Università degli Studi di Roma "Tor Vergata"
Laurea in Informatica

Sistemi Operativi e Reti
(modulo Reti)
a.a. 2024/2025

Esercitazione: Livello di Rete
(piano di controllo)

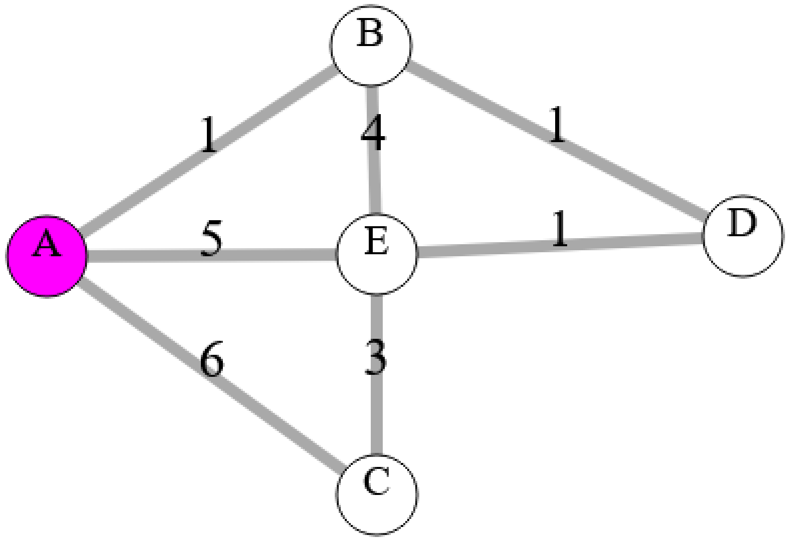
dr. Manuel Fiorelli

manuel.fiorelli@uniroma2.it

https://art.uniroma2.it/fiorelli

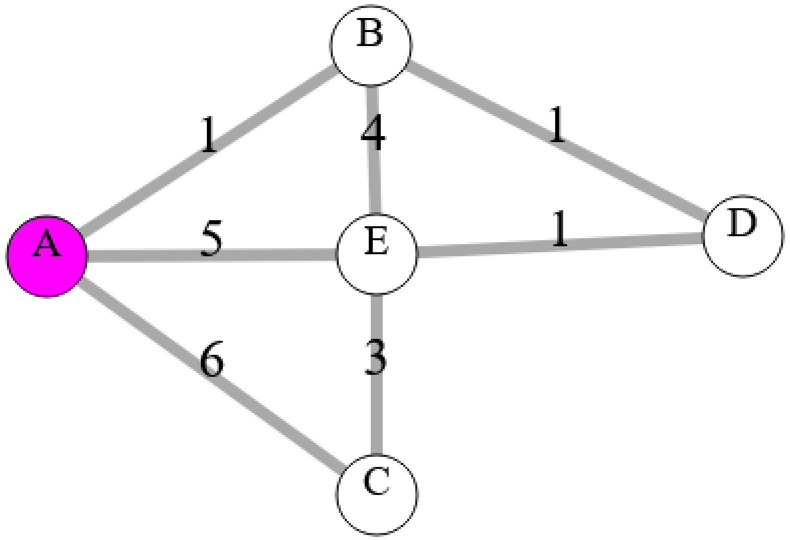
Esercizio 1

Calcolare la tabella di inoltro del router A usando l'algoritmo di Dijkstra



Esercizio 1 (soluzione)

