

Problem Set 2

Da consegnare entro il 20 Gennaio agli indirizzi
guala@mat.uniroma2.it, isabella.ziccardi@graduate.univaq.it

Esercizio 1 Il prof. Gualà deve andare alla cena della vigilia a casa di suo fratello ma è in ritardo perché sta preparando il Problem Set di natale per gli studenti del corso di Algoritmi e Strutture Dati. Proprio quando sta scrivendo l'esercizio su di lui che scrive l'esercizio¹ si ricorda che ha promesso a suo fratello di portare il vino.

Ovviamente, il mondo del prof. Gualà è modellato come un grafo $G = (V, E)$ diretto e pesato con n nodi e m archi, dove ogni arco $e \in E$ ha associato un tempo di percorrenza $w(e)$. La casa di suo fratello si trova nel nodo t mentre la sua è nel nodo s . Il vino può essere comprato solo in un sottoinsieme $U \subseteq V$ di nodi e per ogni nodo $u \in U$ il prof. Gualà conosce il tempo $\tau(u)$ che impiegherebbe se comprasse il vino sul nodo u . Progettate un algoritmo con complessità $O(m + n \log n)$ che calcola il minimo tempo necessario al vostro professore per andare a casa di suo fratello munito di vino.

Esercizio 2 In Australia a Babbo Natale per le consegne hanno messo a disposizione un carro trainato da canguri. E questo crea non pochi problemi. La mappa dell'Australia è modellata come un grafo diretto $G = (V, E)$. Babbo Natale deve andare dal nodo s al nodo t . La cosa non sarebbe complicata se non fosse che i canguri possono solo fare *salti da 3*. Per esempio, se Babbo Natale si trova in un generico nodo u e c'è l'arco (u, v) nel grafo, non è detto che Babbo Natale possa spostarsi sul nodo v (quello poteva farlo quando usava la slitta con le renne, che scivolava su nodi adiacenti senza problemi). Infatti, i canguri possono spostarsi con un salto da u a v se e solo se esiste un cammino in G da u a v di lunghezza esattamente 3.

Progettate e analizzate un algoritmo efficiente che determina se esiste una sequenza di salti che porta Babbo Natale da s a t .

Esercizio 3 Per natale vi viene regalato un difficilissimo puzzle mongolo, inventato dal guerriero Vidrach Itky Leda nel 1473. Questo puzzle consiste in una scacchiera numerata di dimensione $n \times n$ dove si trovano due pedine, una rossa e una blu. Le due pedine, all'inizio, sono posizionate rispettivamente in alto a sinistra ed in basso a destra. Durante tutto il gioco, non devono mai occupare la stessa posizione. L'obiettivo del gioco è di scambiare di posto le due pedine.

In un singolo turno, ciò che potete fare è spostare una delle due pedine in alto, in basso, a destra o a sinistra ma solo della distanza determinata dalla posizione dall'altra pedina. Ad esempio, se la pedina rossa si trova sulla casella di valore 3, potrà muovere la pedina blu in alto, in basso, a destra o a sinistra di tre caselle. Non si può muovere

¹...su di lui che scrive l'esercizio su di lui che scrive l'esercizio su di lui che scrive l'esercizio... La leggenda narra che Il prof. Gualà per un momento infinito resta sospeso nel loop ricorsivo fino a che non pensa che manca il caso base. Questo interrompe la ricorsione, cosa che permetterà al Problem Set di essere concluso in tempo (finito).

una pedina fuori dalla scacchiera, e alla fine di una mossa le due pedine non possono trovarsi nella stessa casella.²

Si dà il caso che è proprio il natale in cui state seguendo il corso di Algoritmi e Strutture Dati. Così decidete, fra una fetta di panettone e un metro cubo di lasagna, di progettare un algoritmo efficiente che calcola la sequenza più corta di mosse che risolve il puzzle, o che riporta correttamente che non c'è soluzione. Un esempio di istanza e soluzione sono date in Figura 1.

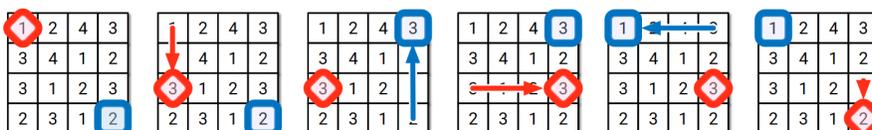


Figura 1: Una soluzione in cinque mosse del puzzle 4×4 di Vidrach Itky Leda puzzle.

Esercizio 4 Babbo Natale è nella vostra città per consegnarvi il regalo. La città è modellata come un grafo non orientato pesato $G = (V, E)$ con n nodi, dove ad ogni arco $e \in E$ è associata una lunghezza $w(e)$. Babbo Natale si trova sul nodo s e casa vostra è sul nodo t . Se tutte le strade (archi) fossero libere, Babbo Natale userebbe il cammino minimo per andare da s a t . Lui non ha tempo da perdere. È più efficiente di un corriere Amazon. Il problema è che in città c'è un mercatino di natale itinerante che si sposta da una parte all'altra, ogni volta occupando una singola strada/arco su cui Babbo Natale non può passare (non sono ammesse né macchine né renne ai mercatini).

Se il mercatino sta occupando la strada $e \in E$, Babbo Natale dovrà usare un cammino che non passa per l'arco e . Questo si chiama *cammino alternativo* per e . Ovviamente, lui vuole usare il più corto cammino alternativo per e . Ora, Babbo Natale non sa quando partirà per consegnare il vostro regalo, ma prima di farlo guarderà su Google in quale strada si trova il mercatino, e quindi saprà solo all'ultimo momento la strada e che deve evitare. Questo esercizio vi chiede di aiutare Babbo Natale. In particolare:

- Dimostrate che esiste un sottografo H di G che ha solo $O(n)$ archi e che contiene i più corti cammini alternativi che Babbo Natale potrebbe dover usare, ovvero, H contiene, per ogni arco $e \in E$, il più corto (in G) cammino alternativo per e .
- Progettate una struttura dati (a volte chiamata *oracolo*) di dimensione $O(n)$ che, per ogni e , è in grado di riportare in tempo costante la lunghezza del più corto cammino P_e alternativo per e , e di restituire P_e in tempo proporzionale al numero di archi di P_e .³

²In una variante del gioco le mosse devono essere alternate, prima la pedina rossa, poi la blu, poi ancora la rossa eccetera. In un'altra non ci sono vincoli sui turni e potete muovere più volte la stessa pedina. Entrambe le varianati sono divertenti.

³Si noti che queste complessità sono ottime, in quanto per restituire una rappresentazione esplicita di P_e è necessario spendere tempo proporzionale al numero di archi di P_e . Inoltre, si osservi che se e non appartiene al cammino minimo fra s e t , P_e coincide proprio con quest'ultimo.

- Progettare un algoritmo efficiente che costruisce l'oracolo.⁴

Suggerimento: Si ragioni sulla struttura del miglior cammino alternativo per un generico arco e e si dimostri che la sua lunghezza è pari a $d_G(s, x) + w(x, y) + d_G(y, t)$, dove (x, y) è un opportuno arco di G e $d_G(\cdot, \cdot)$ denota la distanza in G fra due nodi.

⁴Esiste una soluzione non complicata con complessità $O(mn + n^2 \log n)$ e una difficile che ha tempo di soluzione $O(m + n \log n)$, dove n ed m sono rispettivamente il numero di nodi e archi di G .