



# Redes de Computadores II

## Aula 23

# Roteamento na Internet

**Prof. Elias P. Duarte Jr.**

Universidade Federal do Paraná (UFPR)

Departamento de Informática

[www.inf.ufpr.br/elias/redes](http://www.inf.ufpr.br/elias/redes)

# Sumário da Aula de Hoje

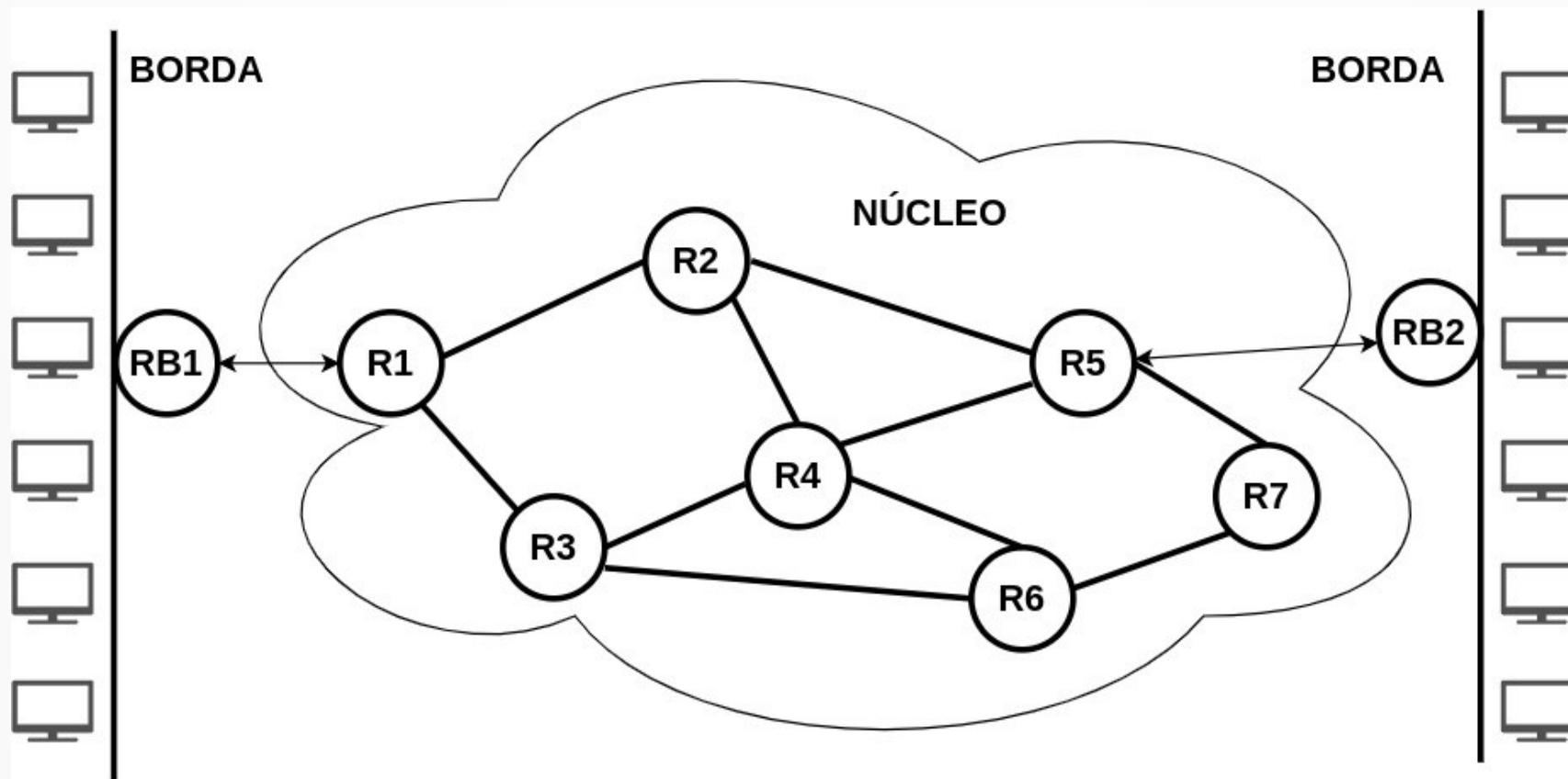
- Nestas aulas vamos estudar como é feito o roteamento na Internet
- Arquitetura de Roteamento da Internet: Sistemas Autônomos
- Algoritmos de Roteamento
  - Bellman-Ford, Dijkstra

