Limiti asintotici e ricorrenze

Esercizio 1

Si consideri la ricorrenza $T(n) = 3T(n/4) + n^2$. Utilizzando il master theorem, fornire un limite asintotico stretto per la soluzione.

Esercizio 2

Dimostrare che la ricorrenza $T(n) = T(n-1) + n \log n$ ammette soluzione $T(n) = O(n^2)$ utilizzando il metodo di sostituzione.

Esercizio 3

Risolvere la ricorrenza $T(n) = T(n/3) + T(2n/3) + n$ utilizzando il metodo della sostituzione per dimostrare che $T(n) = O(n \log n)$.

Divide et impera e Ricorsione

Esercizio 4

Dato un array A[1..n] di interi, definiamo "punto di equilibrio" un indice i tale che la somma degli elementi a sinistra di i (escluso A[i]) è uguale alla somma degli elementi a destra di i (escluso A[i]).

i. Fornire lo pseudocodice di una procedura ricorsiva divide et impera equilibrio(A) che dato un array A[1..n] restituisce un punto di equilibrio se esiste, -1 altrimenti. ii. Valutare la complessità della funzione, utilizzando il master theorem.

Esercizio 5

Dato un array A[1..n] di interi distinti e ordinati in modo crescente, si definisca "picco locale" un elemento A[i] tale che $A[i-1] < A[i] > A[i+1]$.

i. Mostrare che un array di almeno 3 elementi contiene sempre almeno un picco locale. ii. Fornire lo pseudocodice di una procedura ricorsiva divide et impera picco(A) che dato un array A[1..n] restituisce l'indice di un picco locale. iii. Valutare la complessità della funzione.

Alberi Binari di Ricerca e Alberi e ricorsione

Esercizio 6

Si consideri un albero binario di ricerca i cui nodi x hanno i campi x.key, x.left, x.right e x.sum, dove x.sum contiene la somma delle chiavi nel sottoalbero radicato in x.

i. Scrivere una funzione updateSum(T) che aggiorna correttamente il campo sum di tutti i nodi dell'albero T. ii. Scrivere una funzione insert(T,k) che inserisce una nuova chiave k nell'albero T mantenendo correttamente aggiornato il campo sum. iii. Valutare la complessità delle funzioni realizzate.

Esercizio 7

Si definisca "livello completo" di un albero binario un livello in cui tutti i possibili nodi sono presenti (il livello i può contenere al massimo 2^i nodi). Realizzare una funzione ricorsiva maxComplete(T) che dato un albero binario T determina il massimo livello completo presente nell'albero.

i. Fornire lo pseudocodice della funzione. ii. Dimostrare la correttezza dell'algoritmo. iii. Valutare la complessità della soluzione proposta.

Esercizio 8

Si consideri un albero binario di ricerca i cui nodi x hanno un campo aggiuntivo x.maxPath che contiene la lunghezza del cammino più lungo dalla radice a una foglia nel sottoalbero radicato in x.

i. Scrivere una funzione updateMaxPath(T) che aggiorna correttamente il campo maxPath di tutti i nodi dell'albero T. ii. Scrivere una funzione delete(T,k) che elimina la chiave k dall'albero T mantenendo correttamente aggiornato il campo maxPath. iii. Valutare la complessità delle funzioni realizzate.