

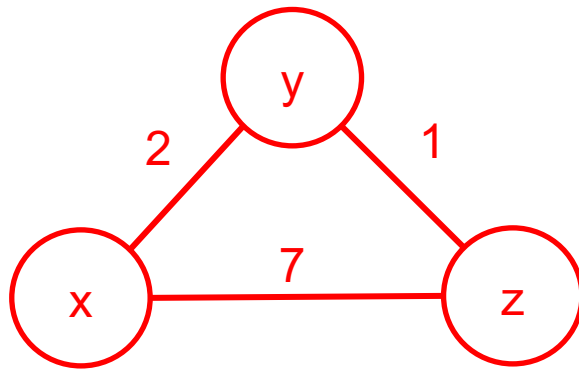
Redes de Computadores II



Temas: Tabelas de Roteamento.

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.



$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

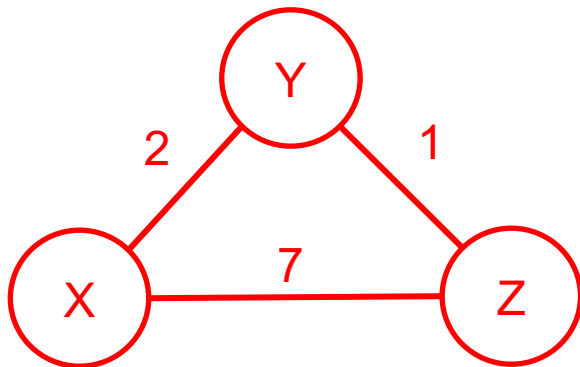


Tabela do nó X

	X	Y	Z
X			
Y			
Z			

t_0

Tabela do nó Y

	X	Y	Z
X			
Y			
Z			

t_0

Tabela do nó Z

	X	Y	Z
X			
Y			
Z			

t_0

* Cada instante de tempo representará cada iteração do processo.

**Cada iteração ocorre com os envios (anúncios) das rotas de cada nó aos seus vizinhos.

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

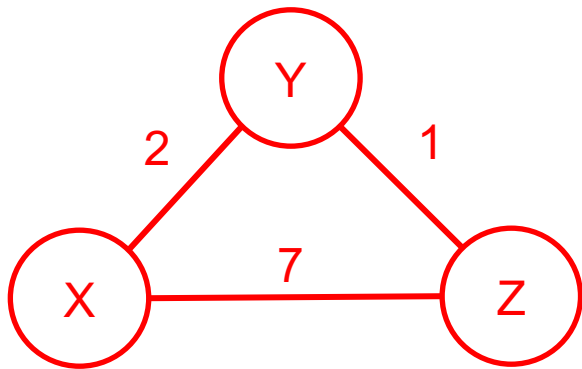


Tabela do nó X

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

* Cada instante de tempo representará cada iteração do processo.

**Cada iteração ocorre com os envios (anúncios) das rotas de cada nó aos seus vizinhos.

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

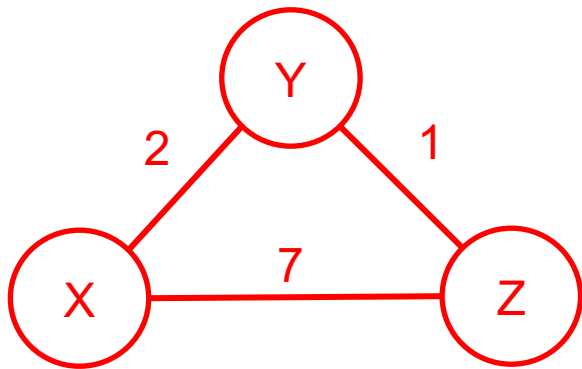


Tabela do nó X

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

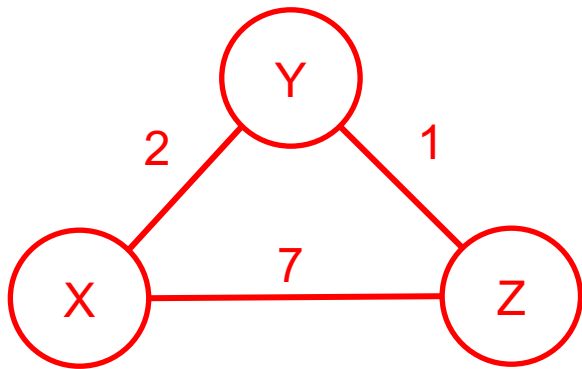


Tabela do nó X

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(x) = \min\{c(x, x) + d_x(x)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

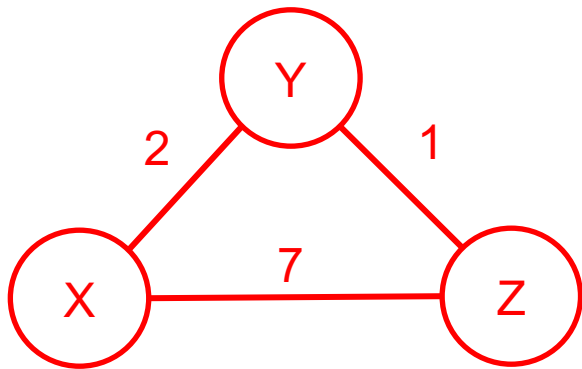


Tabela do nó X

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(x) = \min\{c(x, x) + d_x(x)\}$$

$$d_x(x) = \min\{0 + 0\} = 0$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

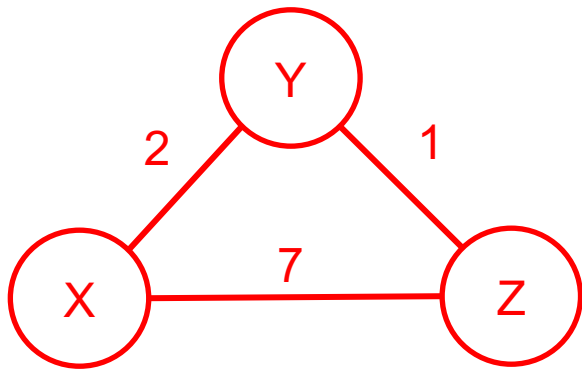


Tabela do nó X

	X	Y	Z
X	0	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(x) = \min\{c(x, x) + d_x(x)\}$$

$$d_x(x) = \min\{0 + 0\} = 0$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

