9 Cammini Minimi

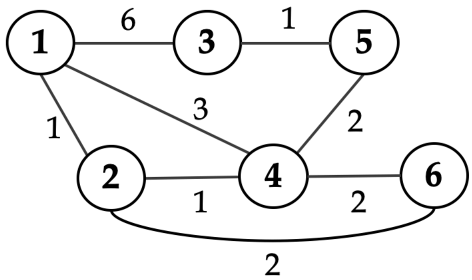
9.1 Verifica di una soluzione

Assumete di avere un vettore d indicizzato sui nodi di un grafo G di distanze minime (da un nodo noto s) e un albero T radicato in s, T sottografo ricoprente di G rappresentato come vettore dei padri.

Dare un algoritmo che verifica se T e d sono effettivamente una soluzione del problema dei cammini minimi per G.

9.2 Diametro, distanze e Dijkstra (IV esonero 7/6/22)

Considerare il grafo in figura:



1. Determinare le distanze minime (in numero di archi) dal nodo 1 a tutti gli altri nodi, e un albero di cammini di lunghezza minima (in numero di archi) radicato nel nodo 1;

2. determinare una radice (diversa da 1) per cui l?albero dei cammini minimi abbia un'altezza maggiore di quello radicato nel nodo 1;

3. Il diametro di un grafo è definito come $\max_{u,v \in V} d(u, v)$, dove $d(u, v)$ è la distanza in numero di archi di u da v. Calcolare il diametro di G;

4. fornire un algoritmo per calcolare il diametro di un grafo;

5. considerando i pesi scritti sugli archi, determinare i pesi dei cammini minimi (in peso) dal nodo 1 a tutti gli altri nodi e fornire l'albero dei cammini minimi, indicando l'ordine con cui l'algoritmo di Dijkstra costruisce tale albero.

9.3 Dijkstra e pesi negativi

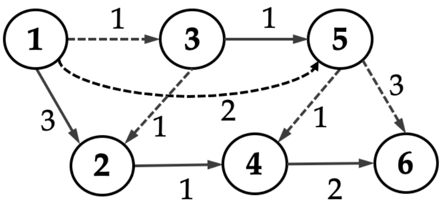
Supponete di avere un un grafo G con pesi sugli archi w, anche negativi. Calcolate $W = min_{e \in E} w(e)$ e sostituite tutti i pesi $w(e)$ con $w'(e) = w(e) + |W|$.

L'algoritmo di Dijkstra usando w' come pesi, restituisce risultati corretti rispetto al problema originario? Fornire un controesempio. Quali cammini vengono "privilegiati"?

9.4 Cammino superminimo (esame 20/6/22)

In un grafo pesato G, un cammino tra due nodi u e v è detto *superminimo* se tra tutti i cammini di peso minimo tra u e v aventi ugual peso, ha anche lunghezza minima (in numero di archi).

Esempio: Nel grafo in figura, sono evidenziati tutti i cammini superminimi (archi tratteggiati) dal nodo 1 a tutti gli altri nodi. Ad esempio il cammino $1 \to 5$ ha peso 2 esattamente come il cammino $1 \to 3 \to 5$, ma è superminimo in quanto più corto. Similmente, il cammino $1 \to 5 \to 6$ ha peso 5 esattamente come il cammino $1 \to 5 \to 4 \to 6$, ma è superminimo in quanto più corto.



1. Assumendo tutti i pesi degli archi *interi positivi*, trovare il modo di usare l'algoritmo di Dijkstra in modo che calcoli un albero di cammini superminimi da un nodo sorgente s a tutti gli altri nodi. Discutere brevemente la correttezza.

2. Come si può rilassare il vincolo che i pesi siano interi? Quale pre-computazione (e di quale complessità) è necessaria per verificare l'applicabilità delle condizioni che avete trovato? Nel caso siano soddisfatte, spiegare come si applica la tecnica del punto precedente.

3. La tecnica discussa nei due punti precedenti, si può applicare anche all'algoritmo di Bellman-Ford?