NOTA SULLA SOLUZIONE DELL'ESERCIZIO 4

Una soluzione molto semplice consiste nell'adattare l'algoritmo di Dijkstra per il calcolo della lunghezza dei cammini minimi da un vertice s a tutti gli altri.
L'algoritmo di Dijkstra (v. pseudocodice Lezione 30 sul sito del corso) quando da un vertice u con distanza D[u] (definitiva) ispeziona ciascun vertice v adiacente, controlla se raggiungendo v tramite l'arco (u,v) si possa diminuire il costo D[v] già calcolato:

```
   if D[u] + ω(u, v) < D[v] then
       D[v] ← D[u] + ω(u, v)
```

Nell'algoritmo modificato, considerando la regola per il calcolo dei pedaggi, è sufficiente sostituire

```
    max(D[u], ω(u, v))
```

al posto di D[u] + ω(u, v), sia nella condizione che nell'assegnamento.

Una soluzione alternativa può essere ottenuta DIMOSTRANDO che, con la regola data per calcolo dei pedaggi, i cammini più economici formano un albero ricoprente di peso minimo (nel senso usuale) del grafo che rappresenta la rete autostradale.
Si può dunque utilizzare l'algoritmo di Kruskal o Prim per ottenere un albero ricoprente di costo minimo. Una volta ottenuto l'albero, per ricavare i pedaggi più economici dalla località A alle altre località, si può utilizzare un algoritmo di visita dell'albero ottenuto a partire da A.
Nel caso si utilizzi Prim, i pedaggi possono anche essere calcolati durante la costruzione dell'albero, se la costruzione avviene a partire da A.

NB
Nell'esercizio, è anche richiesto di scrivere lo pseudocodice. Si raccomanda di seguire lo stile e il livello di dettaglio utilizzati a lezione e negli esempi presentati sul sito del corso.