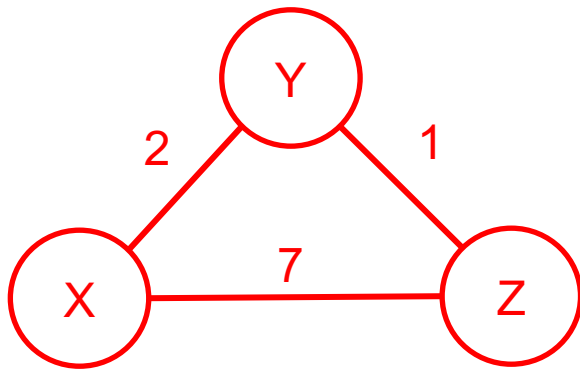


Tabela do nó X

	X	Y	Z
X	0	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

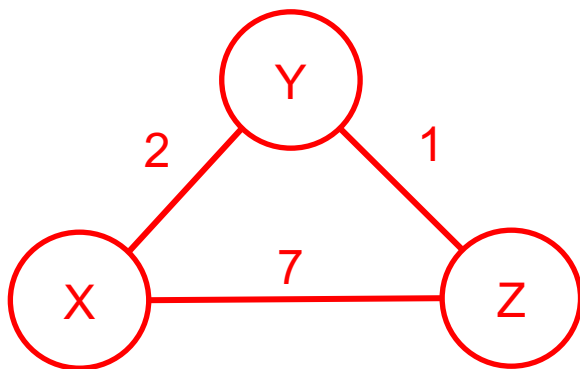


Tabela do nó X

	X	Y	Z
X	0	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

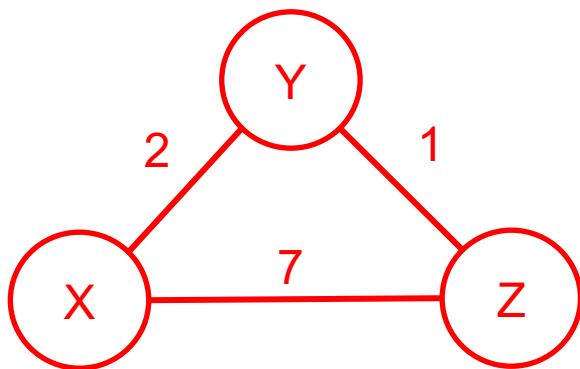


Tabela do nó X

	X	Y	Z
X	0	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

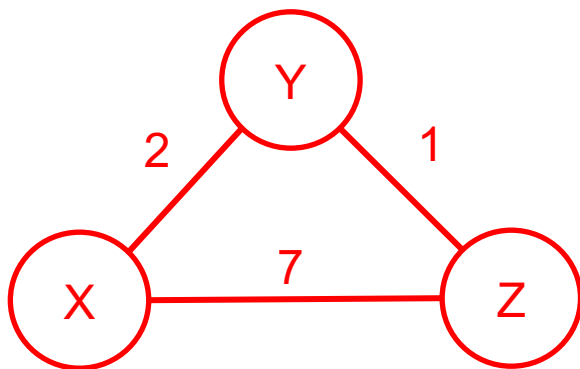


Tabela do nó X

	X	Y	Z
X	0	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + \mathbf{d_z(y)}\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

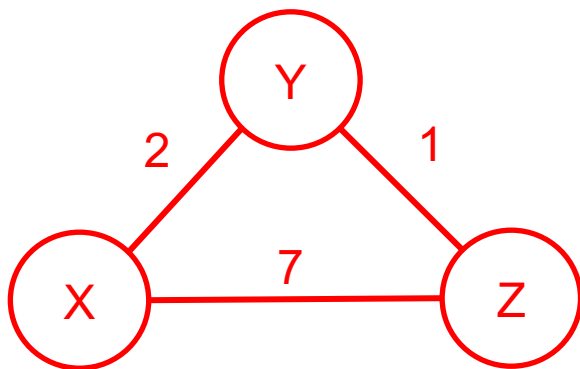


Tabela do nó X

	X	Y	Z
X	0	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + \infty\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

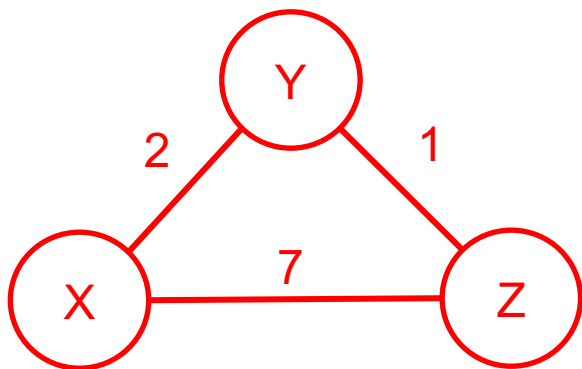


Tabela do nó X

	X	Y	Z
X	0	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + \infty\}$$

$$d_x(y) = \min\{2, \infty\} = 2$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

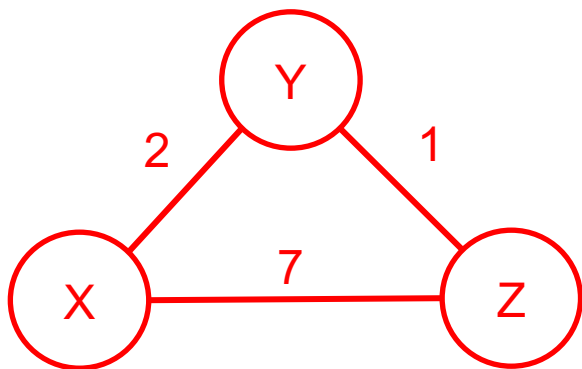


Tabela do nó X

	X	Y	Z
X	0	2	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

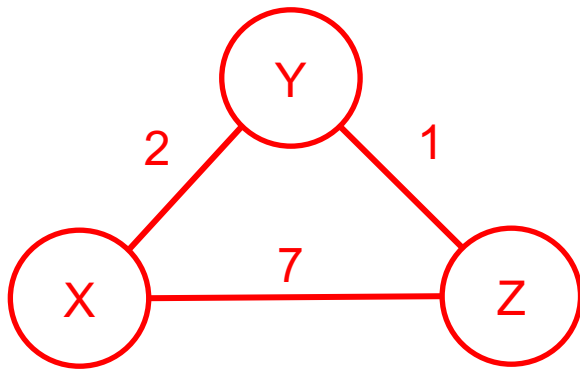
$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + \infty\}$$

$$d_x(y) = \min\{2, \infty\} = 2$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.