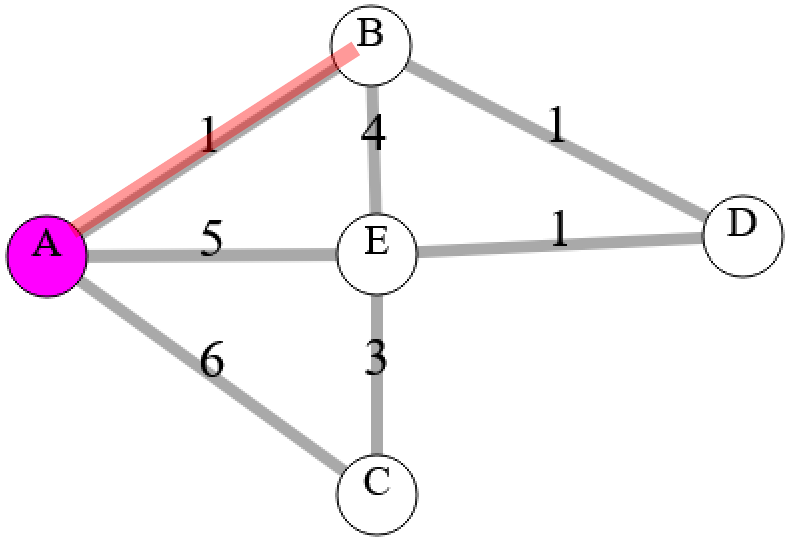
Calcolare la tabella di inoltro del router A usando l'algoritmo di Dijkstra



Passo	N'	*D*(B), *p*(B)	*D*(C), *p*(C)	*D*(D), *p*(D)	*D*(E), *p*(E)
0	A	1, A	6, A	+∞	5, A

Esercizio 1 (soluzione)

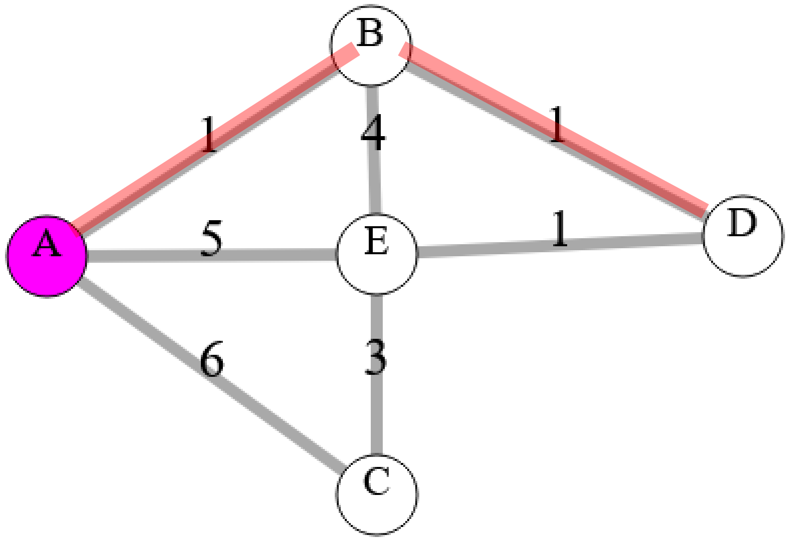
Calcolare la tabella di inoltro del router A usando l'algoritmo di Dijkstra



Passo	N'	*D*(B), *p*(B)	*D*(C), *p*(C)	*D*(D), *p*(D)	*D*(E), *p*(E)
0	A	1, A	6, A	+∞	5, A
1	AB		6, A	2, B	5, A

Esercizio 1 (soluzione)

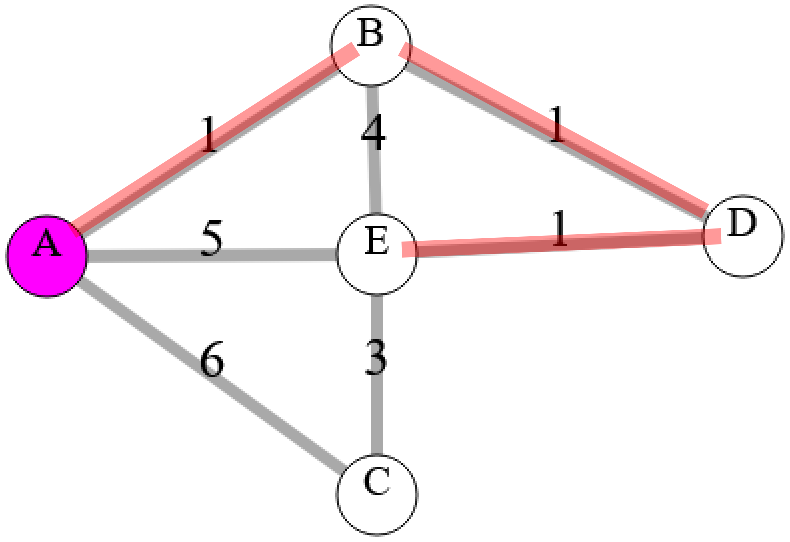
Calcolare la tabella di inoltro del router A usando l'algoritmo di Dijkstra



