f(n)}{g(n)} = \frac{a_d n^d + \ldots + a_0}{n^d} = a_d > 0 \Rightarrow f(n) = \Omega(g(n))$$

Esempi

- Sia $f(n) = 2n^2 - 5n$; applichiamo la dimostrazione appena fatta per mostrare che $f(n)= \Omega(n^2)$:

  $f(n)/n^2 = (2n^2 - 5n)/n^2 = 2 - 5/n$

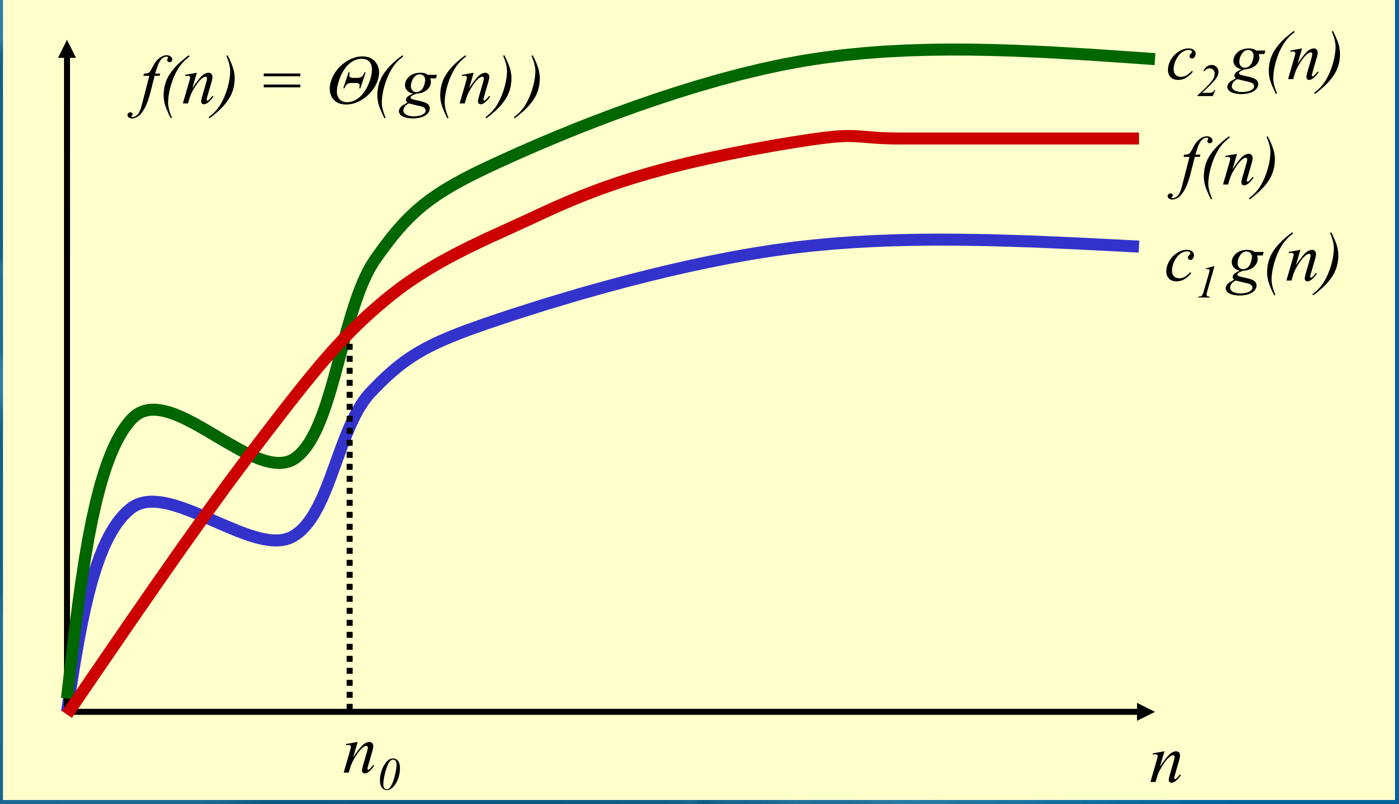
  ma $2 - 5/n > 0$ per $n \geq 3$ (quindi $n_0=3$);

  Scegliendo $c = a_2 - |a_1|/n_0 - |a_0|/n_0^2 = 2-5/3=1/3$

  avremo che $2n^2 - 5n \geq (1/3)\, n^2$ per $n \geq n_0=3$.
- $f(n) = \Omega(n)$ $(c=1, n_0=2)$
- $f(n) = \Omega(\log n)$ $(c=1, n_0=2)$
- Invece, $f(n) \neq \Omega(n^3)$

Notazione asintotica Θ (theta grande)

