

Wolfe in minimum "grand".

E' fondamentale produrre algoritmi efficienti.

● Abbiamo visto che e' possibile progettare algoritmi tramite "gusci" (base brute, divide-et-winna...)

Abbiamo visto che e' possibile unificare e valutare algoritmi in base al numero di passi che essi compiono per produrre output.

Risolvere l'algoritmo con complessita' $O(N)$ DA FARE

L'algoritmo divide-et-impire rappresenta un approccio molto efficace per la risoluzione di vari problemi computazionali. Questi algoritmi dividono ricorsivamente un problema in due o più sotto problemi più piccoli e che questi ultimi diventino di semplice risoluzione, successivamente si combinano le soluzioni al fine di ottenere la soluzione al problema dato. ~~Algoritmo~~ Problema sviluppato secondo questa tecnica e' sostanzialmente diviso in tre parti:

- Base base: parte all'inizio delle funzioni da sviluppare, e risolve in modo diretto il problema quando la taglia risulta inferiore alle soglie previste.
- Divide: in questa parte si procede alle suddivisioni dell'input in sotto problemi di taglia inferiore e alle chiamate ricorsive dello stesso problema con l'input appena creato.
- Impire: l'ultima fase del paradigma prevede di ricombinare l'output ottenuto dalle precedenti chiamate ricorsive al fine di ottenere il risultato finale.