$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

Tabela do nó X

	X	Y	Z
X	0	2	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

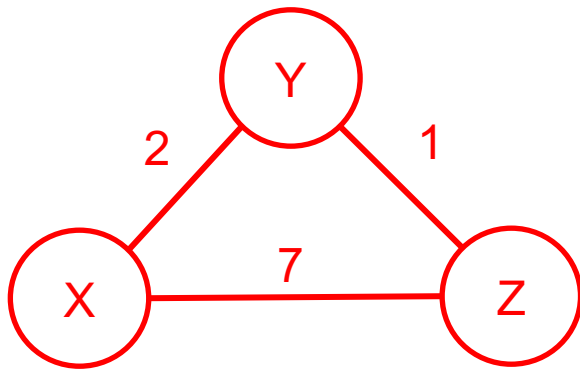


Tabela do nó X

	X	Y	Z
X	0	2	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

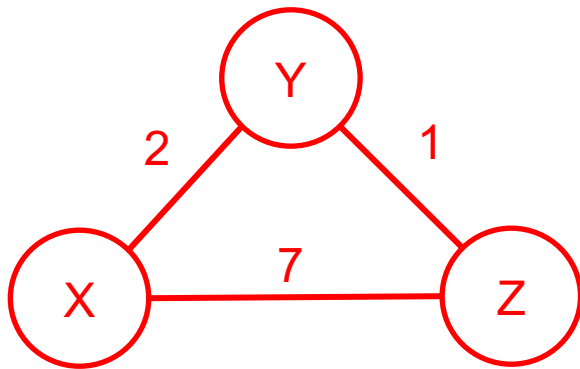


Tabela do nó X

	X	Y	Z
X	0	2	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

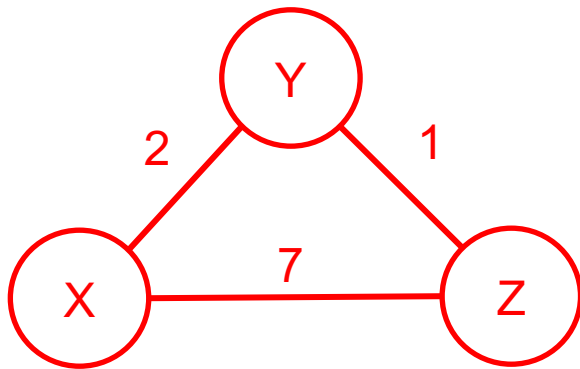


Tabela do nó X

	X	Y	Z
X	0	2	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

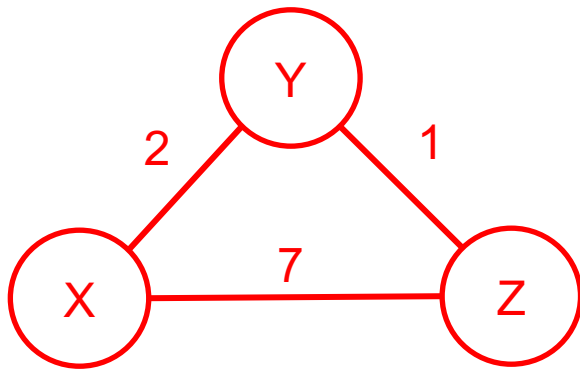


Tabela do nó X

	X	Y	Z
X	0	2	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

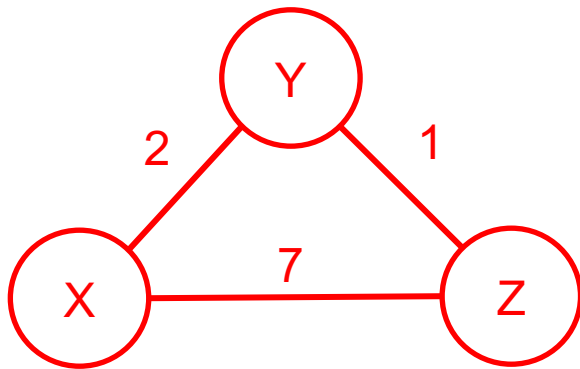


Tabela do nó X

	X	Y	Z
X	0	2	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

$$d_x(z) = \min\{2 + \infty, 7 + 0\} = \min\{\infty, 7\} = 7$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

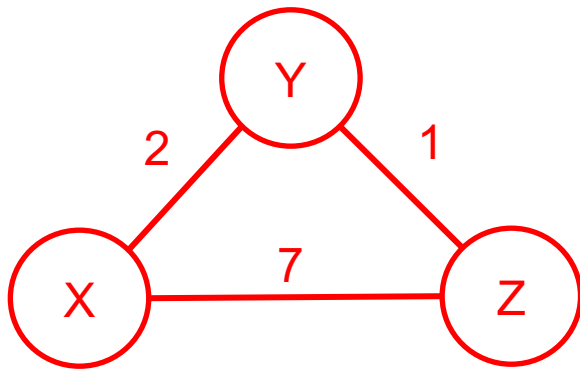


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

$$d_x(z) = \min\{2 + \infty, 7 + 0\} = \min\{\infty, 7\} = 7$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

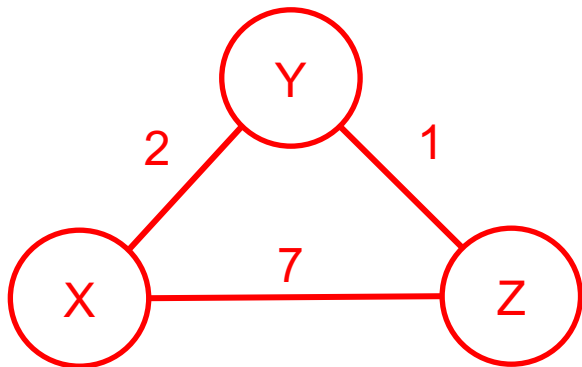


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

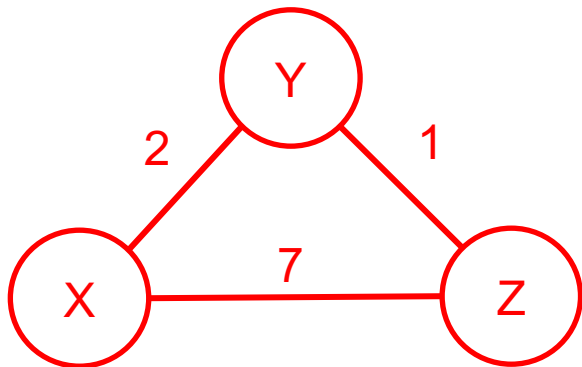


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	∞
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