# O Mistério do Roteamento na Internet

- O protocolo IP é o protocolo da camada 3 (rede) da Internet
- A principal função da camada de rede é o roteamento
- O protocolo IP realmente faz o roteamento dos pacotes de qualquer origem para qualquer destino através da Internet
- Vamos relembrar

# A Internet: Borda & Núcleo

- Infraestrutura da Internet: Núcleo da Rede (Network Core)
- Borda (*Edge*) tem as redes que se conectam à Internet



# O Mistério do Roteamento

- Para cada pacote recebido, o protocolo IP consulta a tabela de roteamento para decidir a rota
- A tabela de roteamento tem 2 colunas: net-id de destino e próximo roteador vizinho para quem o pacote deve ser enviado
- Lembrando que a última entrada indica o roteador default (relembrando nos prox. slides)
- Já que o IP faz o roteamento dos pacotes: tudo dominado? Já sabemos tudo do roteamento?

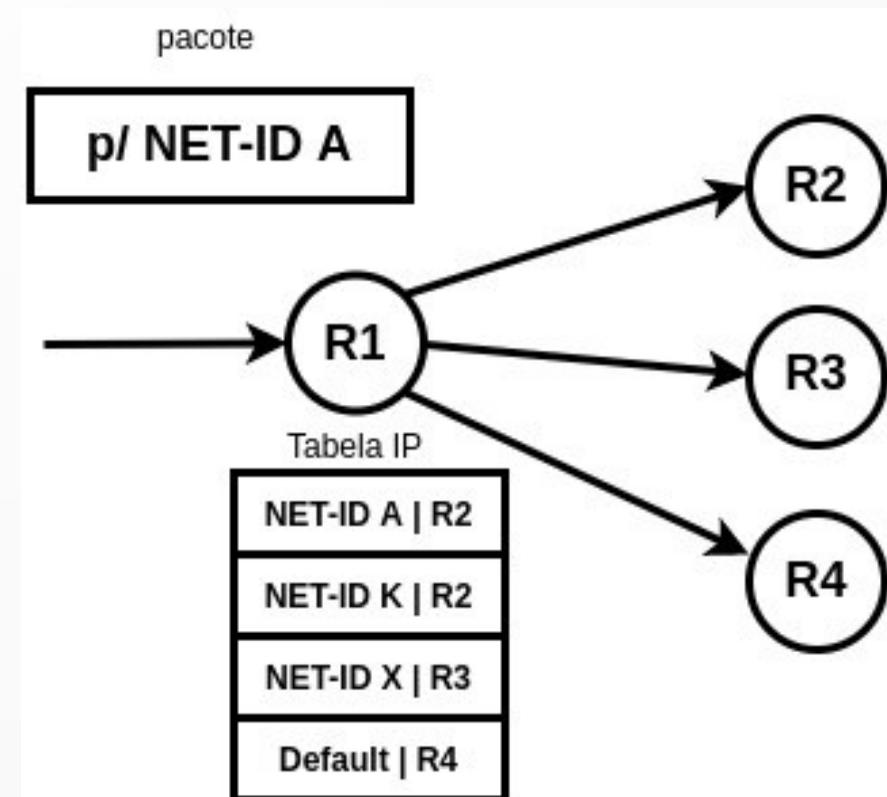
# Como é feito o roteamento IP?

- Quando chega um pacote para ser encaminhado
- O roteador IP extrai o net-id, e faz uma busca na **TABELA DE ROTEAMENTO** ou **TABELA IP**

<i>Net-id Destino</i>	<i>Prox-Passo</i>
net-id-A	R1
net-id-X	R3
...	...
Nenhum acima	<i>Rdefault</i>

