

# REDES PARA INTERNET

## Algoritmos de Roteamento – Dijkstra e Flooding

### 4.2

Universidade Estácio de Sá  
Prof. Welsing Moreira Pereira  
<http://www.professorwelsing.webnode.com>

O que sabemos é uma gota, o que ignoramos é um oceano.  
Isaac Newton (1642-1727) – Físico e Matemático Inglês.



# Estácio

1

## Algoritmos de Roteamento



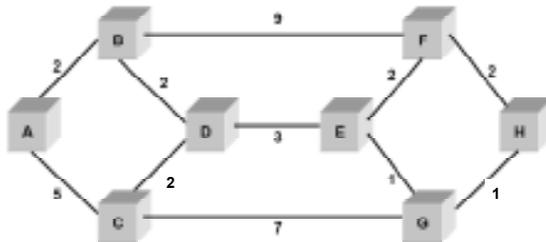
- Link-state
  - Topologia da rede e custos dos enlaces são conhecidos.
  - Um nó conhece as identidades e custos dos nós vizinhos diretamente ligados a ele.
  - Esse conhecimento é divulgado via broadcast a todos os seus vizinhos.
  - Vizinhos divulgam esses dados a todos seus vizinhos de maneira recursiva.
  - Até que todos tenham uma massa de informação na qual permite obter o conhecimento da topologia da rede.

2

## Algoritmos de Roteamento



- Algoritmo de Dijkstra
  - Baseado em métricas
    - Valor unitário (salto ou hop)
    - Distância
    - Capacidade na linha de transmissão
    - Custo da linha de comunicação



3

## Algoritmos de Roteamento



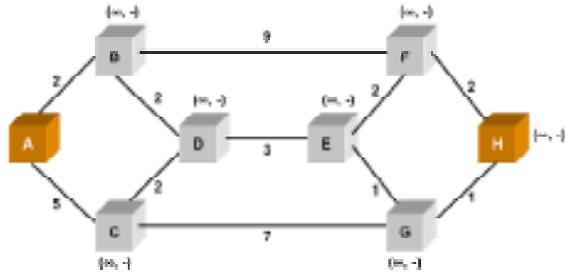
- Algoritmo de Dijkstra
  - Cada nó tem um rótulo que pode ser:
    - Tentativa
    - Permanente
  - o rótulo mantém a menor distância ao nó origem usando o melhor caminho conhecido (inicialmente é infinito)
  - Quando descobre-se que a distância ao nó origem é a menor dentre as tentativas, faz-se o rótulo permanente.

4

## Algoritmos de Roteamento

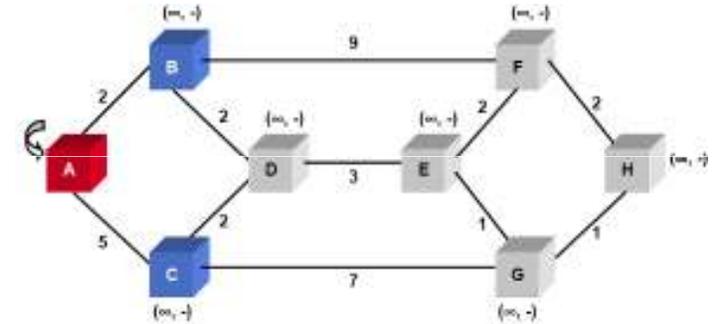


- Algoritmo de Dijkstra



5

## Algoritmo de Dijkstra



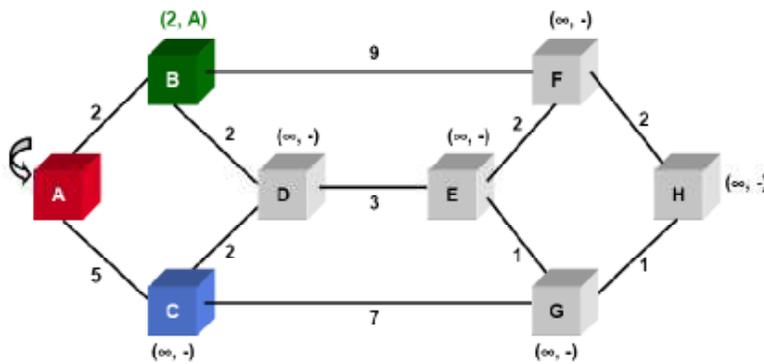
Vizinhos: B, C

Tentativas:

Permanentes: A

6

## Algoritmo de Dijkstra



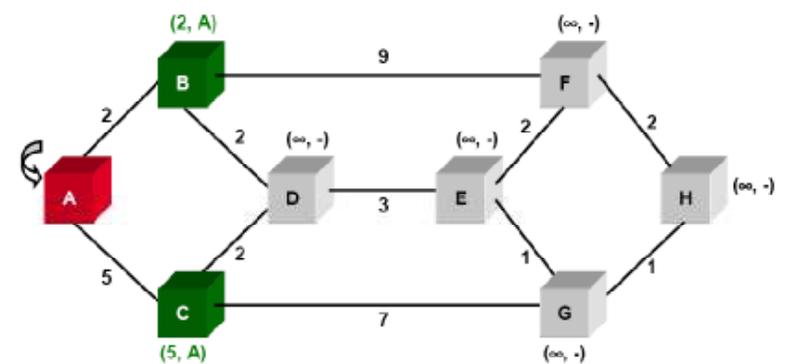
Vizinhos: C

Tentativas: B

Permanentes: A

7

## Algoritmo de Dijkstra



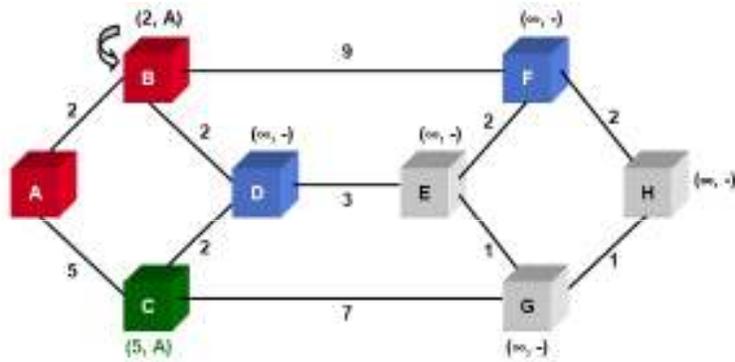
Vizinhos:

Tentativas: B, C

Permanentes: A

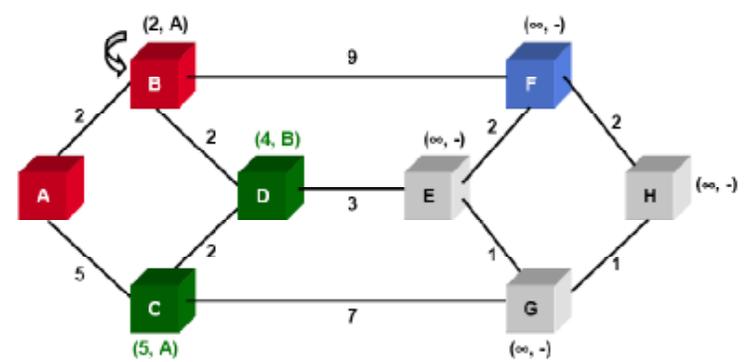
8

# Algoritmo de Dijkstra



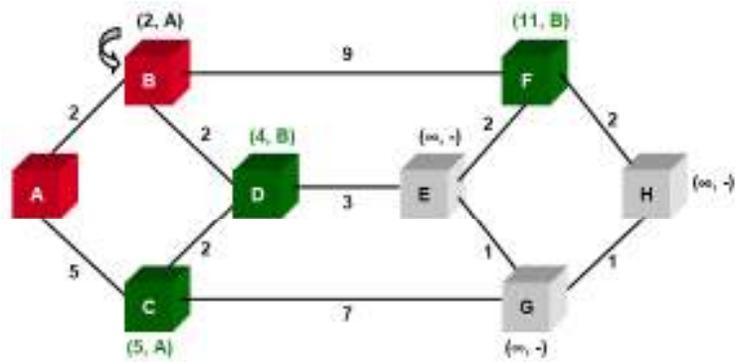
Vizinhos: D, F  
 Tentativas: C  
 Permanentes: A, B

