

Cognome:..... Nome:..... Matr.:.....

Esercizio 1 [16 punti]

A: notazione asintotica. Dire quali delle seguenti relazioni asintotiche sono vere:

$$\begin{array}{llll} n^{1/4} \log n + \sqrt{\log n} = \Omega(n^{1/3}); & \frac{n}{\log^2 n} = o(\frac{n+3}{\log^4 n}); & \frac{n^3 + \log n}{\sqrt{n}} = \Theta(n^{2.5}); & \sqrt[4]{\log n} = O(\log \log n); \\ 2^{\sqrt{\log n}} = o(n^3); & 2^n = \Theta(2^{n-10}); & 2^{n+2} = \Theta(2^{n/2}); & 2^{2n} = \Theta(4^n + 2^{n/2}); \end{array}$$

B: equazioni di ricorrenza. Fornire la soluzione asintotica alle seguenti relazioni di ricorrenza:

$$T(n) = T(\frac{n}{8}) + 8; \quad \text{Soluzione:}$$

$$T(n) = T(n-1) + n^3; \quad \text{Soluzione:}$$

C: algoritmi e complessità. Quale algoritmo useresti e quanto costa se devi:

- Costruire un albero AVL contenente n chiavi prese in input:
- Ordinare n interi compresi fra 1 e n^4 :
- Dato un BST di n nodi, restituire tutte le chiavi associate ai nodi in ordine crescente:
- In un grafo orientato, capire se c'è un cammino da s a t di al più k archi che passa per uno specifico nodo w :

Esercizio 2 [8 punti]

Sia T un albero binario con n nodi, dove ad ogni nodo v è associato un valore positivo $val(v)$. Si progetti un algoritmo che dato T e un valore Δ , restituisca il numero di nodi di T la cui somma dei valori degli antenati è almeno Δ .

Si assuma che T è rappresentato tramite una struttura dati collegata, con record e puntatori, dove il record di ogni nodo contiene il puntatore al figlio sinistro e al figlio destro del nodo. L'algoritmo deve avere complessità $O(n)$. Si fornisca lo pseudocodice dettagliato.

Esercizio 3 [8 punti]

Sia $A[1 : n]$ un vettore di n bit, dove quindi $A[i] \in \{0, 1\}$ per ogni i . Si progetti una struttura dati che prende in input il vettore A e sia in grado poi di rispondere a query del seguente tipo:

- **Differenza(i, j):** dati due indici $i, j \in \{1, 2, \dots, n\}$ restituisce la differenza in modulo fra il numero di uni e di zeri nel sottovettore $A[i : j]$.

La struttura dati deve poter essere costruita in tempo $O(n)$ e l'algoritmo di query deve richiedere tempo costante. Si forniscano i due pseudocodici dettagliati dell'algoritmo che dato A costruisce la struttura dati, e dell'algoritmo di query.

A: notazione asintotica. Dire quali delle seguenti relazioni asintotiche sono vere:

$$n^{1/4} \log n + \sqrt{\log n} = \Omega(n^{1/3}); \text{ F } \frac{n}{\log^2 n} = o(\frac{n+3}{\log^4 n}); \text{ F } \frac{n^3 + \log n}{\sqrt{n}} = \Theta(n^{2.5}); \text{ F } \sqrt[4]{\log n} = O(\log \log n); \text{ F } \\ 2^{\sqrt{\log n}} = o(n^3); \text{ F } 2^n = \Theta(2^{n-10}); \text{ V } 2^{n+2} = \Theta(2^{n/2}); \text{ F } 2^{2n} = \Theta(4^n + 2^{n/2}); \text{ V }$$

B: equazioni di ricorrenza. Fornire la soluzione asintotica alle seguenti relazioni di ricorrenza:

$$T(n) = T(\frac{n}{8}) + 8;$$

Soluzione:

$$T(n) = T(n-1) + n^3;$$

Soluzione:

$$T(n) = T(\frac{n}{8}) + 8$$

$$n^{\log_8 1} \log_8 n$$

$$n^{\log_2 1} \log_2 n$$

$$T(n) = \log n$$

$$T(n) = T(n-1) + n^3$$

$$T(n) = \Theta(n^4)$$

C: algoritmi e complessità. Quale algoritmo useresti e quanto costa se devi:

- Costruire un albero AVL contenente n chiavi prese in input:
- Ordinare n interi compresi fra 1 e n^4 :
- Dato un BST di n nodi, restituire tutte le chiavi associate ai nodi in ordine crescente:
- In un grafo orientato, capire se c'è un cammino da s a t di al più k archi che passa per uno specifico nodo w :

$$① \text{ n INSERT } \rightarrow T(n) = O(n \cdot \log n)$$

$$② \text{ Bucket sort con } b=2 \rightarrow O(n \cdot \frac{\log k}{\log n}) = O(n)$$

$$③ \text{ visita BFS in ordine simmetrico } \rightarrow O(n+m)$$

$$④ \text{ visita BFS } \rightarrow O(n+m)$$

Esercizio 2 [8 punti]

Sia T un albero binario con n nodi, dove ad ogni nodo v è associato un valore positivo $val(v)$. Si progetti un algoritmo che dato T e un valore Δ , restituisca il numero di nodi di T la cui somma dei valori degli antenati è almeno Δ .

Si assuma che T è rappresentato tramite una struttura dati collegata, con record e puntatori, dove il record di ogni nodo contiene il puntatore al figlio sinistro e al figlio destro del nodo. L'algoritmo deve avere complessità $O(n)$. Si fornisca lo pseudocodice dettagliato.

RICERCA

ATTENTIVO

$\emptyset \rightarrow 1$ non \emptyset ha $val(v) + \sum_{v \in V} val(v)$
o non ancora

QUANT, INFO BASSO + INFO BASSO

RICERCA P(T, v, α, Δ):

IF $v = \text{NULL}$:

RETURN 0

$\alpha = \alpha + val(v)$

$sin = \text{sin}(v)$

$des = \text{des}(v)$

IF $\alpha \geq \Delta$

RETURN $1 + \text{RICERCA P}(T, \text{sin}, \alpha, \Delta) + \text{RICERCA P}(T, \text{des}, \alpha, \Delta)$.

ELSE

RETURN $\text{RICERCA P}(T, \text{sin}, \alpha, \Delta) + \text{RICERCA P}(T, \text{des}, \alpha, \Delta)$

CHIAMATA

$\text{RES} = \text{RICERCA}(T, r, \emptyset, \Delta)$.

3

Esercizio 3 [8 punti]

Sia $A[1 : n]$ un vettore di n bit, dove quindi $A[i] \in \{0, 1\}$ per ogni i . Si progetti una struttura dati che prende in input il vettore A e sia in grado poi di rispondere a query del seguente tipo:

- **Differenza(i, j)**: dati due indici $i, j \in \{1, 2, \dots, n\}$ restituisce la differenza in modulo fra il numero di uni e di zeri nel sottovettore $A[i : j]$.

La struttura dati deve poter essere costruita in tempo $O(n)$ e l'algoritmo di query deve richiedere tempo costante. Si forniscano i due pseudocodici dettagliati dell'algoritmo che dato A costruisce la struttura dati, e dell'algoritmo di query.