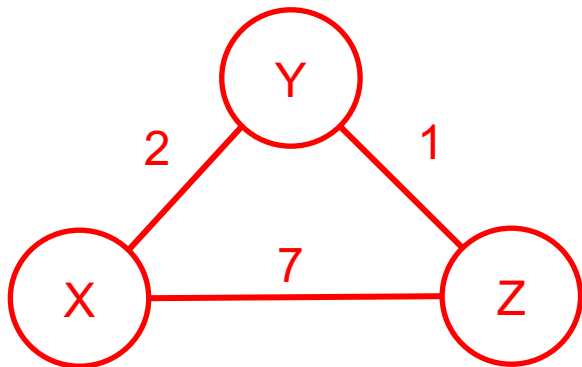


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

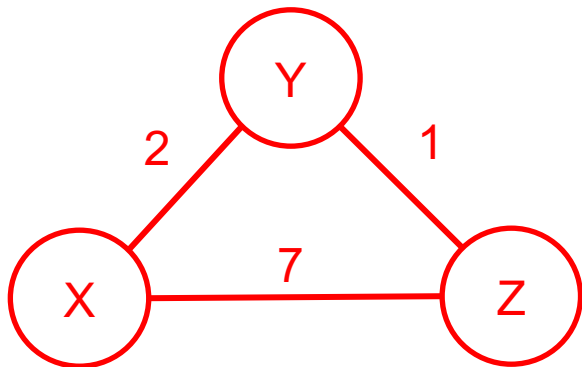


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	∞	∞

t_0

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

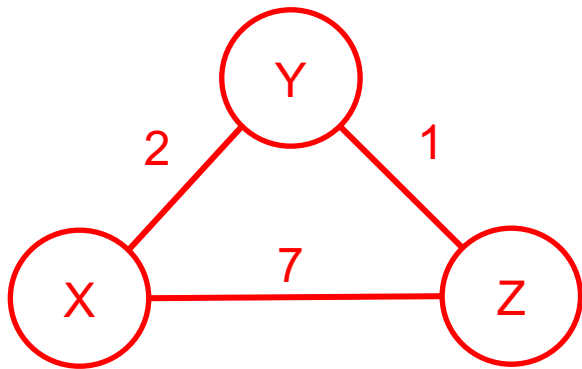


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	∞

t_0

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

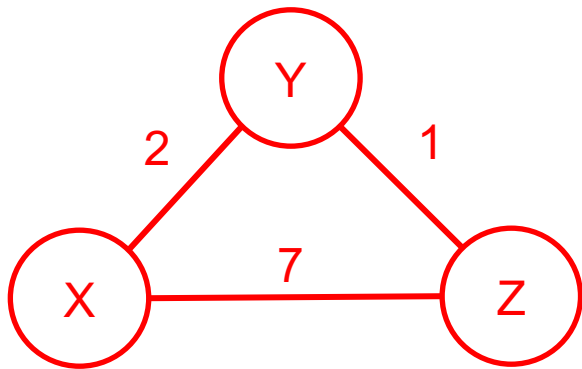


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

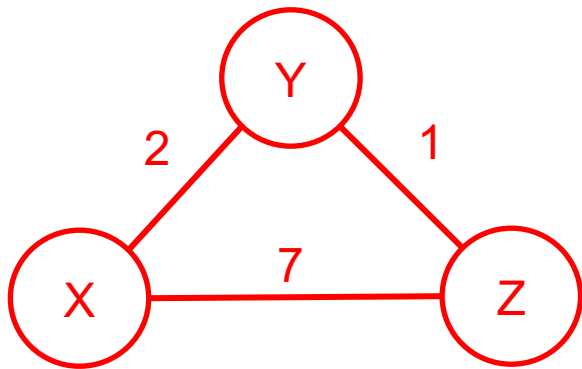


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

Realizar os anúncios das rotas e recalcular os vetores em um novo instante de tempo t_1 .

