

**Alcuni prompt per ChatGPT, utili per approfondimenti della lezione 4 di Programmazione I e Lab P I,  
(Lezioni P1-14-01, P1-14-04)**

**prof. Giunta**

Agisci come un tutor universitario di un corso di programmazione di primo livello. Introduci in generale il concetto di ricorsività, facendo riferimento prima alle definizioni ricorsive in un linguaggio naturale, ti suggerisco la definizione ricorsiva del concetto di Antenato in un linguaggio naturale. Chi sono gli antenati di G ? Gli antenati di G sono suo padre e sua madre più gli antenati di suo padre più gli antenati di sua madre.

Sei in grado di fare altri esempi simili?

Come si descrive la ricorsione in un linguaggio di programmazione C-like? Fai l'esempio del calcolo del fattoriale.

Descrivi come è organizzato lo stack in C per la ricorsione. Facendo riferimento all'algoritmo ricorsivo del fattoriale, esamina in dettaglio il riempimento e lo svuotamento dello stack nel caso  $n=5$ .

Descrivi l'algoritmo ricorsivo per la ricerca binaria, usando il size dell'array come parametro (oltre alla chiave e all'array). Non usare la notazione " ? : " . La function deve restituire 0 se la chiave non è trovata e 1 se la chiave è trovata. Chiama tale function RicercaBinariaRicTF.

Riscrivi la function ricorsiva precedente RicercaBinariaRicTF in modo che essa restituisca -1 se la chiave non è trovata e restituisca l'indice (globale) se la chiave è trovata. Calcola l'indice corrente direttamente nella chiamata ricorsiva. Non usare la notazione " ? : " . Chiama tale function RicercaBinariaRic. Descrivi in dettaglio come è calcolato l'indice e qual è il ragionamento.

Considera l'if a 3 vie come operazione principale. Qual è la complessità di tempo dell'algoritmo di ricerca binaria. Non usare la notazione asintotica. E' possibile legare l'analisi di complessità all'albero binario delle decisioni?

Qual è quindi la complessità asintotica?

Esamina in dettaglio il riempimento e lo svuotamento dello stack nel caso dell'algoritmo di ricerca binaria con  $n=8$ .

Descrivi l'algoritmo ricorsivo per la somma degli elementi di una array basato sull'approccio divide et impera, usando il size dell'array come parametro.

Considera la somma come operazione principale. Qual è la complessità di tempo dell'algoritmo per la somma degli elementi di una array basato sull'approccio divide et impera. Non usare la notazione asintotica.

Esamina in dettaglio il riempimento e lo svuotamento dello stack nel caso di una array di 4 elementi.

Scrivi unaa versione ricorsiva della function strlen di string.h.

Esamina in dettaglio il riempimento e lo svuotamento dello stack nel caso dell'algoritmo precedente con una stringa di lunghezza 5.

Scrivi e analizza l'algoritmo ricorsivo di ricerca sequenziale. Usa 0 (falso) e 1 (vero).

Descrivi la dinamica dello stack per l'esempio precedente.

E' possibile sviluppare un algoritmo ricorsivo che restituisca l'indice del target nell'array, nel caso in cui target appartenga all'array?

Scrivi un algoritmo ricorsivo che determina il massimo elemento di un array, basato sulla cosiddetta tail-recursion. Intendi la tail recursion come caso estremo di divide et impera in cui la suddivisione dell'array non avviene a metà ma bensì in una porzione di size  $(n-1)$  e una porzione di size 1. Alla luce di questa definizione, riscrivi l'algoritmo che determina il massimo elemento di un array. Usa solo l'array e il suo size come parametri. Non usare la notazione " ? : " .

Scrivi l'algoritmo di ricerca binaria ricorsiva per un array ordinato di puntatori a stringhe costanti. La chiave è una stringa. Usa il size come parametro.