

Un algoritmo $O(n \log n)$ è sicuramente...

Risposta corretta

- $O(n^2)$



Una classe di complessità temporale contiene:

Risposta corretta

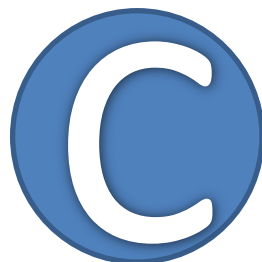
- **tutti i linguaggi riconoscibili entro una quantità di tempo fissata**



**Se il problema P
ammette lower bound
 $\Omega(n^2)$ allora**

Risposta corretta

- nessun algoritmo lo risolve
in tempo $O(n)$**



Un automa a stati finiti

Risposta corretta

- è un dispositivo formale per il riconoscimento di linguaggi regolari



Il problema della terminazione:

Risposta corretta

- è un tipico esempio di problema non decidibile



L'altezza minima di un albero di n nodi è

Risposta corretta

• 1



Se l'algoritmo A calcola la funzione $a(x)$ in tempo $O(f_A(|x|))$ (avendo indicato con $|x|$ il numero di bit necessari e sufficienti a rappresentare x) e l'algoritmo B calcola la funzione $b(x)$ in tempo $O(f_B(|x|))$, allora esiste un algoritmo C che calcola la funzione $a(b(x))$ in tempo:

Risposta corretta

- $f_B(|x|) + f_A(|b(x)|)$



Una tautologia è:

Risposta corretta

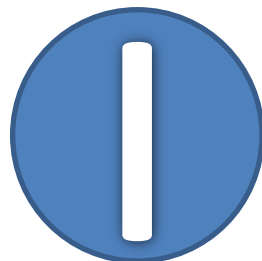
- una formula proposizionale che è soddisfatta da tutti i modelli



In un grafo semplice e non orientato di n nodi il massimo numero di spigoli è

Risposta corretta

- $n*(n-1)/2$



Il linguaggio

$\{a^n b^n \mid n \geq 1\}$ è:

Risposta corretta

- un classico esempio di linguaggio context-free ma non regolare



Le macchine a registri

Risposta corretta

- sono in grado di calcolare le stesse funzioni calcolabili con macchine di Turing



Nel caso di array già ordinato, qual è il più conveniente fra i seguenti algoritmi di ordinamento?

Risposta corretta

- **Insertion-sort**

