Algoritmi e Strutture Dati (modulo II) - testo prova scritta 22/06/2022

docenti: Luciano Gualà & Andrea Clementi

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 $w(e)$. Si esegua l'algoritmo di Prim su G a partire da un nodo $s \in V$ e sia T l'albero risultante.

1. Dire quale delle seguenti affermazioni è vera:

- T è un MST di G solo se G ha pesi distinti.
- T è un MST di G.
- T è un albero dei cammini minimi di G con sorgente s se i pesi sono distinti.
- T è un albero dei cammini minimi di G con sorgente s se tutti gli archi pesano 1.
- T è un albero dei cammini minimi di G con sorgente s se G è un albero.

2. Si consideri il caso in cui G è *non connesso*. In questo caso l'albero T ritornato dall'algoritmo non è un MST di tutto G. Cosa è T e quanti archi ha? (Max 3 righe)

Esercizio 2 [11 punti] Si consideri il problema di massimizzazione *Interval Scheduling* il cui input è un insieme di n job dove il job i-esimo ha un tempo di inizio s_i e un tempo di fine f_i. Si consideri l'algoritmo ottimo $\mathcal{A}$ basato sull'approccio *greedy*.

1. Si dica quale è l'ordine con cui $\mathcal{A}$ considera i job. (Max una riga.)
2. Si mostri che il criterio greedy secondo cui i migliori job sono quelli che durano di meno (ovvero che minimizzano $f_i - s_i$) in generale non consente di trovare una soluzione ottima del problema. (Max 3 righe.)
3. Si enunci in modo formale e preciso la proprietà chiave che permette di dimostrare che $\mathcal{A}$ è un algoritmo ottimo per Interval Scheduling. (Max 5 righe.)

Esercizio 3 [11 punti] Avete davanti a voi una pila di n monete dove la moneta i-esima a partire dall'alto ha valore v_i. Giocate un gioco. Le mosse a disposizione sono di due tipi:

- `PrimaDiTre`: Vi mettete in tasca la moneta in cima alla pila ma dovete poi rimuovere e buttare le successive due monete (questa mossa può essere eseguita solo se la pila ha almeno 3 monete);
- `SecondaDiDue`: Rimuovete e buttate la prima moneta e vi mettete in tasca la successiva (questa mossa può essere eseguita solo se la pila ha almeno 2 monete).

Ovviamente siete interessati a trovare una sequenza di mosse che vi fa guadagnare il più possibile. Progettate un algoritmo di programmazione dinamica che calcola il massimo guadagno.