

**Alcuni prompt per ChatGPT, utili per approfondimenti della lezione 4 di Programmazione I e Lab P I,
(Lezioni P1-10-01, P1-10-03)**

prof. Giunta

Agisci come un tutor universitario di un corso di programmazione di primo livello. Supponi di aver finora analizzato il numero di operazioni principali di un algoritmo. Adesso si vuole formalizzare tale analisi introducendo il concetto di complessità di tempo e di complessità di spazio. Non parlare ancora di notazione asintotica $O(\cdot)$

Adesso introduci e definisci formalmente la notazione asintotica $O(\cdot)$. Fai alcuni esempi.

Aiutami a determinare la complessità di tempo dell'algoritmo di determinazione del massimo elemento di un array (operazione dominante: confronto tra elementi dell'array). Poi usa la notazione asintotica per descrivere tale complessità spiegando il ragionamento.

Aiutami a determinare la complessità di tempo dell'algoritmo di determinazione contemporanea del massimo e del minimo elemento di un array (operazione dominante: confronto tra elementi dell'array). Poi usa la notazione asintotica per descrivere tale complessità spiegando il ragionamento.

Aiutami a determinare la complessità di tempo dell'algoritmo di determinazione del secondo massimo elemento di un array (operazione dominante: confronto tra elementi dell'array). Non usare l'approccio semplice, ma determina direttamente il secondo massimo. Poi usa la notazione asintotica per descrivere tale complessità spiegando il ragionamento.

Utilizzando la complessità asintotica è possibile classificare gli algoritmi in classi di complessità. Considera anche la complessità lin-log, esponenziale e fattoriale. Fai alcuni esempi.

Discuti la conclusione che, di fatto, gli algoritmi a complessità più che polinomiale (e in particolare esponenziale e fattoriale) non possono essere usati nelle applicazioni pratiche.

Spiega il problema del commesso viaggiatore. E' vero che gli unici algoritmi per risolvere esattamente il tale problema sono a complessità esponenziale o fattoriale?

Discuti brevemente quello che ritieni essere il più semplice algoritmo euristico, cioè che restituisce una soluzione vicina alla soluzione vera, per il problema del commesso viaggiatore e analizza la sua complessità di tempo.

Scrivi una function C che implementa tale algoritmo dell'euristica del vicino più vicino. Non usare l'allocazione dinamica.

Sei in grado di calcolare la soluzione esatta del problema descritto nel main del precedente codice?

Descrivi in modo semplice e breve il problema teorico noto come $P = NP?$.

Perché un problema NP non può essere definito solo come problema per cui si conoscono solo algoritmi risolutivi a complessità più che polinomiale?

Ma un problema in P è certamente in NP ?

Quindi si tratta di capire l'insieme differenza NP-P è vuoto o non vuoto?

E naturalmente NP è incluso nell'insieme dei problemi che possono essere verificati in tempo più che polinomiale.

Spiega quali sarebbero le principali conseguenze se si dimostrasse che P coincide con NP.

Spiega anche quali sarebbero le principali conseguenze se si dimostrasse che P non coincide con NP.