

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

### Esercizio 1 [16 punti]

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

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

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:

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

- Costruire un albero AVL contenente  $n$  chiavi prese in input:

$\log(n) \Rightarrow n \log(n)$

- Ordinare  $n$  interi compresi fra 1 e  $n^4$ :

$\text{Radix } T(n) = O(n \frac{\log n^4}{\log n}) = O(n)$

- Dato un BST di  $n$  nodi, restituire tutte le chiavi associate ai nodi in ordine crescente:

$\text{DFS simmetrica } T(n) = O(n)$

- 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  $\Delta$ , restituisca il numero di nodi di  $T$  la cui somma dei valori degli antenati è almeno  $\Delta$ .

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.

$$n^{\frac{1}{4}} \log n + \sqrt{\log n} = \Omega(n^{\frac{1}{3}})$$

$$n^{\frac{1}{4}} \log n + \sqrt{\log n} \geq \Omega(n^{\frac{1}{3}})$$

$$\frac{n}{\log^2 m} = o\left(\frac{n+3}{\log^4 m}\right)$$

$$\lim_{n \rightarrow +\infty} \frac{n}{\log^2 m} \cdot \frac{\log^4 m}{n+3}$$

$$\frac{\cancel{n}}{\cancel{\log^2 m}} \cdot \frac{\log^4 m}{\cancel{n}} \xrightarrow{n \rightarrow +\infty} +\infty \quad \log^2 m$$

$$n^3 \sim \frac{n^3 + \log(n)}{n} = \Theta(n^{\frac{5}{2}})$$

$$n^2 = \Theta(n^{\frac{5}{2}}) \quad F$$

$$\sqrt[4]{\log m} = O(\log(\log(m)))$$

$$\frac{\sqrt[4]{x}}{\log(x)} \xrightarrow{x \rightarrow +\infty} +\infty$$

$$\log(n) = x$$