

Grafopoli

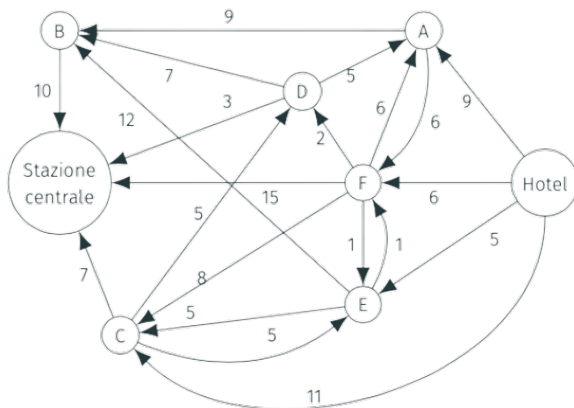
Livello di difficoltà: intermedio

Parole chiave

- Teoria dei grafi
- Grafi diretti
- Algoritmo di Dijkstra
- Algoritmi euristici
- Programmazione Lineare Intera
- Excel Solver
- Python+PuLP

Descrizione del problema

Per Remo è arrivato il giorno di lasciare Grafopoli e di tornare a casa. Il treno di ritorno partirà dalla Stazione Centrale alle 11:03 ma Remo, come sempre in ritardo, riesce a lasciare l'Hotel solo alle 10:50. Si considerano affidabili i tempi di percorrenza (in minuti) riportati nel grafo sottostante, rappresentante la rete cittadina.



Quesiti

1. Qual è il percorso che permetterebbe a Remo di risparmiare più tempo possibile? Ce n'è solo uno o più di uno? Perché?

2. Remo avrebbe quindi una possibilità di arrivare in tempo in stazione per prendere il treno delle 11:03? Se sì, con quanti minuti di anticipo potrebbe arrivare? Se no, con quanti minuti di ritardo?
3. Si supponga ora che Remo, scoraggiato dal ritardo, decida di trascorrere ancora qualche ora a visitare la città di Grafopoli. In particolare, ogni nodo del grafo rappresenta un punto di interesse da visitare (e.g., un museo, una piazza, un monumento, etc.). In tabella sono riportati i tempi stimati di visita per ogni attrazione e gli eventuali orari di chiusura, oltre il quale non è più possibile accedervi (se c'è il “–”, allora l'attrazione è sempre aperta):

	A	B	C	D	E	F
Tempo di visita	20	5	30	60	10	20
Orari di chiusura	12:00	–	13:00	13:30	–	–

Supponendo che Remo parta dall'hotel alle ore 11:00, quale cammino può fare per visitare più attrazioni possibili e non arrivare in stazione dopo le 14:00?

- (a) Formulare e descrivere passo per passo (i.e., input, output e tutti gli step) un algoritmo che risolva il problema proposto.
 - (b) Trovare una possibile soluzione per il problema applicando l'algoritmo sviluppato.
4. Formulare ora il problema di partenza come un problema di Programmazione Lineare Intera e risolverlo con l'ausilio di Excel Solver.
5. Implementare il modello matematico in Python e risolverlo sfruttando la libreria PuLP.