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

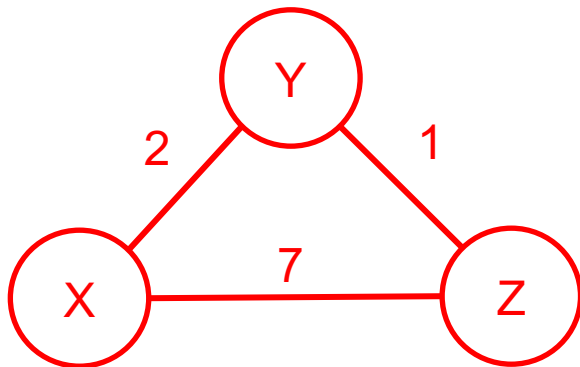


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_1

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_1

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

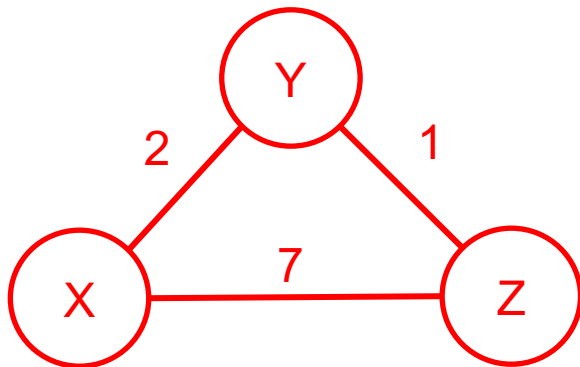


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_1

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_1

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

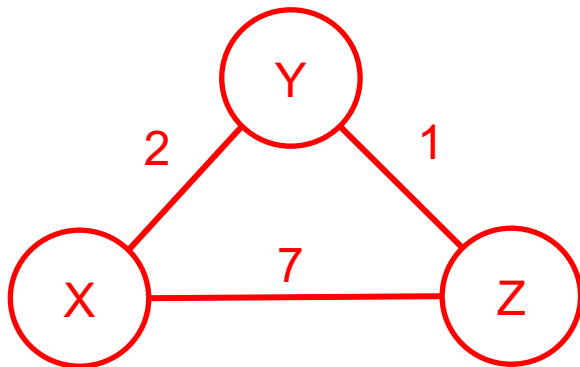


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_1

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_1

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

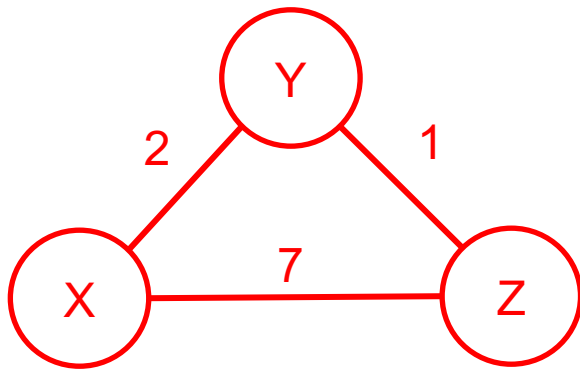


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y			
Z	∞	∞	∞

t_1

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_1

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

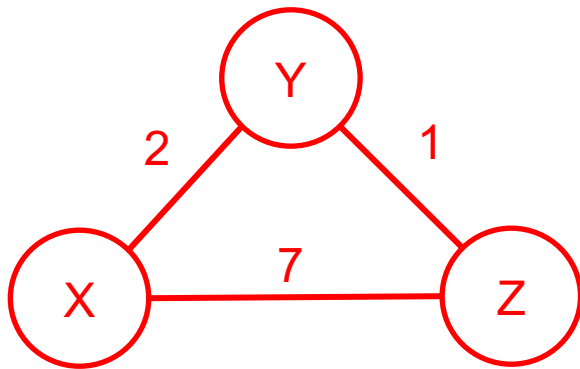


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	∞	∞	∞

t_1

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_1

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

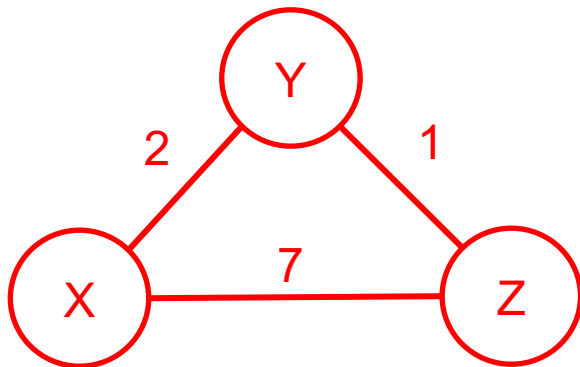


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	∞	∞	∞

t_1

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_1

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

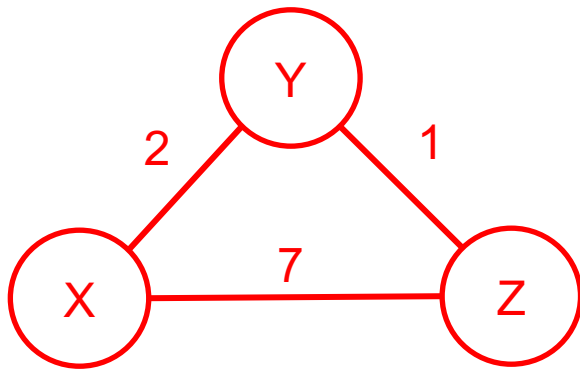


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	∞	∞	∞

t_1

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_1

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

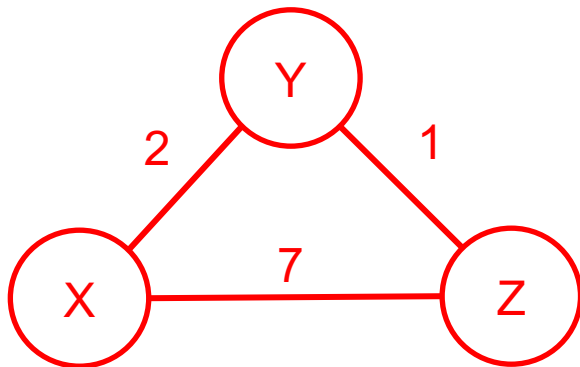


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z			

t_1

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_1

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

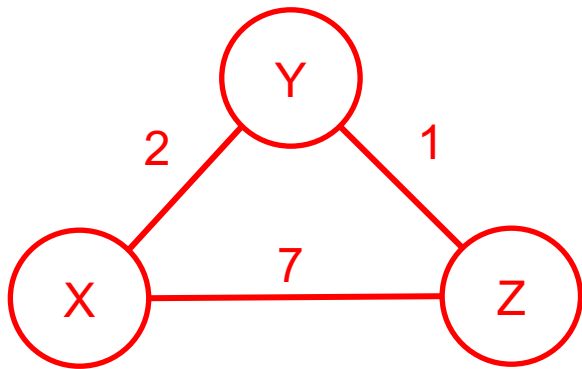


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_1

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

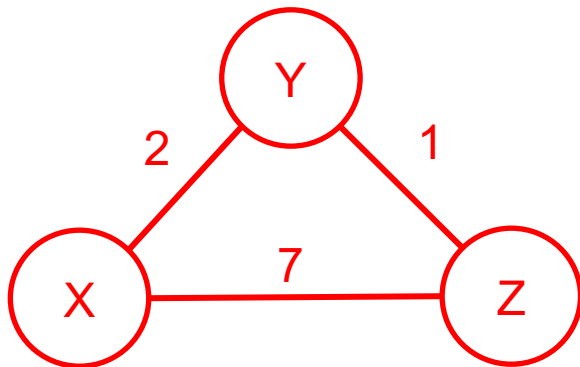


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

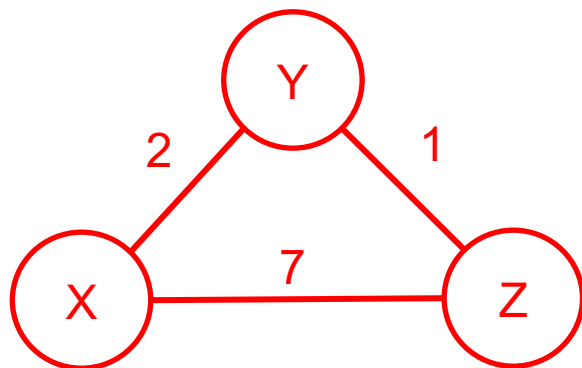


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

Novas rotas

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

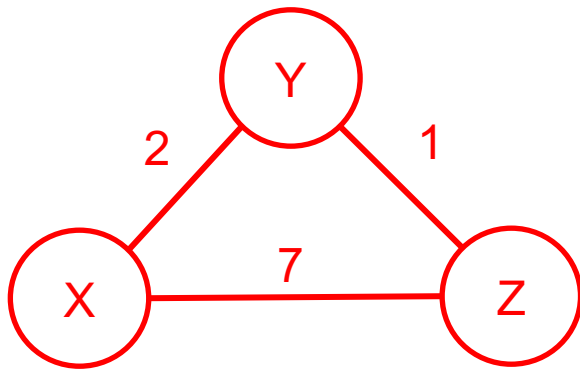


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

Novas rotas

Recalcular X

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

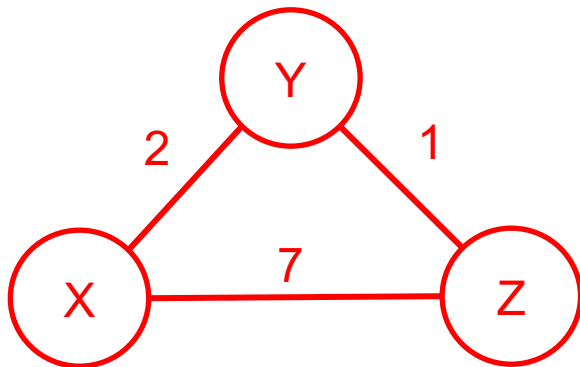


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

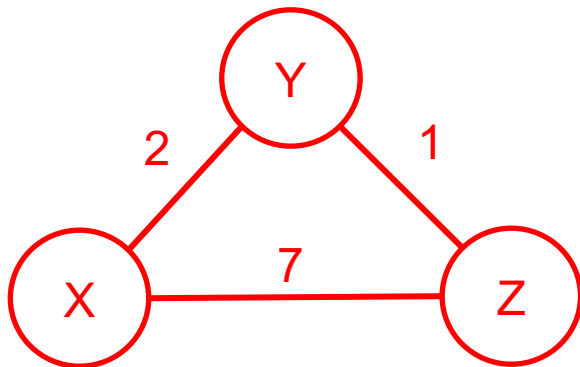


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

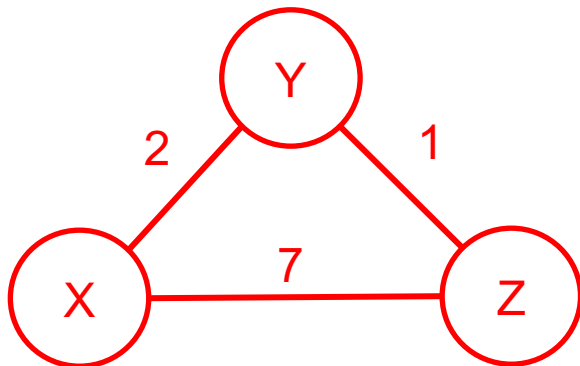


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

$$d_z(y) = \min\{c(z, y), +d_y(y)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

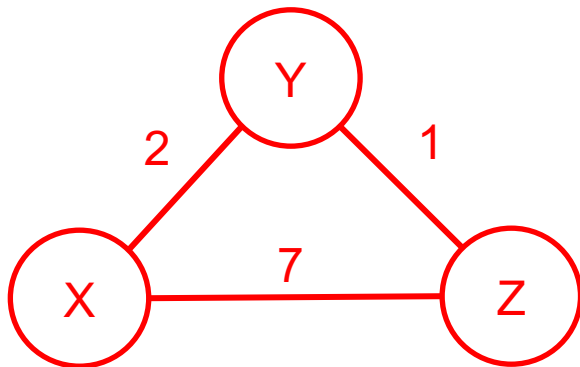


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

$$d_z(y) = \min\{c(z, y) + d_y(y)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

