

Analisi e Progetto di Algoritmi

a.a. 2005/06

Compito del 7/6/2006

Cognome: _____ Nome: _____

Matricola: _____ E-mail: _____

Desidero sostenere la prova orale (barrare la casella corrispondente alla scelta desiderata):

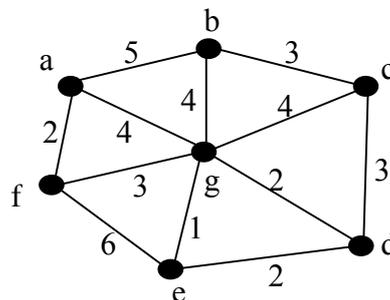
il 12/13 giugno il 28/29 giugno

Parte I

1. Sia G un grafo non orientato con n vertici e m archi. Si determini quale delle seguenti affermazioni è vera e quale è falsa (fornendo dimostrazioni nel primo caso e controesempi nel secondo):
 - (a) Se G è connesso, allora $n \leq m+1$;
 - (b) Se $n \leq m+1$, allora G è connesso;
 - (c) Se $n > m+1$, allora G ha almeno due componenti connesse;
 - (d) Se G è connesso, allora G è completo;
 - (e) Se G è completo, allora G è connesso.
2. Si scriva un algoritmo di complessità $O(n^2 \log n)$ per determinare le distanze tra tutte le coppie di vertici in una foresta G avente pesi sugli archi positivi, dove n è il numero di vertici in G .
3. Data una rete di flusso $G=(V,E)$, siano f_1, f_2 e f_3 tre flussi in G e sia h la funzione da $V \times V$ in \mathbf{R} definita come segue: $h(u,v) = [f_1(u,v) + f_2(u,v) + f_3(u,v)]/3$, per tutti gli $u,v \in V$. Stabilire se h è un flusso in G , giustificando "tecnicamente" la risposta.
4. Si supponga di aver determinato un algoritmo polinomiale per CICLO-HAMILTONIANO. Possiamo dire qualcosa riguardo alla complessità computazionale di SODDISFATTIBILITA'? E cosa potremmo dire sulla complessità di SODDISFATTIBILITA` se l'algoritmo polinomiale lo avessimo trovato invece per ISOMORFISMO-DI-GRAFI?

Parte II

1. Si scriva l'algoritmo di Kruskal, si discuta la sua complessità computazionale e lo si utilizzi (simulandone accuratamente l'esecuzione) per determinare **due** alberi di copertura minimi nel seguente grafo:



2. Sia G un grafo orientato pesato in cui gli archi che escono dalla sorgente s possono avere pesi negativi, tutti gli altri pesi degli archi non sono negativi e non esistono cicli di peso negativo. Si dimostri che l'algoritmo di Dijkstra trova correttamente i cammini minimi da s in questo grafo.
3. Si scriva un algoritmo polinomiale per determinare una clique massimale in un grafo non orientato, se ne dimostri la correttezza e si discuta la sua complessità computazionale. L'algoritmo proposto è in grado di determinare *sempre* una clique massima? In caso affermativo si fornisca una dimostrazione, altrimenti un controesempio. Infine, si mostri che è improbabile che esista un algoritmo polinomiale per determinare la clique massima in un grafo, dimostrando che CLIQUE è un problema NP-completo.