$f(n) = \Theta(g(n))$ se $\exists$ tre costanti $c_1, c_2 > 0$ e $n_0 \geq 0$ tali che $c_1\, g(n) \leq f(n) \leq c_2\, g(n)$ per ogni $n \geq n_0$



Legame con il concetto di limite

Se g(n) è definitivamente diversa da 0 per $n\rightarrow\infty$ (praticamente, tutti i casi di nostro interesse), avremo che

$$f(n)=\Theta(g(n)) \Leftrightarrow \lim_{n\rightarrow\infty}\frac{f(n)}{g(n)}=L \quad \text{con} \quad 0<L<\infty$$

ovvero $f(n)=\Theta(g(n))$ se e solo se f(n) è un infinito dello stesso ordine di g(n)

Relazioni tra O, Ω e Θ

$$f(n) = \Theta(g(n)) \Rightarrow f(n) = O(g(n))$$

$$f(n) = O(g(n)) \not\Rightarrow f(n) = \Theta(g(n))$$

$$f(n) = \Theta(g(n)) \Rightarrow f(n) = \Omega(g(n))$$

$$f(n) = \Omega(g(n)) \not\Rightarrow f(n) = \Theta(g(n))$$

$$f(n) = \Theta(g(n)) \Leftrightarrow f(n) = \Omega(g(n)) \text{ e } f(n) = O(g(n))$$

Notazione asintotica o ('o' piccolo)

- $f(n) = o(g(n))$ se $\forall\ c>0$, $\exists\ n_0 \geq 0$ tale che

  $f(n) \leq c\ g(n)$ per ogni $n \geq n_0$

- Se $g(n)$ è definitivamente diversa da 0 per $n \to \infty$ (praticamente, tutti i casi di nostro interesse), avremo che:

$$f(n) = o(g(n)) \quad \Leftrightarrow \quad \lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$$

- Notare che $o(g(n)) \subset O(g(n))$

Notazione asintotica ω (omega piccolo)

- $f(n) = \omega(g(n))$ se $\forall\ c>0$, $\exists\ n_0 \geq 0$ tale che

  $f(n) \geq c\ g(n)$ per ogni $n \geq n_0$
- Se $g(n)$ è definitivamente diversa da 0 per $n \to \infty$ (praticamente, tutti i casi di nostro interesse), avremo che:

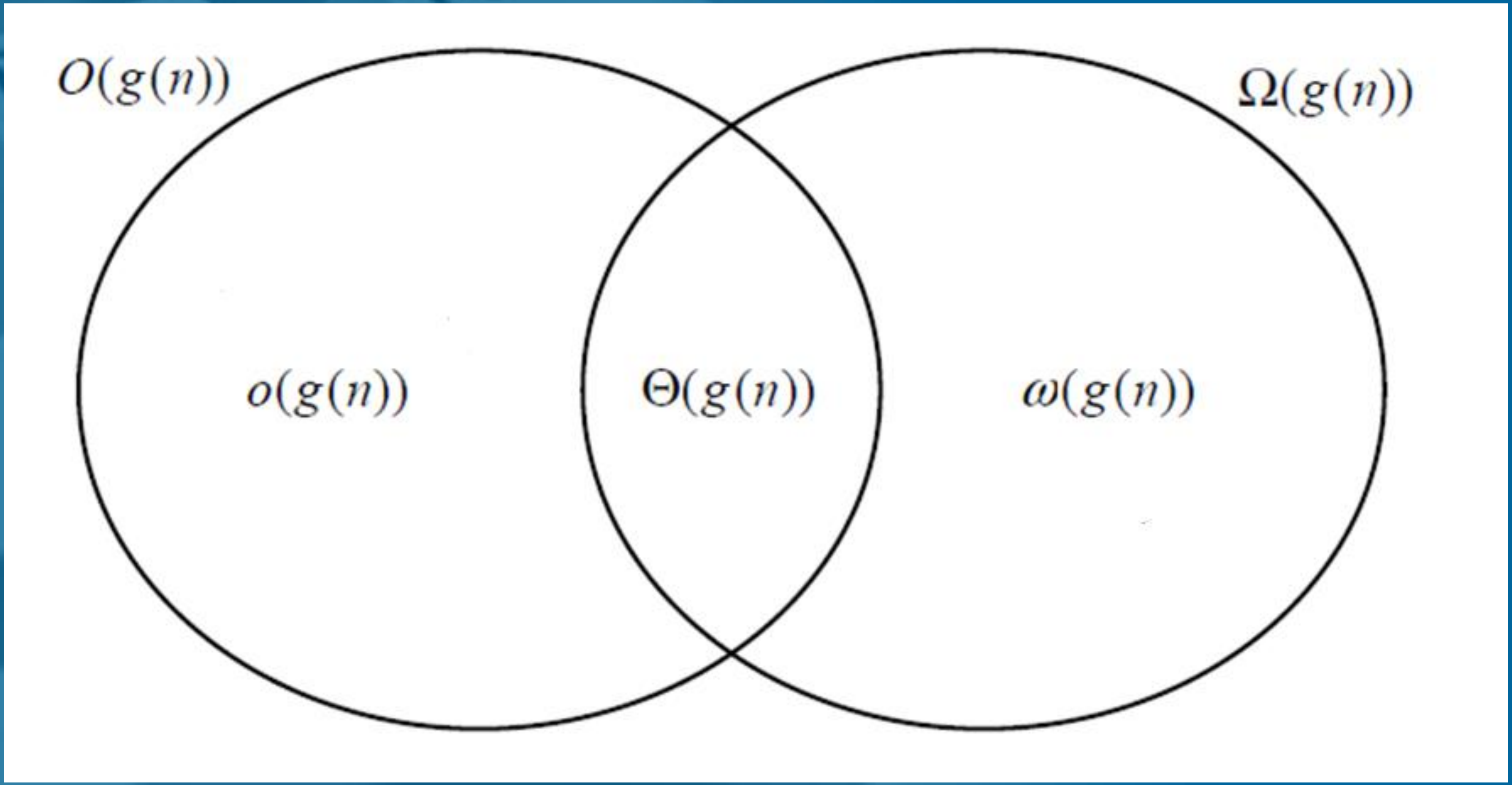
$$f(n) = \omega(g(n)) \quad \Leftrightarrow \quad \lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$$