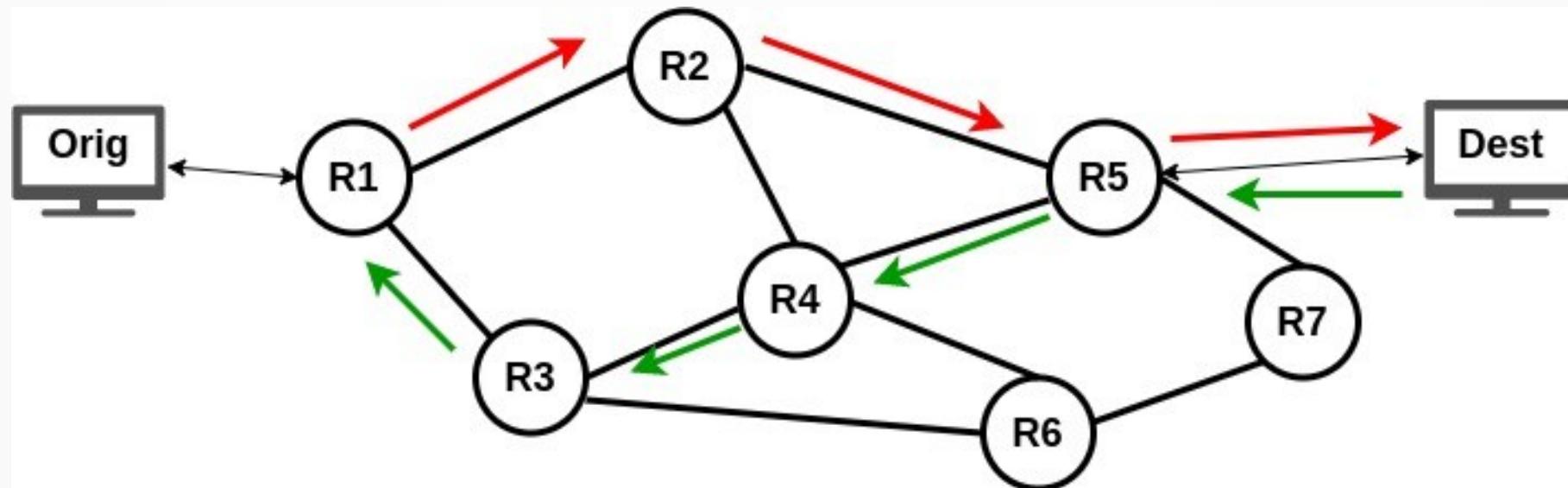
# Tabela IP: Duas Colunas

- Na primeira coluna: net-id do destino
- Na segunda: coluna: próximo passo
- Cada roteador define apenas 1 passo da rota!
- E passa o pacote para o seguinte:



# Tomada de Decisão: Roteador

- Cada roteador decide baseado na sua tabela local
- Decisão independente dos demais
- Situações que podem acontecer: o caminho de A para B pode ser diferente do caminho de B para A



# Tomada de Decisão: Roteador

- Cada roteador decide baseado na sua tabela local
- Decisão independente dos demais
- Situações que podem acontecer: o caminho de A para B pode ser diferente do caminho de B para A
- Além disso: vimos inclusive que podem acontecer loops

# O Mistério do Roteamento

- Para cada pacote recebido, o protocolo IP consulta a tabela de roteamento para decidir a rota
- A tabela de roteamento tem 2 colunas: net-id de destino e próximo roteador vizinho para quem o pacote deve ser enviado (figura prox slide)
- Lembrando que a última entrada indica o roteador default
- Já que o IP faz o roteamento dos pacotes: tudo dominado? Já sabemos tudo do roteamento? **NÃO!**

# O IP Vs. Protocolos de Roteamento

- Desta forma: o IP usa a tabela de roteamento para encaminhar os pacotes
- Mas quem preenche esta tabela??