Passo	N'	*D*(B), *p*(B)	*D*(C), *p*(C)	*D*(D), *p*(D)	*D*(E), *p*(E)
0	A	1, A	6, A	+∞	5, A
1	AB		6, A	2, B	5, A
2	ABD		6, A		3, D

Esercizio 1 (soluzione)

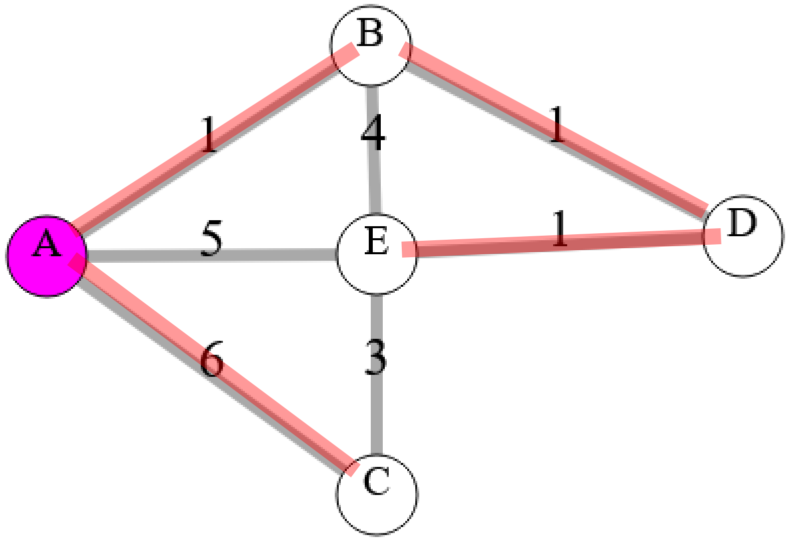
Calcolare la tabella di inoltro del router A usando l'algoritmo di Dijkstra



Passo	N'	*D*(B), *p*(B)	*D*(C), *p*(C)	*D*(D), *p*(D)	*D*(E), *p*(E)
0	A	1, A	6, A	+∞	5, A
1	AB		6, A	2, B	5, A
2	ABD		6, A		3, D
3	ABDE		6, A		

Esercizio 1 (soluzione)

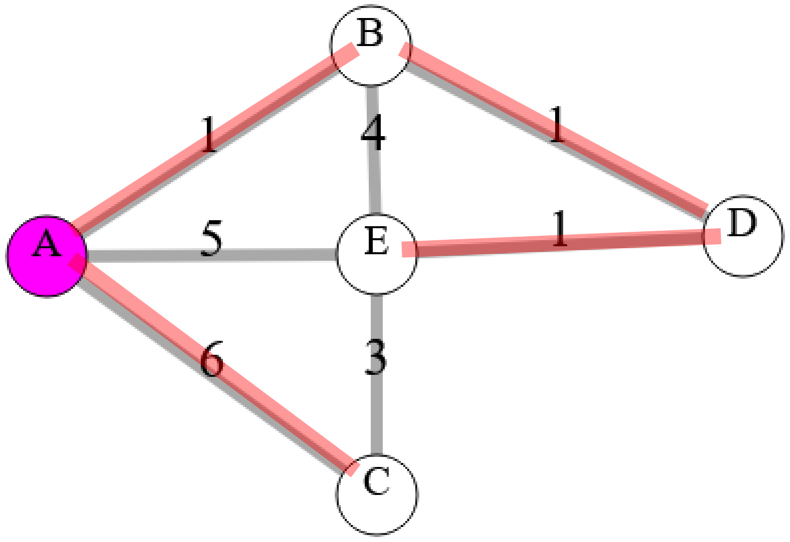
Calcolare la tabella di inoltro del router A usando l'algoritmo di Dijkstra