- Notare che $\omega(g(n)) \quad \subset \quad \Omega(g(n))$

Analogie

O	Ω	Θ	o	ω
$\leq$	$\geq$	$=$	$<$	$>$

Graficamente



Proprietà della notazione asintotica

Transitività

$$f(n)=\Theta(g(n)) \quad e \quad g(n)=\Theta(h(n)) \quad \Rightarrow \quad f(n)=\Theta(h(n))$$

$$f(n)=O(g(n)) \quad e \quad g(n)=O(h(n)) \quad \Rightarrow \quad f(n)=O(h(n))$$

$$f(n)=\Omega(g(n)) \quad e \quad g(n)=\Omega(h(n)) \quad \Rightarrow \quad f(n)=\Omega(h(n))$$

$$f(n)=o(g(n)) \quad e \quad g(n)=o(h(n)) \quad \Rightarrow \quad f(n)=o(h(n))$$

$$f(n)=\omega(g(n)) \quad e \quad g(n)=\omega(h(n)) \quad \Rightarrow \quad f(n)=\omega(h(n))$$

Riflessività

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

$$f(n)=O(f(n))$$

$$f(n)=\Omega(f(n))$$

Simmetria

$$f(n)=\Theta(g(n)) \quad \Leftrightarrow \quad g(n)=\Theta(f(n))$$

Simmetria trasposta

$$f(n)=O(g(n)) \quad \Leftrightarrow \quad g(n)=\Omega(f(n))$$

$$f(n)=o(g(n)) \quad \Leftrightarrow \quad g(n)=\omega(f(n))$$

Relazioni asintotiche notevoli

Polinomi

$P(n) = a_d n^d + a_{d-1} n^{d-1} + \dots + a_0$
$a_d > 0$

$\Rightarrow$

$P(n) = O(n^d)$, $P(n) = \Omega(n^d) \Rightarrow P(n) = \Theta(n^d)$
$P(n) = O(n^k)\ \forall\ k \geq d$, $P(n) \neq O(n^k)\ \forall\ k<d$
$P(n) = \Omega(n^k)\ \forall\ k \leq d$, $P(n) \neq \Omega(n^k)\ \forall\ k>d$
$P(n) = o(n^k)\ \forall\ k>d$, $P(n) = \omega(n^k)\ \forall\ k<d$

Esponenziali

$f(n) = a^n$
$a > 1$

$$\lim_{n\to\infty} \frac{a^n}{n^d} = \infty$$

$\Rightarrow$

$a^n = \omega(n^d)\ \forall\ d>0$
$\Rightarrow a^n = \Omega(n^d)\ \forall\ d>0$

Logaritmi

$f(n) = \log_b n \quad b>1$

$$\lim_{n\to\infty} \frac{(\log_b n)^c}{n^d} = 0, \forall c,d \geq 1$$

$\Rightarrow$

$(\log_b n)^c = o(n^d)\ \forall\ c,d \geq 1$
$\Rightarrow (\log_b n)^c = O(n^d)\ \forall\ c,d \geq 1$

Fattoriali

$f(n) = n! = n\cdot(n-1)\cdot(n-2)\cdot\ldots\ldots\cdot 2\cdot 1$

$\Rightarrow$

$n! = o(n^n) \Rightarrow n! = O(n^n)$
$n! = \omega(a^n) \Rightarrow n! = \Omega(a^n)\ \forall\ a>0$