

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

Esercizio 1 [16 punti]

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

$\checkmark n^{1/5} \log n + \sqrt{\log n} = o(n^{1/4})$; $\checkmark \frac{n}{\log n} = \omega(\frac{n+3}{\log^3 n})$; $\checkmark \frac{n^3 + \log n}{\sqrt{n}} = \Theta(n^{2.5})$; $\checkmark \log \log n = o(\sqrt[4]{\log n})$;
 $2^{\sqrt{\log n}} = o(n^{1/3})$; $\text{F } 2^n = \Theta(2^{n+\log n})$; $\text{F } 2^{n+2} = \Theta(2^{n/2})$; $\checkmark 2^{2n} = \Theta(4^n + n^2)$;

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

$T(n) = 2T(n/4) + \sqrt{n}$; Soluzione: $\Theta(\sqrt{n} \log(n))$

$T(n) = T(n-1) + \sqrt{n}$; Soluzione: $\Theta(n\sqrt{n})$

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

- In un grafo diretto rappresentato con *matrice di adiacenza*, calcolare i nodi raggiungibili da un nodo specifico s :

- In un grafo non orientato e pesato, individuare il nodo a distanza massima da un nodo v :

- In un albero AVL di n nodi, trovare il secondo minimo: $V = \min(C)$, $X = \text{succ}(V)$
 $O(\log(n))$

- In un vettore ordinato, calcolare il numero di nodi di valore minimo: *ricerca binaria*
 $T(n) = O(\log n)$

Esercizio 2 [8 punti]

Sia T un albero binario con n , dove ogni nodo v di T ha un colore $v.col$ che può essere *Blu* (B) o *Giallo* (G), e quindi $v.col \in \{B, G\}$. Diciamo che un nodo v ha *antenati ben colorati* se il cammino dalla radice al nodo v è composto da una sequenza (potenzialmente vuota) di nodi di colore Blu seguita da una sequenza (potenzialmente vuota) di nodi di colore Giallo.¹

Si progetti un algoritmo che dato T , restituisca il numero di nodi di T che hanno antenati ben colorati. Si assuma che T è rappresentato tramite una struttura dati collegata, con record e puntatori, dove il record di ogni nodo, oltre al campo $v.col$, contiene anche 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 $G = (V, E)$ un grafo diretto con n nodi ed m archi. Ci sono Alice e Bob che vogliono incontrarsi in un nodo di G . Inizialmente, Alice si trova sul nodo s_A ed ha a disposizione Δ_A monete di tipo A, mentre Bob si trova sul nodo s_B ed ha a disposizione Δ_B monete di tipo B. Ad ogni arco $e \in E$, sono associati due interi, c_e^A e c_e^B , che rappresentano rispettivamente il numero di monete di tipo A che Alice deve pagare per attraversare e , e il numero di monete di tipo B che Bob deve pagare per poter attraversare e .

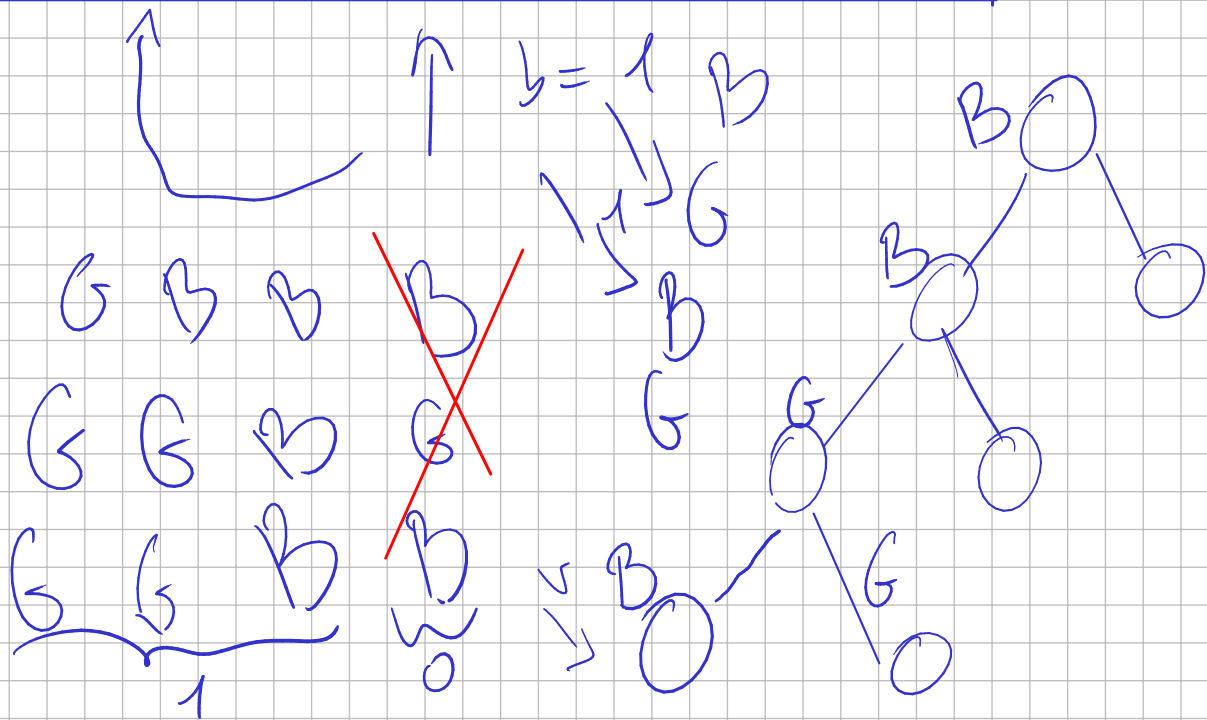
Progettate un algoritmo di complessità $O(m + n \log n)$ che calcola, se esiste, un modo per far incontrare Alice e Bob.

¹Quindi un nodo v che ha tutti gli antenati di colore Giallo o tutti gli antenati di colore Blu ha antenati ben colorati. Si ricorda inoltre che un nodo è per definizione antenato di se stesso.

$v = \min()$

$S_{\min} = \text{succ}(v)$

1	2	>					
2	2	4	6	7	8	10	



$\text{CalcoloColorati}(\text{nodo } v, \text{Color}, \text{bool})$

if $v == \text{null}$ return 0

$Sx = \text{CalcoloColorati}(sx(v), \text{color}, \text{bool})$

$dx = \text{CalcoloColorati}(dx(v), \text{color}, \text{bool})$

if $\text{color} == v.cd$

return $1 + Sx + dx$

```
else v.col  $\neq$  color AND bool == 0  
    color = v.col  
    bool = 1  
    return 1 + 5x + dx
```

```
else return 5x + dx
```