Passo	N'	*D*(B), *p*(B)	*D*(C), *p*(C)	*D*(D), *p*(D)	*D*(E), *p*(E)
0	A	1, A	6, A	+∞	5, A
1	AB		6, A	2, B	5, A
2	ABD		6, A		3, D
3	ABDE		6, A		
4	ABDEC				

Esercizio 1 (soluzione)

Calcolare la tabella di inoltro del router A usando l'algoritmo di Dijkstra

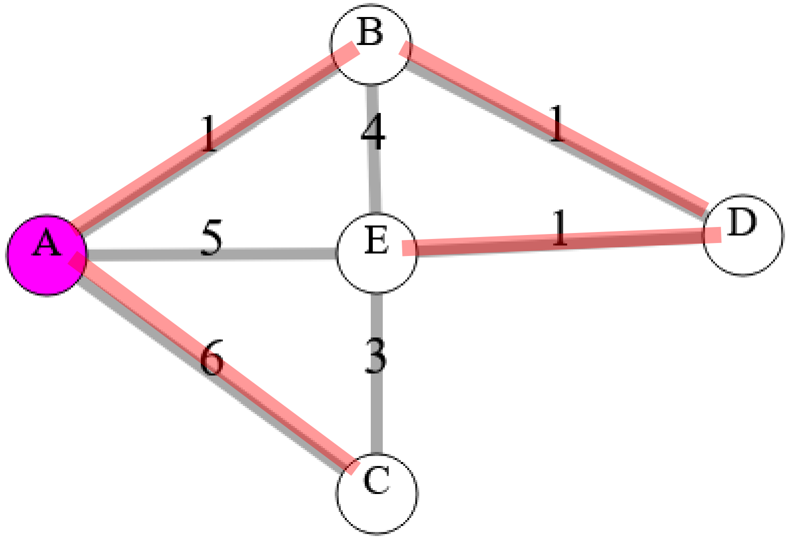


Quando l'algoritmo termina, abbiamo un albero dei cammini di costo minino dal router A verso ciascuna destinazione.
Per costruire la tabella di inoltro, per ciascuna destinazione, consideriamo il primo collegamento lungo il percorso corrispondente.

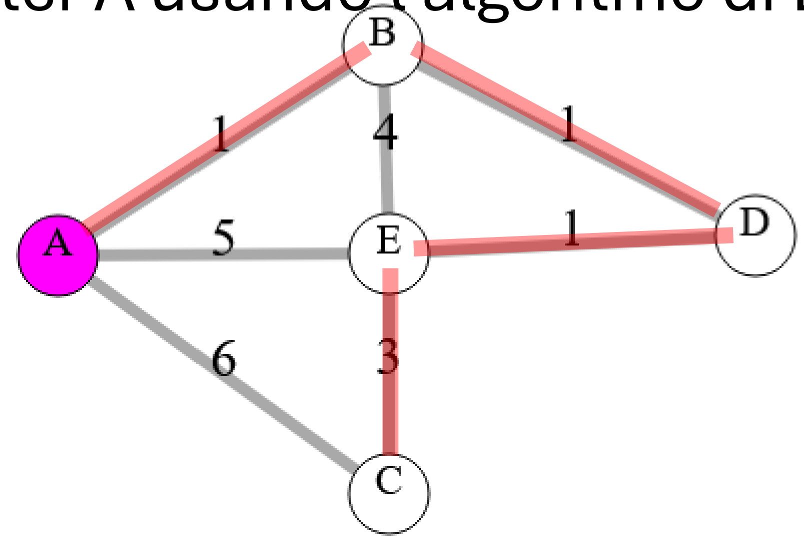
Destinazione	Collegamento
B	(A,B)
C	(A,C)
D	(A,B)
E	(A,B)

Esercizio 1 (soluzione)