# Algoritmo de Dijkstra



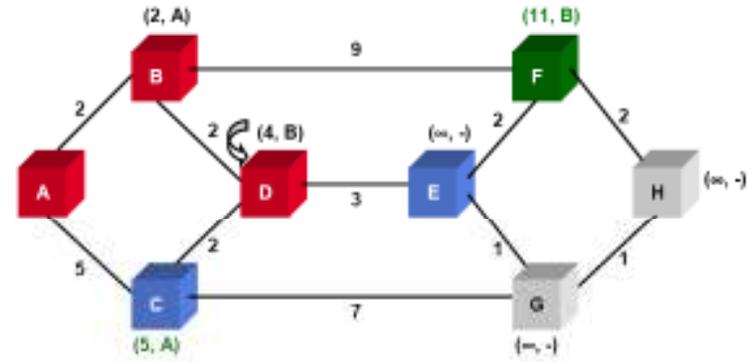
Vizinhos: F  
 Tentativas: C, D  
 Permanentes: A, B

# Algoritmo de Dijkstra



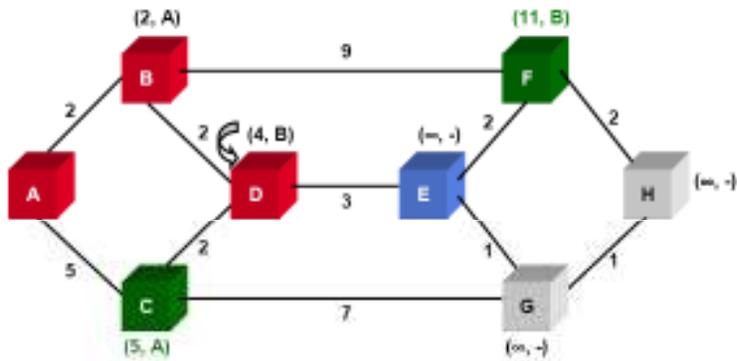
Vizinhos:  
 Tentativas: C, D, F  
 Permanentes: A, B

# Algoritmo de Dijkstra



Vizinhos: C, E  
 Tentativas: C, F  
 Permanentes: A, B, D

## Algoritmo de Dijkstra



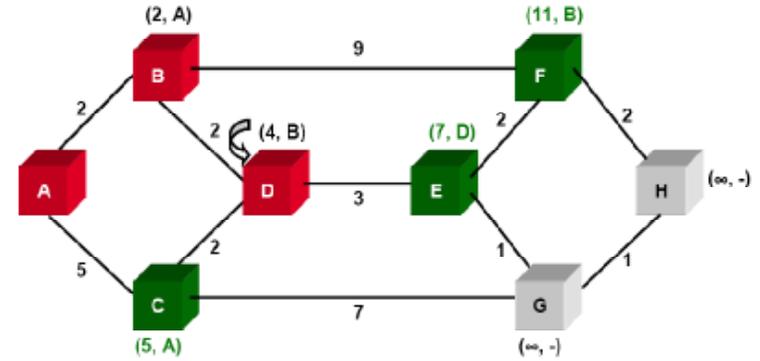
Vizinhos: E

Tentativas: C, F, E

Permanentes: A, B, D

13

## Algoritmo de Dijkstra



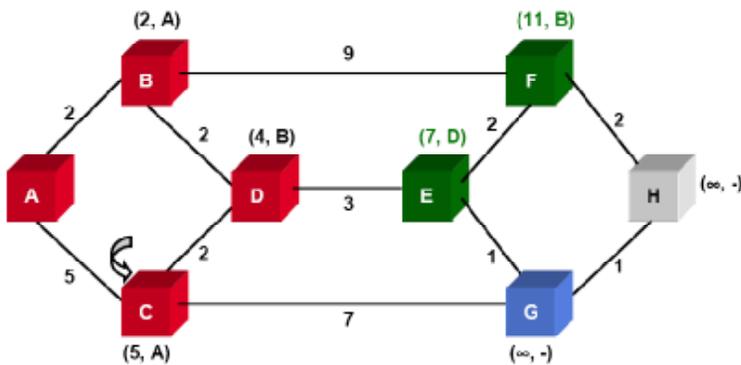
Vizinhos:

Tentativas: C, F, E

Permanentes: A, B, D

14

## Algoritmo de Dijkstra



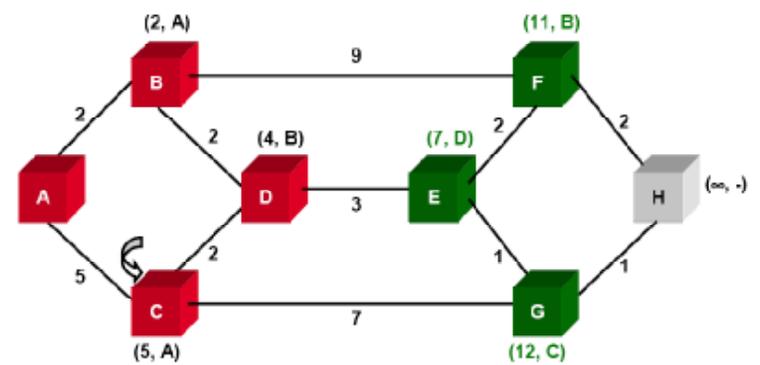
Vizinhos: G

Tentativas: F, E

Permanentes: A, B, D, C

15

## Algoritmo de Dijkstra



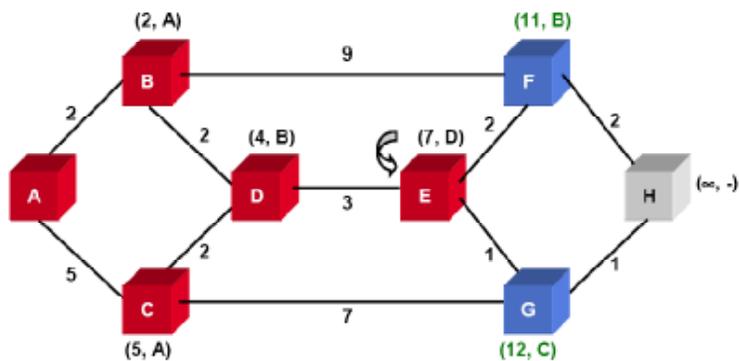
Vizinhos:

Tentativas: F, E, G

Permanentes: A, B, D, C

16

## Algoritmo de Dijkstra



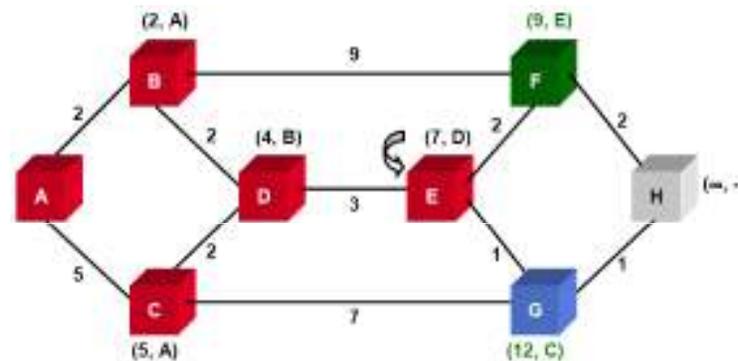
Vizinhos: F, G

Tentativas: F, G

Permanentes: A, B, D, C, E

17

## Algoritmo de Dijkstra



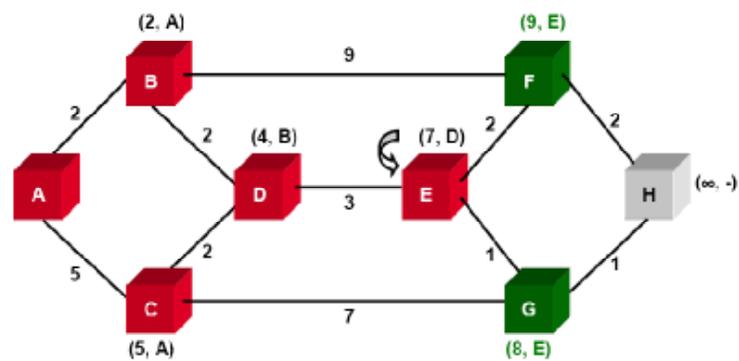
Vizinhos: G

Tentativas: F, G

Permanentes: A, B, D, C, E

18

## Algoritmo de Dijkstra



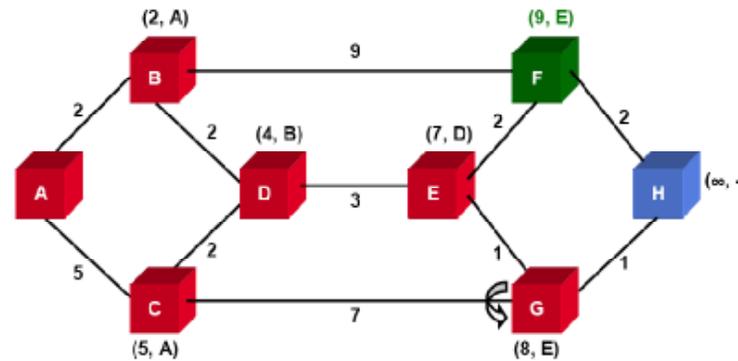
Vizinhos:

Tentativas: F, G

Permanentes: A, B, D, C, E

19

## Algoritmo de Dijkstra



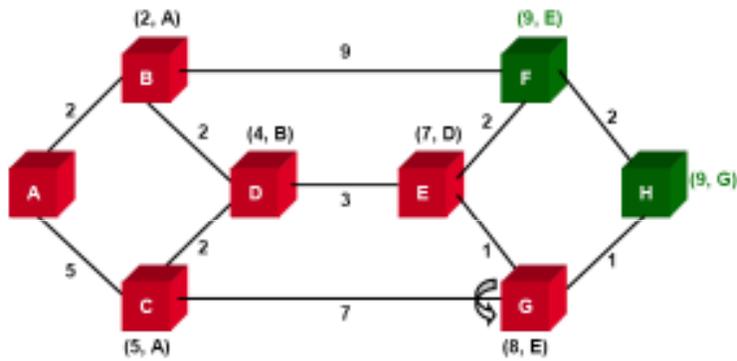
Vizinhos: H

Tentativas: F

Permanentes: A, B, D, C, E, G

20

## Algoritmo de Dijkstra



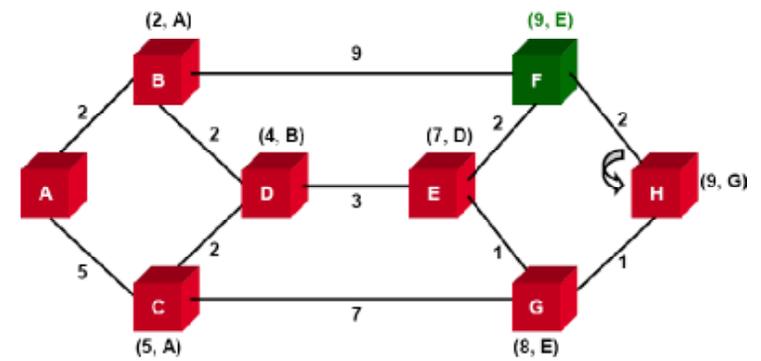
Vizinhos:

Tentativas: F, H

Permanentes: A, B, D, C, E, G

21

## Algoritmo de Dijkstra



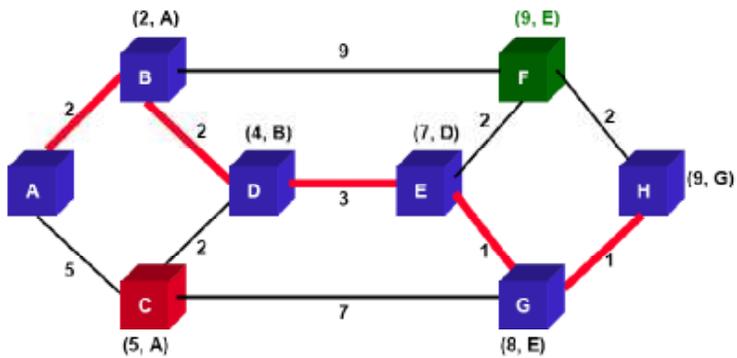
Vizinhos:

Tentativas: F

Permanentes: A, B, D, C, E, G, H

22

## Algoritmo de Dijkstra



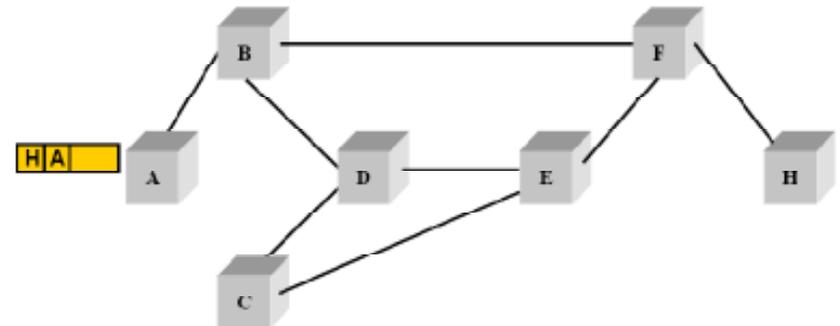
Vizinhos:

Tentativas: F

Permanentes: A, B, D, C, E, G, H

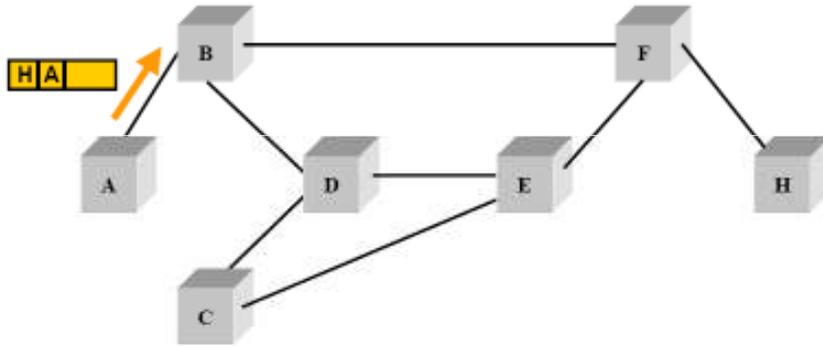
23

## Mecanismo de Flooding



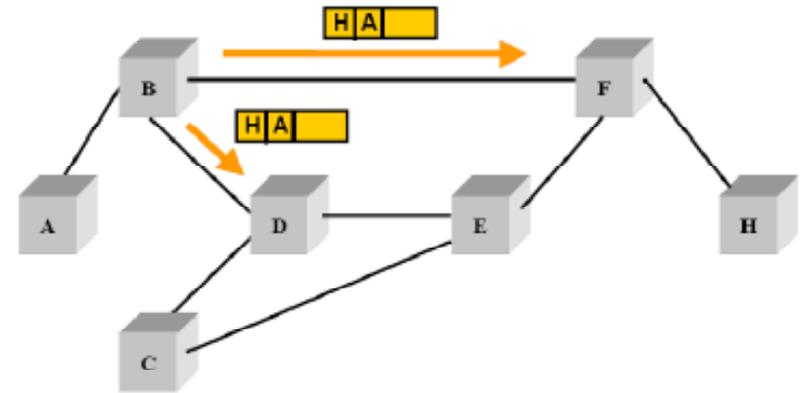
24

## Mecanismo de Flooding



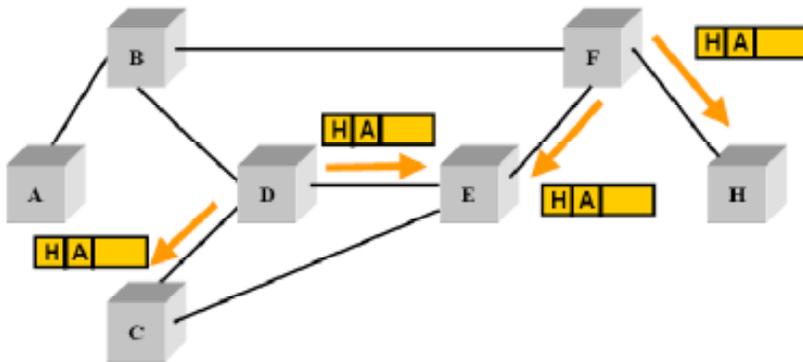
25

## Mecanismo de Flooding



26

## Mecanismo de Flooding



27

## Mecanismo de Flooding



- Crescimento exponencial do número de pacotes na rede
- Exige a utilização de algum mecanismo de descarte dos pacotes para evitar que pacotes fiquem eternamente circulando na rede
  - Contador de saltos no cabeçalho de cada pacote
  - Nó guarda a informação dos pacotes que já foram repassados por ele
- Pouco prático, porém útil em algumas aplicações
  - Atualização de bancos de dados distribuídos
  - Métrica de avaliação para outros algoritmos de roteamento (sempre utiliza o caminho mais curto, pois usa todos os caminhos em paralelo)

28

## Exercícios:



- Qual o objetivo do algoritmo de Dijkstra?
- Como funciona o mecanismo de flooding e que cuidados devemos tomar ao utilizar tal mecanismo?