

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

Esercizio 1 [11 punti] Si consideri un grafo non orientato e connesso $G = (V, E)$ di n nodi dove ad ogni arco $e \in E$ è associato un peso $w(e)$.

1. Dire quale delle seguenti affermazioni è vera:

- Il peso dell'albero calcolato con l'algoritmo di Kruskal è sempre uguale al peso di quello calcolato con l'algoritmo di Prim.
- Quando il grafo è non pesato, l'algoritmo di Prim restituisce un albero dei cammini minimi radicato sul nodo sorgente su cui è chiamato.
- Se si implementa l'algoritmo di Kruskal con la struttura dati *Quick-Find* senza euristica di bilanciamento *union-by-size*, la complessità dell'algoritmo nel caso peggiore è $O(n^2)$.
- Se T non è un MST di G allora esiste un arco $e \in T$ e un arco $f \notin T$ tale che e è l'arco più pesante del ciclo che si forma quando si aggiunge f a T .
- Se T non è un MST di G allora esiste un arco e che non appartiene a T tale che e è l'arco più leggero del ciclo che si forma in T quando si aggiunge e .

2. Sia G un grafo non orientato e pesato con pesi distinti. Si consideri la seguente affermazione: *Se l'arco e appartiene all'unico MST di G , allora e è l'arco di peso minimo di tutti i tagli che attraversa.* Dire se l'affermazione è vera o falsa argomentando la risposta. (Max 5 righe.)

Esercizio 2 [11 punti] Si consideri il problema del massimo flusso.

1. Si definisca formalmente il problema. (Max 5 righe.)

2. Si enunci la complessità temporale dell'algoritmo di Ford-Fulkerson, argomentando sulla sua polinomialità o meno. (Max 5 righe.)

Esercizio 3 [11 punti] I tuoi amici Alice e Bob ti hanno invitato a cena a casa loro e avete ordinato una teglia di pizza con gusti misti. La pizza arriva in un cartone rettangolare che contiene n pezzi di pizza già tagliati allineati in una lunga striscia. Non tutti i gusti ti piacciono allo stesso modo. Se mangi il pezzo i -esimo (immagina i pezzi numerati da sinistra a destra) godi g_i . Ora, quando prendi un pezzo di pizza, per questioni di praticità, puoi scegliere solo uno dei due pezzi esterni (quelli interni si porterebbero via troppa mozzarella dai pezzi adiacenti e i tuoi amici non sarebbero contenti). Tu sei ospite e sei invitato per primo a scegliere il primo pezzo da mangiare. Una volta scelto il pezzo, Alice e Bob si mangeranno i due più esterni fra quelli che restano, e poi toccherà ancora te scegliere. Se per esempio tu decidi di mangiare il pezzo numero 1 (quello più a sinistra), Alice e Bob mangeranno il numero 2 e il numero n . Mentre se scegli il pezzo più a destra, loro mangeranno il numero 1 e il numero $n - 1$. E così via.

Progetta un algoritmo di programmazione dinamica che calcola il massimo godimento ottenibile, dove il godimento che ottieni da un sottoinsieme di pezzi $S \subseteq \{1, 2, \dots, n\}$ è definito come

$$g(S) = \sum_{i \in S} g_i.$$

Si discuta la complessità temporale dell'algoritmo proposto.