Calcolare la tabella di inoltro del router A usando l'algoritmo di Dijkstra



Destinazione	Collegamento
B	(A,B)
C	(A,C)
D	(A,B)
E	(A,B)



altra soluzione

Destinazione	Collegamento
B	(A,B)
C	(A,B)
D	(A,B)
E	(A,B)

Esercizio 1 (soluzione)

Tool *online* che vi fornisce la soluzione passo a passo per il calcolo dei cammini minimi con l'algoritmo di Dijkstra:

https://mdahshan.github.io/dijkstra/

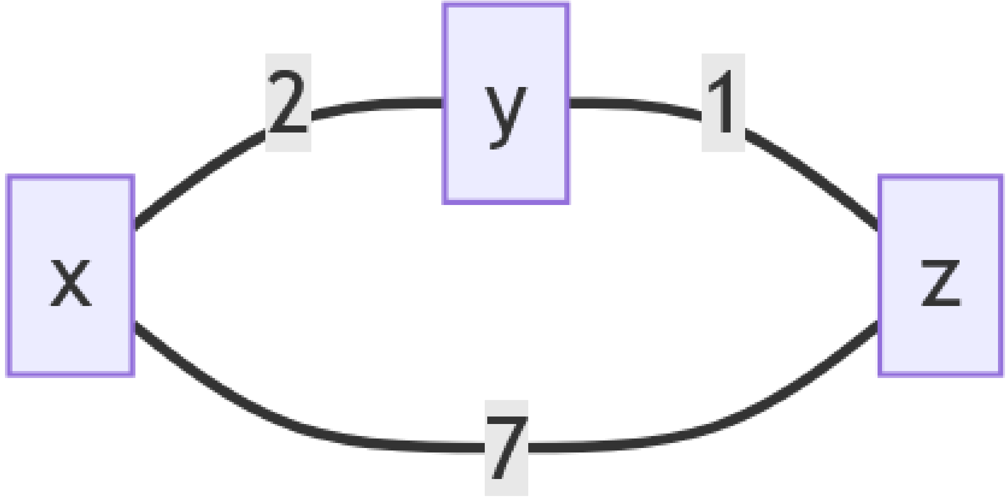
Nota: il tool produce una tabella leggermente diversa, ma può essere facilmente adattata al formato usato dal libro.

Esercizi interattivi (generati dinamicamente):

- https://gaia.cs.umass.edu/kurose_ross/interactive/dij.php

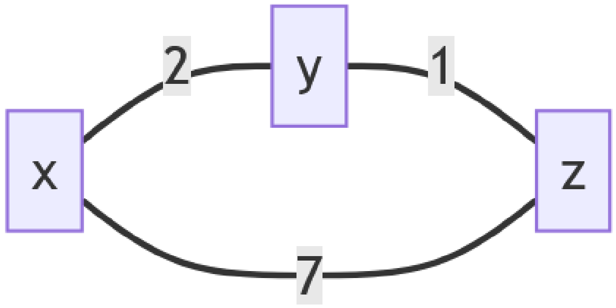
Esercizio 2

Calcolare la tabella di inoltro del router x usando l'algoritmo Distance Vector.



(Soluzione)

Esercizio 2



nodo x

costo verso

da

	x	y	z
x	0 (x)	2 (y)	7 (z)
y	∞	∞	∞
z	∞	∞	∞

nodo y

costo verso

da

	x	y	z
y	2 (x)	0 (y)	1 (z)
x	∞	∞	∞
z	∞	∞	∞

nodo z

costo verso

da

	x	y	z
z	7 (x)	1 (y)	0 (z)
y	∞	∞	∞
x	∞	∞	∞

Un qualsiasi router α gestisce un vettore delle distanze $D_\alpha(\beta)$ [vedi prima riga di ciascuna tabella] che fornisce la distanza da α verso ogni altra destinazione β. Ovviamente, $D_\alpha(\alpha) = 0$.
Inizialmente α conosce soltanto la distanza verso i propri vicini.
Un nodo mantiene una copia dell'ultimo vettore delle distanze ricevuto dai vicini. [vedi altre righe di ciascuna tabella].

