

## 2.1 Ordini di grandezza delle funzioni

In informatica si usa la seguente notazione asintotica per indicare il possibile ordine di grandezza di una funzione. La variabile indipendente  $n$  è in genere un intero positivo e indica la "dimensione" dei dati di un problema (per es. il numero di bit necessario a descrivere i dati). La funzione di cui si studia l'ordine di grandezza è in genere proporzionale al tempo di calcolo (complessità in tempo), oppure alla memoria impiegata oltre a quella necessaria ai dati d'ingresso (complessità in spazio).

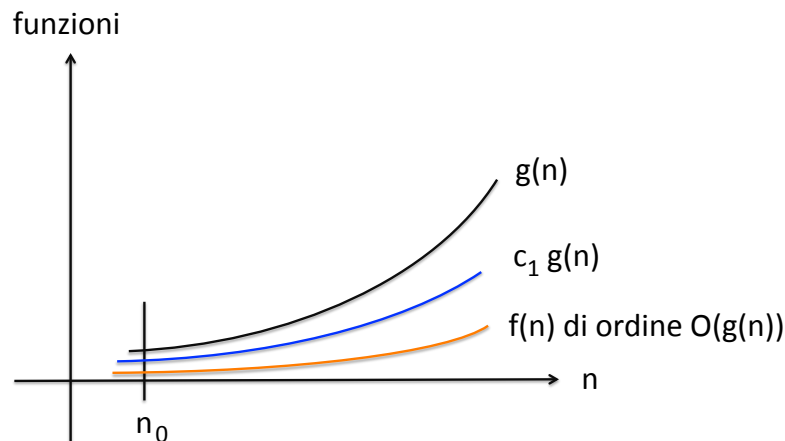
Con riferimento alle figure riportate sotto:

### Notazione $O$

$f(n)$  è di ordine  $O(g(n))$  se esistono due costanti positive  $c_1, n_0$ , tali che  $0 \leq f(n) \leq c_1 g(n)$  per ogni  $n \geq n_0$ .

Cioè la funzione  $f(n)$  ha un andamento che non sale al di sopra di  $g(n)$  al crescere di  $n$ , a meno di una costante moltiplicativa. Anche qui contano solo i termini di ordine massimo. Per esempio  $f(n) = 2n^2 - 3n + 1$  è di ordine  $O(n^2)$ , ma anche  $O(n^3)$  ecc..

La notazione  $O$  si impiega per esempio per indicare il tempo di un algoritmo di cui **non** si conosce compiutamente il comportamento, ma che si sa che non può superare  $g(n)$ ; oppure che **non** si comporta allo stesso modo per tutti gli insiemi di dati di dimensione  $n$  che gli si presentano, ma per alcuni richiede tempo proporzionale a  $g(n)$ , per altri meno.

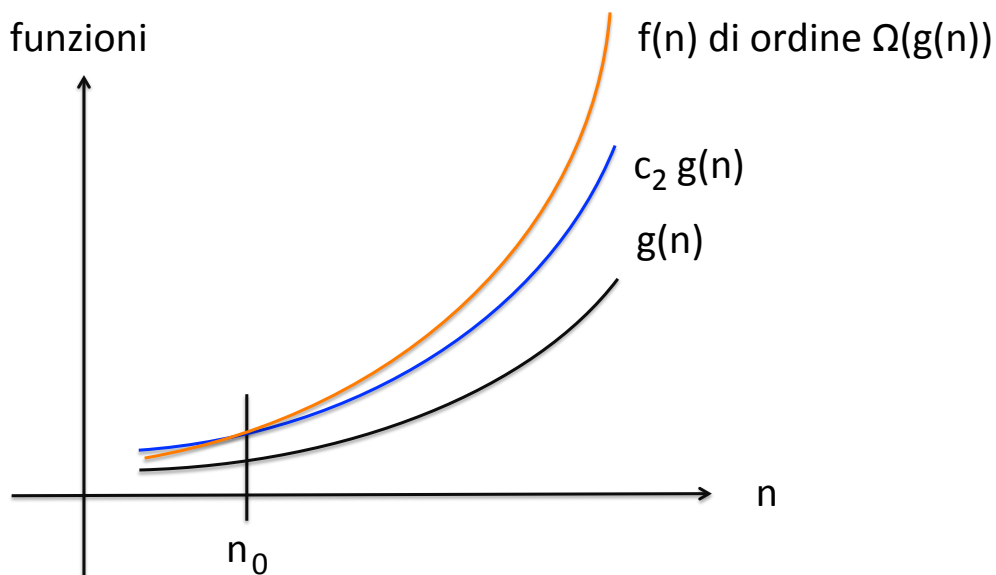


**Notazione  $\Omega$** 

$f(n)$  è di ordine  $\Omega(g(n))$  se esistono due costanti positive  $c_2, n_0$ , tali che  $0 \leq c_2 g(n) \leq f(n)$  per ogni  $n \geq n_0$ .

Cioè la funzione  $f(n)$  ha un andamento che non scende al di sotto di  $g(n)$  al crescere di  $n$ , a meno di una costante moltiplicativa. Anche qui contano solo i termini di grado massimo. Per esempio  $f(n) = 2n^2 - 3n + 1$  è di ordine  $\Omega(n^2)$ , ma anche  $\Omega(n \log n)$  ecc..

La notazione  $\Omega$  si impiega per esempio per indicare il limite inferiore al tempo di soluzione di un problema, che si applica quindi a *tutti* i suoi algoritmi di soluzione.



## Notazione $\Theta$

$f(n)$  è di ordine  $\Theta(g(n))$  se esistono tre costanti positive  $c_1$ ,  $c_2$ ,  $n_0$ , tali che  $0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$  per ogni  $n \geq n_0$ .

Cioè le funzioni  $f(n)$  e  $g(n)$  hanno lo stesso andamento al crescere di  $n$  a meno di costanti moltiplicative, e contano solo i termini di ordine massimo. Per esempio  $f(n) = n^2 - 3n + 1$  è di ordine  $\Theta(n^2)$ , ma anche  $f(n) = 3n^2 - 5$  è di ordine  $\Theta(n^2)$  perché non si considerano le costanti moltiplicative.

La notazione  $\Theta$  si impiega per esempio per indicare il tempo di un algoritmo di cui si conosce compiutamente il comportamento e che, a pari valore di  $n$ , si comporta allo stesso modo per tutti gli insiemi di dati di dimensione  $n$  che gli si presentano.

