



Tutte le complessità fondamentali APA

Algoritmi e programmazione (Politecnico di Torino)

Tutte le complessità fondamentali di APA a parer mio, ovvero quelle che hanno chiesto all'orale 😊.

Non mi assumo nessuna responsabilità. Quelle di calcolo combinatorio che non si trovano da nessuna parte le ho messe perché state dette da degli studenti a cui è stata fatta la domanda, quindi suppongo siano giuste dato che non hanno saputo rispondere e sono state dette successivamente dai prof agli studenti sotto orale.

Quick find: $O(\text{numero coppie} * \text{dim vettore})$

Quick union: $O(\text{numero coppie} * \text{lung catena})$

Quick union pesata: $O(\text{numero coppie} * \text{lung catena}) \Rightarrow$ lunghe che cresce in modo logaritmico

Fibonacci: $T(n) = 1 + T(n-1) + T(n-2) = O(2^n)$

Hanoi: $T(n) = 1 + 2T(n-1) = O(2^n)$

Fattoriale: $O(n)$

Moltiplicazione di 2 interi: $O(n^2)$

Karatsuba: $O(n^{\log 3})$, $a=3$ moltiplic ricorsive e non 4

Theta: limite asintotico stretto: $0 \leq c_1g(n) \leq T(n) \leq c_2g(n)$

LIS: $O(n^2)$

Post-order: $O(n)$

Liste: $O(n)$

Dicotomica: $O(\log n)$

Bubble sort: $O(n^2)$ metti il MAX

Selection sort: $O(n^2)$ trova MIN

Insertion sort: $O(n^2)$

Shellsort, sequenz Knuth: $O(n^{3/2})$

Merge sort: $O(n \log n)$

Quick sort: $O(n^2)$ oppure $O(n \log n)$

Counting sort: $O(n)$

ADT SET unione insiemistica: $O(N)$; intersezione $O(N^2)$

HEAPify: $O(\log n)$

HEAPbuild: $O(n)$, imprecisa: $O(n \log n)$

HEAPsort: $O(n \log n)$

PQ insert, showMax, extraMax: $1, N, N \text{ /-/ } N, 1, 1 \text{ /-/ } \log n, 1, \log n$

PQinsert: $O(\log n)$

PQextraMax: $O(\log n)$

PQchange: $O(n) = \text{search}O(n) + \text{Heapify}O(\log n)$

Madj: $O(|V|^2)$

Ladj: $O(|E| + |V|)$

BFS, DFS: madj, lasj

Cammino semplice: $O(|E| + |V|)$

Hamilton: esponenziale

Grafo Trasposto: $O(|E|+|V|)$

Verifica Eulero: $O(|E|)$

Matrici: $T=O(n^3)$, $S=O(n^2)$

Kruskal: $O(|E| \log |E|)$

Prim: $O(|E| \log |V|)$, heap fibonacci $O(|E| + |V| \log |V|)$

Dijkstra: estrai u da V-S $O(\log |V|)$, inserisci u in S $O(\log |V|)$, rilassa archi $O(|E|)$

$O((|V| + |E|) \log |V|)$, se tutti V raggiungibili da s: $O(|E| \log |V|)$

DAG: $O(|E| + |V|)$

BF: $O(|V| |E|)$

BST: $O(\log n) < O(h) < O(n)$

Linear chaning: ricerca: $O(1+\alpha)$, cancell e inser $O(1)$

Linear probing: miss: $\frac{1+\frac{1}{(1-\alpha)^2}}{2}$, hit: $\frac{1+\frac{1}{1-\alpha}}{2}$

Double hashing: miss: $\frac{1}{\alpha \ln(\frac{1}{1-\alpha})}$, hit: $\frac{1}{1-\alpha}$

Disp sempl: $T(N)=NT(N-1)+1$;

Disp ripet:

Permut sempl: $T(N)=NT(N-1)+1$;

Permut ripet:

Combin sempl: $T(n)=\sum_0^n T(n-i) + 1$;

Combin ripet:

[NB. Sono tutte esponenziali quelle di calc. Comb.]