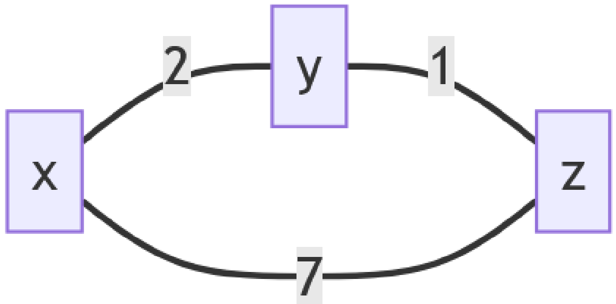
All'inizio ogni router invia il proprio vettore delle distanze ai vicini.

Simuleremo l'esecuzione dell'algoritmo distribuito per istanti discreti, assumendo che gli aggiornamenti inviati in un dato istante arrivino tutti nell'instante successivo, quando ogni nodo utilizza tali informazioni per ricalcolare il proprio vettore delle distanze usando l'equazione di Bellman-Ford. Se il vettore cambia, il nodo invia gli aggiornamenti ai vicini (vedi teoria!).

t

(Soluzione)

Esercizio 2



nodo x

da \ costo verso	x	y	z
x	0 (x)	2 (y)	7 (z)
y	∞	∞	∞
z	∞	∞	∞

da \ costo verso	x	y	z
x	0 (x)	2 (y)	***3 (y)***
y	∞	***0***	***1***
z	∞	***1***	***0***

$$\min\{c_{x,y} + D_y(y), c_{x,z} + D_z(y)\} = \min\{2 + 0, 7 + 1\} = 2\ (y)$$

$$\min\{c_{x,y} + D_y(z), c_{x,z} + D_z(z)\} = \min\{2 + 1, 7 + 0\} = 3\ (y)$$

nodo y

da \ costo verso	x	y	z
y	2 (x)	0 (y)	1 (z)
x	∞	∞	∞
z	∞	∞	∞

da \ costo verso	x	y	z
y	2 (x)	0 (y)	1 (z)
x	***0***	∞	***7***
z	***7***	∞	***0***

$$\min\{c_{y,x} + D_x(x), c_{y,z} + D_z(x)\} = \min\{2 + 0, 1 + 7\} = 2\ (x)$$

$$\min\{c_{y,x} + D_x(z), c_{y,z} + D_z(z)\} = \min\{2 + 7, 1 + 0\} = 1\ (z)$$

nodo z

da \ costo verso	x	y	z
z	7 (x)	1 (y)	0 (z)
y	∞	∞	∞
x	∞	∞	∞

da \ costo verso	x	y	z
z	***3 (y)***	1 (y)	0 (z)
y	***2***	***0***	∞
x	***0***	***2***	∞

$$\min\{c_{z,y} + D_y(x), c_{z,x} + D_x(x)\} = \min\{1 + 2, 7 + 0\} = 3\ (y)$$

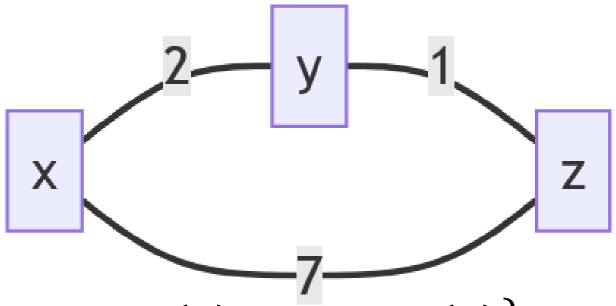
$$\min\{c_{z,y} + D_y(y), c_{z,x} + D_x(y)\} = \min\{1 + 0, 7 + 2\} = 1\ (y)$$

I router x e z hanno calcolato un vettore delle distanze differente e pertanto devono inviare un aggiornamento ai vicini.
Si presti attenzioni agli infiniti inseriti a causa della inversione avvelenata.

t

(Soluzione)