# Nos Primórdios da Internet

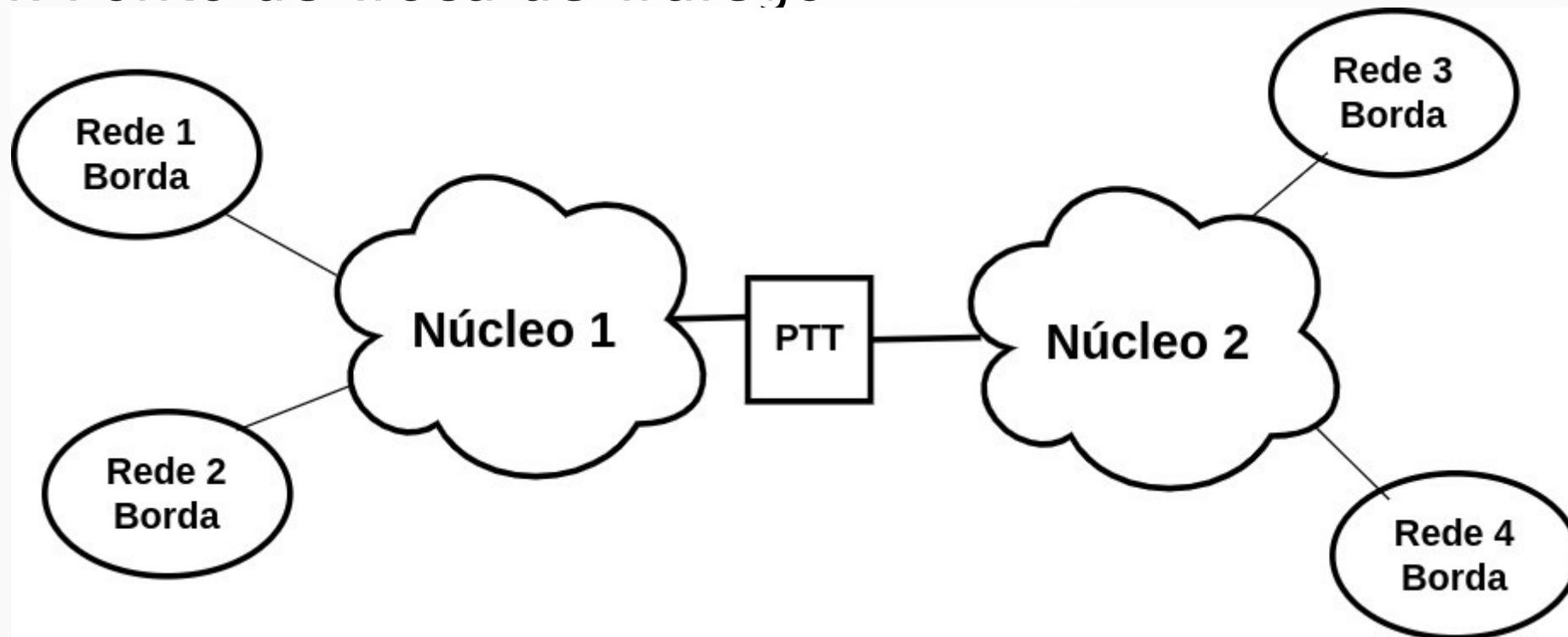
- A tabela de roteamento era preenchida manualmente :-)
- Quando a rede começou a crescer: inviável!
- Neste momento houve uma classificação dos roteadores:
  - Roteadores da Borda: redes que se conectam à Internet
  - Roteadores do Núcleo: redes que foram a infraestrutura da Internet

# Roteadores: Núcleo & Borda

- Roteadores de núcleo: mantêm informações completas, sobre *todos* os destinos
- Roteadores da borda: mantêm apenas informações de como chegar ao núcleo

# De Um para Vários Núcleos

- Em um momento chave da história da Internet um novo núcleo de grande porte conectou (NSFNET) ao que já existia (ARPANET)
- PTT: Ponto de Troca de Tráfego



# Múltiplos Núcleos

- Quando há mais de um núcleo, os roteadores do núcleo não precisam de informações completas
- Basta terem informações sobre seus próprios destinos (aí sim completas) e sobre como chegar aos núcleos dos demais destinos

# Arquitetura de Roteamento

- Mais uma vez vamos redefinir a Internet!
- A Internet é uma rede de redes independentes, cada uma com administração autônoma
- Sistema Autônomo, ou em inglês *Autonomous System* (AS)
- Cada sistema autônomo tem, inclusive, liberdade para escolher sua estratégia de roteamento interna
- Deve propagar para outros roteadores informações sobre seus destinos (não a topologia!)

# Roteamento Interno/Externo

- Os ASes são a base do roteamento na Internet
- Classificação importante:
  - roteamento dentro de um AS: interno
  - roteamento entre ASes: externo
- Os protocolos de roteamento da Internet (aqueles que preenchem as tabelas de roteamento) são:
  - internos: RIP, OSPF, ...
  - externos: BGP
- Roteadores são configurados adequadamente

# Algoritmos & Protocolos

- Para estudar o roteamento na Internet temos que investigar tanto protocolos de roteamento como algoritmos de roteamento
- Qual a diferença de ALGORITMO para PROTOCOLO?

