

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

Esercizio 1 [11 punti] Si consideri il problema del mantenimento di insiemi disgiunti attraverso una struttura dati Union-Find.

1. Dire quale delle seguenti affermazioni è vera:

- Nella QuickFind con euristica *union by size* ogni operazione di **union** richiede tempo $O(\log n)$ nel caso peggiore, dove n è il numero di **makeSet**.
- Usando la struttura dati QuickFind con euristica *union by size*, ogni sequenza di n **makeSet**, $n - 1$ **union** e m **find**, richiede nel caso peggiore tempo $O(m + n \log n)$.
- Usando la struttura dati QuickUnion con euristica *union by size*, è sempre garantito che l'albero corrispondente ad ogni insieme ha nel caso peggiore altezza $O(\log n)$, dove n è il numero di **makeSet**.
- Ogni struttura dati, per eseguire una sequenza di n **makeSet**, $n - 1$ **union** e m **find**, deve impiegare nel caso peggiore tempo $\Omega(m + n)$.
- Si assuma di implementare l'algoritmo di Kruskal usando una struttura dati QuickFind con euristica *union by size*. Si assuma inoltre di avere già gli archi del grafo ordinati in ordine non decrescente rispetto al loro peso. Allora l'esecuzione dell'algoritmo di Kruskal ha complessità temporale $O(m + n \log n)$.

2. (a) Si mostri un grafo non orientato e pesato G di almeno 5 nodi e 7 archi il cui unico *Minimum Spanning Tree (MST)* contiene sia l'arco di peso minimo che quello di peso massimo di G . (Max 5 righe.)

(b) Si fornisca una proprietà dell'istanza che caratterizza quando il grafo G ammette un MST contenente l'arco di peso massimo. (Max 5 righe.)

Esercizio 2 [11 punti] Si consideri il problema del calcolo del minimo albero ricoprente (*Minimum Spanning Tree problem*).

1. Si definisca formalmente il problema. (Max 5 righe.)
2. Si enunci formalmente la proprietà del ciclo (*cycle property*). (Max 5 righe.)
3. Si fornisca una dimostrazione della proprietà del ciclo. (Max 10 righe.)

Esercizio 3 [11 punti] (*porcospini colorati*)

Nel nuovo videogioco che avete scaricato sul vostro smartphone siete un porcospino che può cambiare colore e dovete raccogliere il maggior numero di anelli. Ogni livello è una sequenza di n tratte. La tratta i contiene r_i anelli rossi e g_i anelli gialli. Anche il vostro porcospino può colorarsi di rosso o giallo, e in base al colore che avete quando attraversate una tratta raccogliete automaticamente gli anelli del vostro colore. Potete iniziare il gioco con uno dei due colori a vostra scelta. Inoltre, all'inizio di ogni tratta potete decidere se cambiare colore. Questo cambio ha un costo: ogni volta che lo fate perdete α anelli di quelli che avete raccolto, dove $\alpha > 0$ è un parametro del livello.¹

Progettate un algoritmo di programmazione dinamica che, data un'istanza del gioco, calcoli il numero massimo di anelli che potete raccogliere.

¹Quando avete meno di α anelli e decidete di cambiare colore il numero di anelli diventa 0.