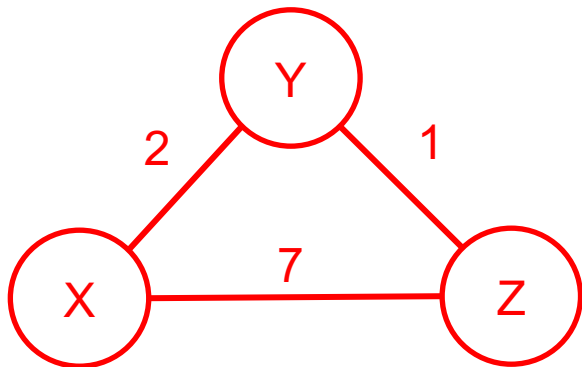


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

$$d_z(y) = \min\{c(z, y) + d_y(y)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

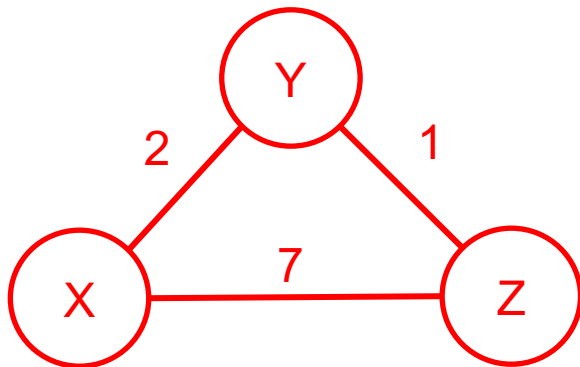


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

$$d_z(y) = \min\{c(z, y) + d_y(y)\}$$

$$d_z(y) = \min\{1 + 0\} = 1$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

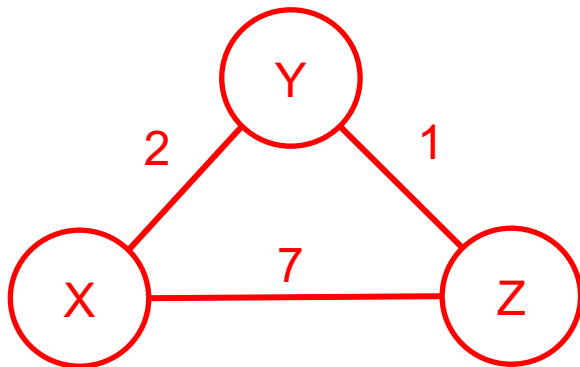


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

$$d_z(y) = \min\{c(z, y) + d_y(y)\}$$

$$d_z(y) = \min\{1 + 0\} = 1$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

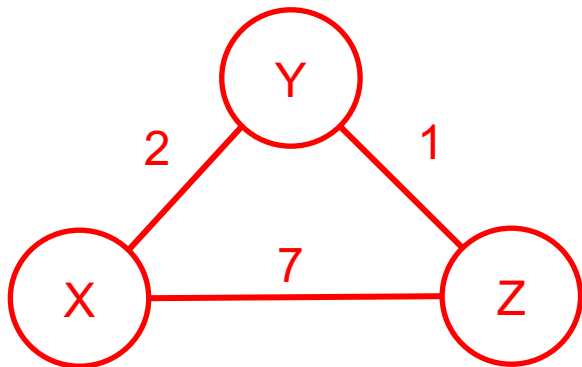


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + 1\}$$

$$d_z(y) = 1$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

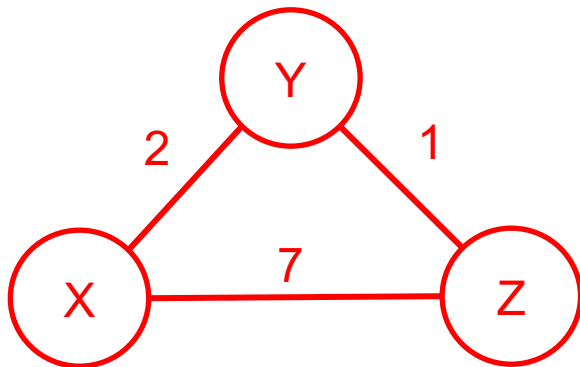


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(y) = \min\{c(x, y) + d_y(y), c(x, z) + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + d_z(y)\}$$

$$d_x(y) = \min\{2 + 0, 7 + 1\}$$

$$d_x(y) = \min\{2, 8\} = 2$$

$$d_z(y) = 1$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

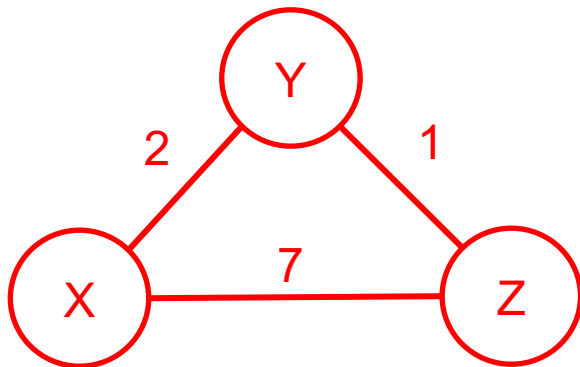


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

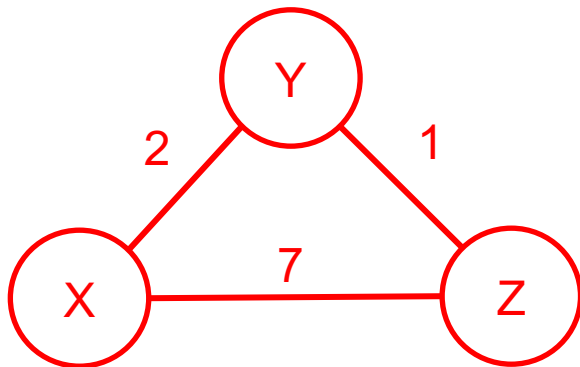


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

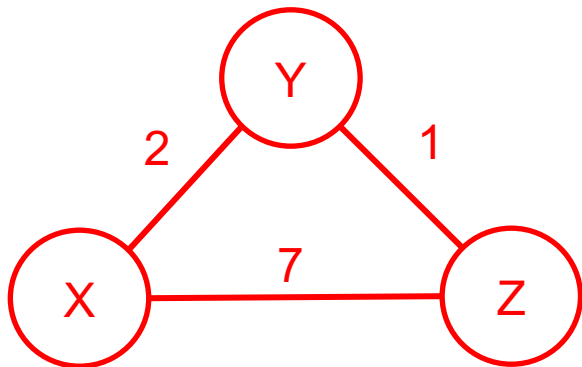


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

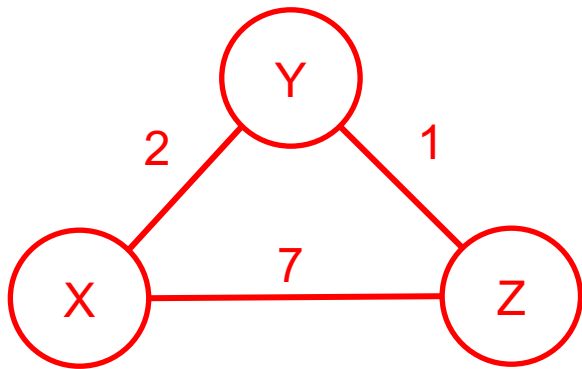


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

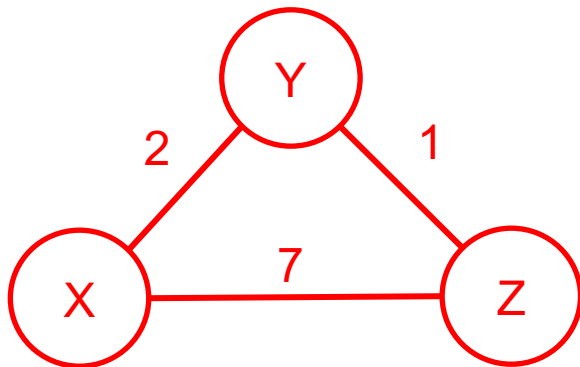


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

$$d_x(z) = \min\{2 + 1, 7 + 0\}$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