Esercizio 2



nodo x

da \ costo verso	x	y	z
x	0 (x)	2 (y)	7 (z)
y	∞	∞	∞
z	∞	∞	∞

da \ costo verso	x	y	z
x	0 (x)	2 (y)	***3 (y)***
y	∞	***0***	***1***
z	∞	***1***	***0***

da \ costo verso	x	y	z
x	0 (x)	2 (y)	3 (y)
y	∞	0	1
z	***3***	1	0

$$\min\{c_{x,y} + D_y(y), c_{x,z} + D_z(y)\} = \min\{2+0, 7+1\} = 2\ (y)$$

$$\min\{c_{x,y} + D_y(z), c_{x,z} + D_z(z)\} = \min\{2+1, 7+0\} = 3\ (y)$$

nodo y

da \ costo verso	x	y	z
y	2 (x)	0 (y)	1 (z)
x	∞	∞	∞
z	∞	∞	∞

da \ costo verso	x	y	z
y	2 (x)	0 (y)	1 (z)
x	***0***	∞	***7***
z	***7***	∞	***0***

da \ costo verso	x	y	z
y	2 (x)	0 (y)	1 (z)
x	0	∞	***∞***
z	***∞***	∞	0

$$\min\{c_{y,x} + D_x(x), c_{y,z} + D_z(x)\} = \min\{2+0, 1+\infty\} = 2\ (x)$$

$$\min\{c_{y,x} + D_x(z), c_{y,z} + D_z(z)\} = \min\{2+\infty, 1+0\} = 1\ (z)$$

nodo z

da \ costo verso	x	y	z
z	7 (x)	1 (y)	0 (z)
y	∞	∞	∞
x	∞	∞	∞

da \ costo verso	x	y	z
z	***3 (y)***	1 (y)	0 (z)
y	***2***	***0***	∞
x	***0***	***2***	∞

da \ costo verso	x	y	z
z	3 (y)	1 (y)	0 (z)
y	2	0	∞
x	0	2	***3***

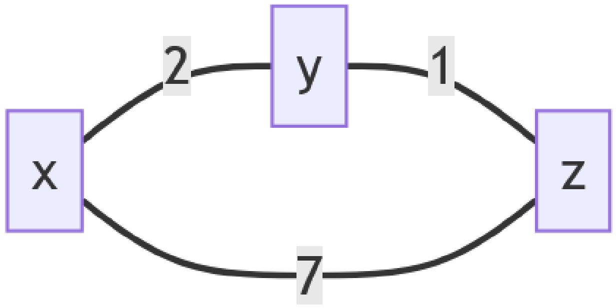
$$\min\{c_{z,y} + D_y(x), c_{z,x} + D_x(x)\} = \min\{1+2, 7+0\} = 3\ (y)$$

$$\min\{c_{z,y} + D_y(y), c_{z,x} + D_x(y)\} = \min\{1+0, 7+2\} = 1\ (y)$$

t

(Soluzione)

Esercizio 2



nodo x

costo verso

da	x	y	z
x	0 (x)	2 (y)	7 (z)
y	∞	∞	∞
z	∞	∞	∞

costo verso

da	x	y	z
x	0 (x)	2 (y)	***3 (y)***
y	∞	***0***	***1***
z	∞	***1***	***0***

costo verso

da	x	y	z
x	0 (x)	2 (y)	3 (y)
y	∞	0	1
z	***3***	1	0

nodo y

costo verso

da	x	y	z
y	2 (x)	0 (y)	1 (z)
x	∞	∞	∞
z	∞	∞	∞

costo verso

da	x	y	z
y	2 (x)	0 (y)	1 (z)
x	***0***	∞	***7***
z	***7***	∞	***0***

costo verso

da	x	y	z
y	2 (x)	0 (y)	1 (z)
x	0	∞	***∞***
z	***∞***	∞	0

nodo z

costo verso

da	x	y	z
z	7 (x)	1 (y)	0 (z)
y	∞	∞	∞
x	∞	∞	∞

costo verso

da	x	y	z
z	***3 (y)***	1 (y)	0 (z)
y	***2***	***0***	∞
x	***0***	***2***	∞

costo verso

da	x	y	z
z	3 (y)	1 (y)	0 (z)
y	2	0	∞
x	0	2	***3***

t

Non è cambiato alcun vettore delle distanze.
Pertanto l'algoritmo termina.