# Algoritmos & Protocolos

- Para estudar o roteamento na Internet temos que investigar tanto protocolos de roteamento como algoritmos de roteamento
- Qual a diferença de ALGORITMO para PROTOCOLO?
- Um protocolo é um conjunto de regras: neste caso usadas por roteadores (e hosts!) para manter informações de roteamento
- Um algoritmo é uma sequência finita de passos que resolve um problema

# Algoritmos de Roteamento

- Problema que resolvem: determinar a “melhor” rota de qualquer origem para qualquer destino
- Melhor: depende da métrica que se escolhe
- Na Internet: número de saltos (*hops*), tamanho da rota
- Outras são possíveis: carga, confiabilidade, ...
- Os algoritmos de roteamento usam um grafo para representar a rede

# 2 Famílias de Algoritmos de Roteamento

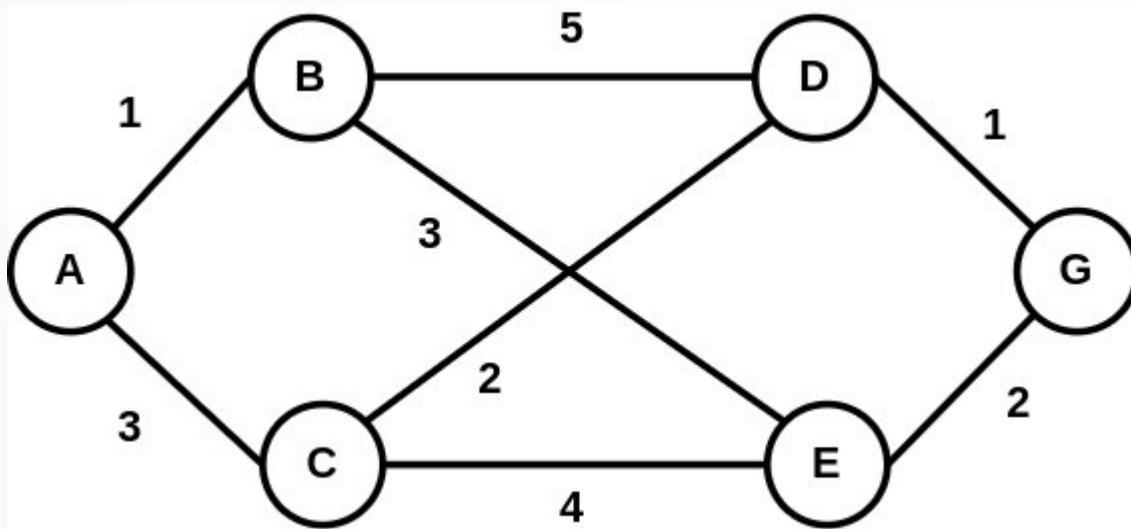
- Apesar de que é possível fazer outras classificações podemos agrupar todos os algoritmos de roteamento em 2 famílias:
  - Bellman-Ford (Vetor de Distâncias)
  - Shortest-Path-First (Estado de Enlace)

# O Algoritmo de Bellman-Ford

- A principal característica do algoritmo de Bellman-Ford é que os nodos não têm a topologia
- Inicialmente cada nodo conhece apenas seus vizinhos
- Uma premissa muito razoável!
- O algoritmo funciona em rodadas
  - periodicamente (por ex. a cada 30s uma rodada é executada)