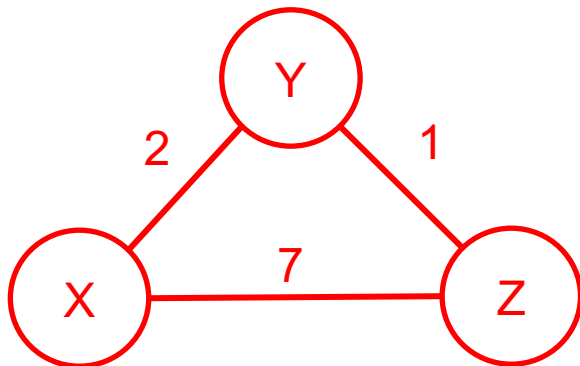


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

$$d_x(z) = \min\{2 + 1, 7 + 0\}$$

$$d_x(z) = \min\{3, 7\} = 3$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

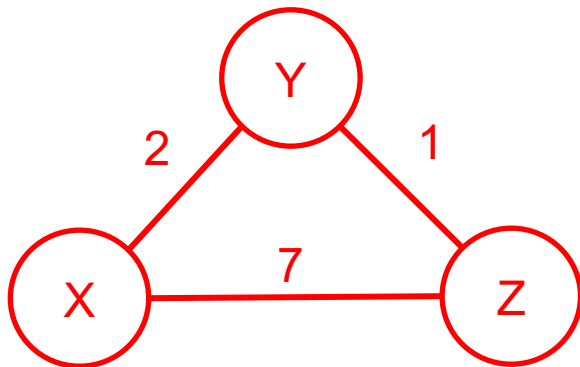


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	7	1	0

t_1

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

$$d_x(z) = \min\{2 + 1, 7 + 0\}$$

$$d_x(z) = \min\{3, 7\} = 3$$

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.

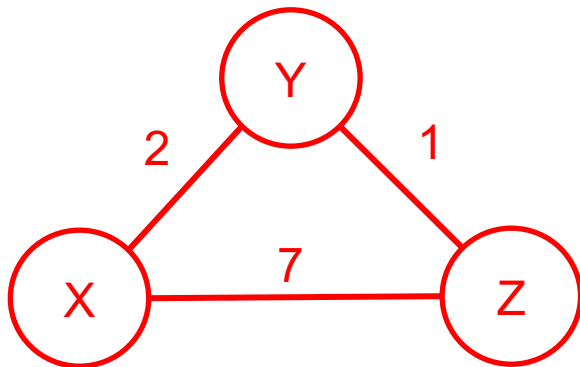


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	7	1	0

t_1

$$d_x(z) = \min\{c(x, y) + d_y(z), c(x, z) + d_z(z)\}$$

$$d_x(z) = \min\{2 + d_y(z), 7 + 0\}$$

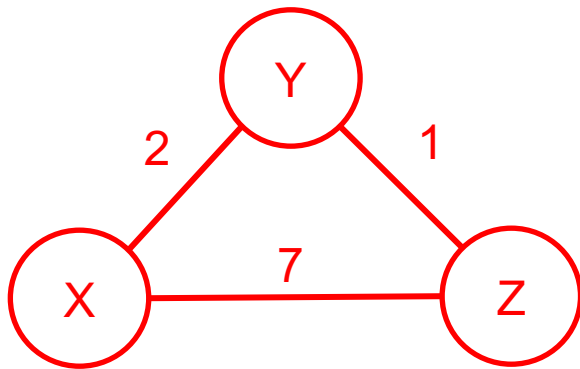
$$d_x(z) = \min\{2 + 1, 7 + 0\}$$

$$d_x(z) = \min\{3, 7\} = 3$$

*Cada tabela atualiza apenas seu vetor.

Tabelas de Roteamento

- Após o cálculo do *Distance Vector*, o vetor gerado precisa ser armazenado e para isso cada nó possui uma tabela com as rotas armazenadas;
- As rotas são atualizadas constantemente a cada mudança na rede.



*Cada tabela atualiza apenas seu vetor.

Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	7	1	0

t_1

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

Tabela do nó Z

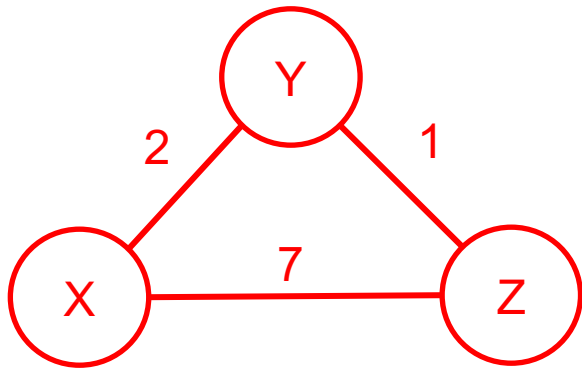
	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	3	1	0

t_1

Tabelas de Roteamento



*Cada tabela atualiza apenas seu vetor.

Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	7	1	0

t_2

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_2

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

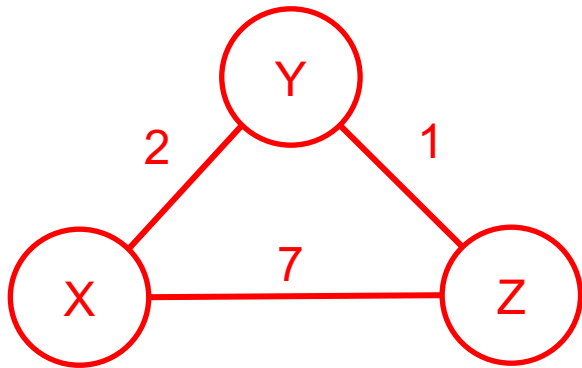
	X	Y	Z
X	0	2	7
Y	2	0	1
Z	3	1	0

t_1

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	3	1	0

t_2

Tabelas de Roteamento



*Cada tabela atualiza apenas seu vetor.

Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	3	1	0

t_2

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_2

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

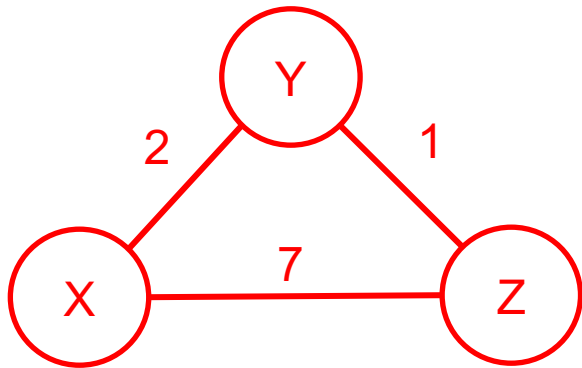
	X	Y	Z
X	0	2	7
Y	2	0	1
Z	3	1	0

t_1

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	3	1	0

t_2

Tabelas de Roteamento



*Cada tabela atualiza apenas seu vetor.

Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	3	1	0

t_2

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	3	1	0

t_2

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	3	1	0

t_1

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	3	1	0

t_2

Tabelas de Roteamento

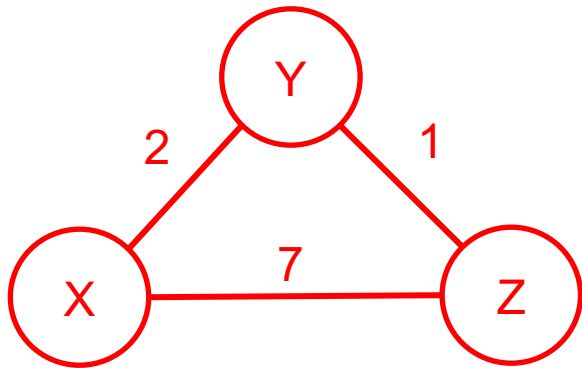


Tabela do nó X

	X	Y	Z
X	0	2	7
Y	∞	∞	∞
Z	∞	∞	∞

t_0

Tabela do nó Y

	X	Y	Z
X	∞	∞	∞
Y	2	0	1
Z	∞	∞	∞

t_0

Tabela do nó Z

	X	Y	Z
X	∞	∞	∞
Y	∞	∞	∞
Z	7	1	0

t_0

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	7	1	0

t_1

	X	Y	Z
X	0	2	7
Y	2	0	1
Z	3	1	0

t_1

Tempo de convergência

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	3	1	0

t_2

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	3	1	0

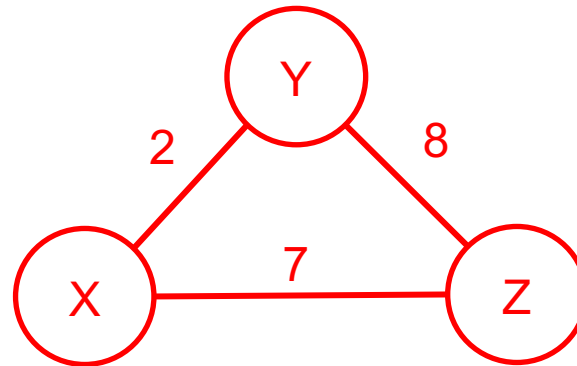
t_2

	X	Y	Z
X	0	2	3
Y	2	0	1
Z	3	1	0

t_2

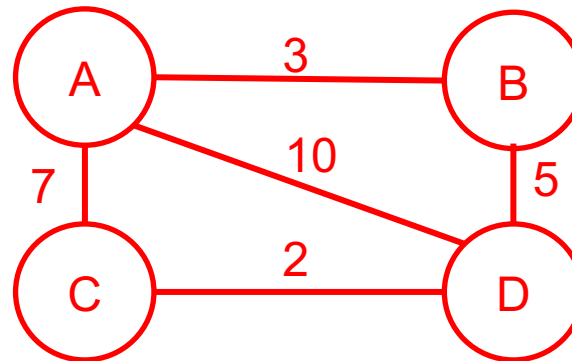
Exercícios propostos:

1. Baseado nos exemplos anteriores (conforme grafo abaixo), considere que no momento t_2 houve atualização do custo do segmento \overline{YZ} , em que $\overline{YZ}=8$. Determine para o momento t_3 as atualizações nas tabelas de cada nó.



2. Considere o seguinte grafo abaixo:

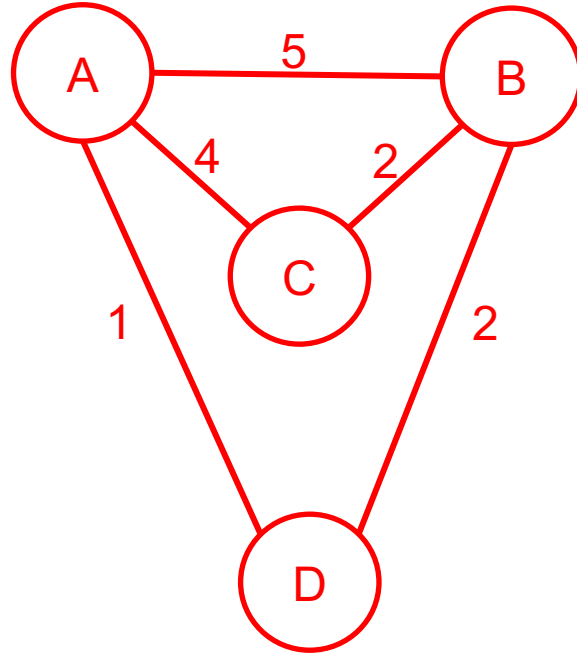
- Construa as tabelas de cada nó em cada instante de tempo necessário até a convergência;
- Atualize as tabelas para o segmento $\overline{CD}=4$;
- Atualize as tabelas para a exclusão do segmento \overline{AD} .



Exercícios propostos:

3. Considere o seguinte grafo abaixo:

- a) Construa as tabelas de cada nó em cada instante de tempo necessário até a convergência.



Bibliografia

■ BÁSICA:

- BRITO, S. H. B. **IPv6: o novo protocolo da internet**. São Paulo: Novatec, 2013.
- **COMER, D. Interligação de redes com TCP/IP: princípios, protocolos e arquitetura**. Rio de Janeiro: Elsevier; Campus, 2006. v.1.
- SOUSA, L. B. **Projetos e implementação de redes: Fundamentos, soluções, arquiteturas e planejamento**. 2. ed. São Paulo: Érica, 2011.

■ COMPLEMENTAR:

- BIRKNER, MATTHEW H. (ED.). **Projeto de interconexão de redes: CISCO Internetwork Design - CID**. São Paulo: Pearson Education, 2003.
- BRITO, S. H. B. **Laboratórios de tecnologias cisco em infraestrutura de redes**. 2.ed. São paulo: Novatec, 2014.
- FREITAS, A. E. S.; BEZERRA, R. M. S. **IPv6: conceitos e aspectos práticos**. Rio Janeiro: Ciência Moderna, 2015.
- LIMA, João Paulo de. **Administração de redes Linux: passo a passo**. Goiânia: Terra, 2003.
- STARLIN, G. **Redes de computadores: comunicação de dados TCP/IP: conceitos, protocolos e uso**. Rio de Janeiro: Alta Books, 2004.
- VASCONCELOS, L.; VASCONCELOS, M. **Manual prático de redes**. Rio de Janeiro: Laércio Vasconcelos Computação, 2008.