Il vettore delle distanze del router x ci permette di determinare la tabella di inoltro cercata.

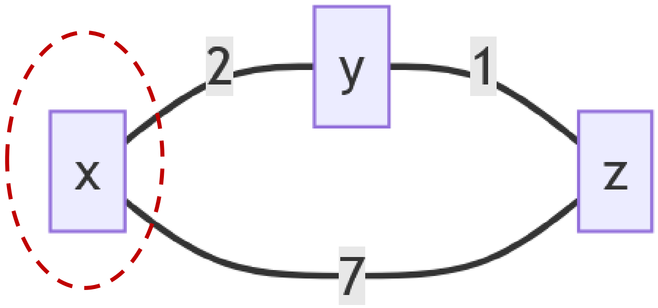
Destinazione	Collegamento
y	(x,y)
z	(x,y)

(Soluzione)

Esercizio 2

Osservazione 1: in ogni singolo router, per calcolare la distanza (e il *next hop*) per ciascuna destinazione è sufficiente la colonna corrispondente (e il costo dei collegamenti diretti di quel router)

$$\min\{c_{x,y} + D_y(y), c_{x,z} + D_z(y)\}$$
$$= \min\{2 + 0, 7 + 1\} = 2\ (y)$$

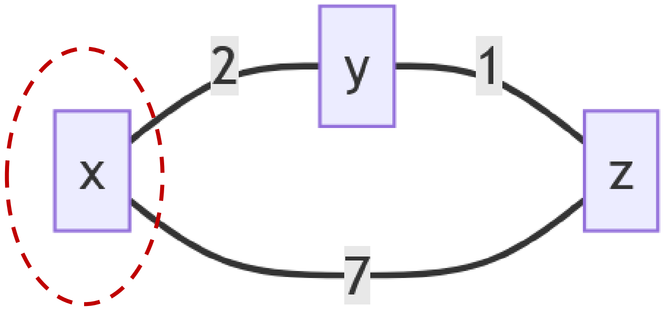


da	x	y	z
x	0 (x)	2 (y)	3 (y)
y	∞	0	1
z	*3*	1	0

$$\min\{c_{x,y} + D_y(z), c_{x,z} + D_z(z), \}$$
$$= \min\{2 + 1, 7 + 0\ \} = 3\ (y)$$

(Soluzione)

Esercizio 2



Osservazione 2: altro modo per comprendere i calcoli presso un router (nell'esempio *x*)

costo verso

da		x	y	z
	x	0 (x)	2 (y)	3 (y)
	y	∞	0	1
	z	*3*	1	0

$$\min \begin{pmatrix} \cancel{\infty} & 2 & 3 \\ \cancel{10} & 8 & 7 \end{pmatrix} = \begin{bmatrix} \cancel{10} & 2 & 3 \end{bmatrix}$$

sommo a tutte le componenti di ciascuna riga il costo del collegamento dal router corrente, in questo caso x, al vicino corrispondente alla riga, qui in generale α,

$$c_{x,\alpha} + D_{alpha}(\beta)$$

(ovviamente la colonna corrispondente al router x può essere trascurata, perché nel vettore delle distanze ci mettiamo sempre 0).

Ciascuna riga così modificata fornisce per ciascuna destinazione la distanza relativa all'instradamento il cui next hop è α.

Il vettore delle distanze di x può essere determinato calcolando il minimo per colonne (quindi determinando l'instradamento di costo minimo per ciascuna destinazione).

Esercizio 2 (Soluzione)

Tool *online* che vi fornisce la soluzione passo a passo per il calcolo dei cammini minimi con l'algoritmo Distance Vector:

https://jasoncarloscox.com/creations/dv-sim/

Esercizi interattivi (generati dinamicamente):

- https://gaia.cs.umass.edu/kurose_ross/interactive/disVector.php