# O Algoritmo de Bellman-Ford

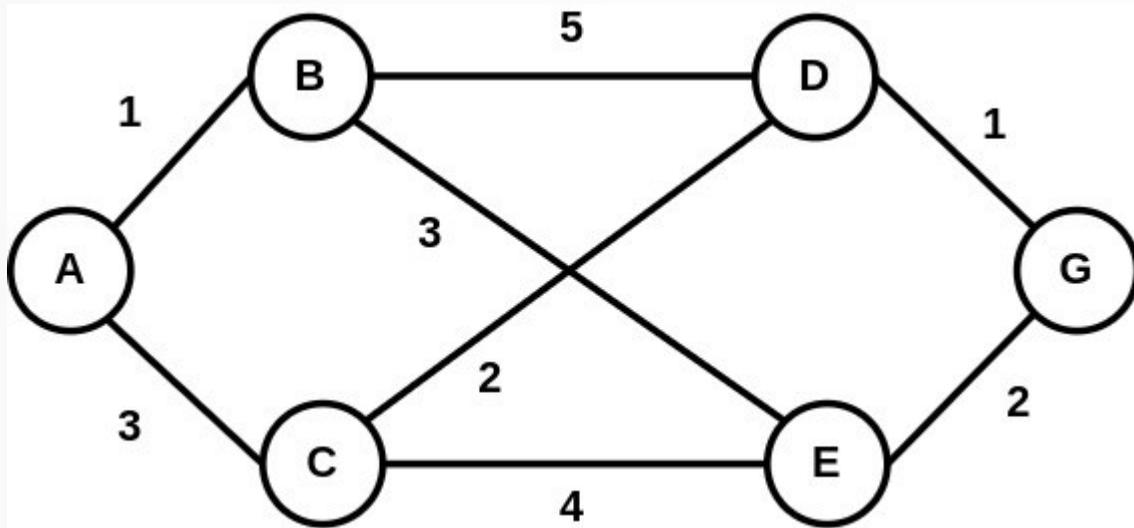
- Considere a seguinte rede exemplo e veja a tabela com que o nodo A inicia:



Destino	Distância
B	1
C	3

# Troca Tabelas por Rodada

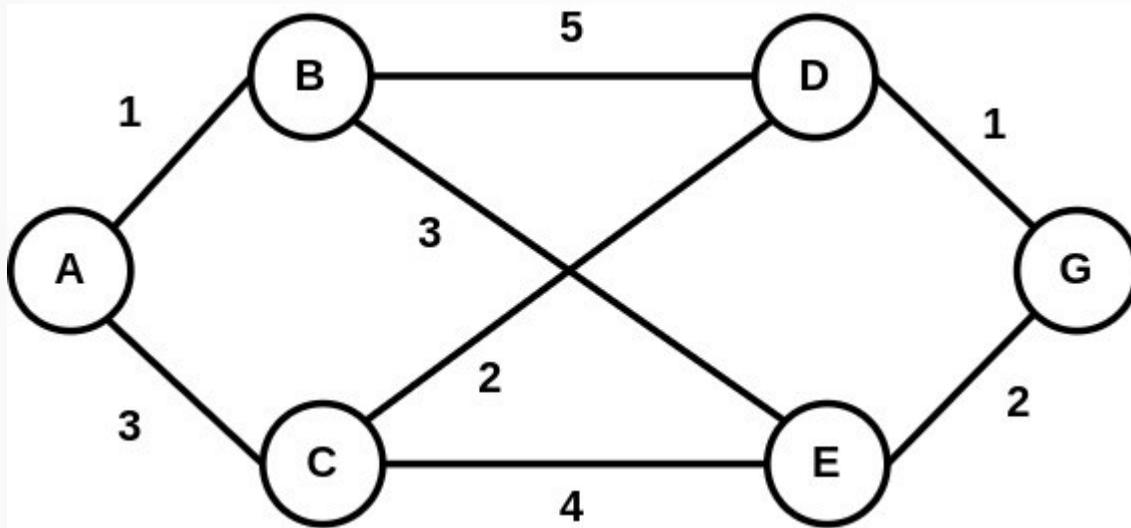
- A cada rodada os vizinhos trocam suas tabelas de roteamento
- Após receber a tabela do B, a tabela do A fica assim:



Destino	Distância	Por onde?
B	1	direto
C	3	direto
D	6	B
E	4	B

# Atualiza Tabela

- Quando descobre um caminho melhor para um destino, atualiza a tabela
- Veja como A atualiza a entrada para D quando recebe a tabela de C



Destino	Distância	Por onde?
B	1	Direto
D	3	Direto
D	5	C
E	4	B

# Passo a Passo: Topologia

- Bellman-Ford
- Inicialmente: conhece vizinhos
- Após 1 rodada: conhece vizinhos dos vizinhos
- Após 2 rodadas: conhece vizinhos dos vizinhos dos vizinhos...
- E assim por diante...

# Problemas Bellman-Ford

- Latência muito alta para definir todas as rotas
- Demora também para refletir alteração da topologia em todas as tabelas
- Se a rede é muito dinâmica: pode nunca estabilizar
  - Por exemplo acontece uma falha e a informação vai sendo refletida nas tabelas...
  - Antes de todas as tabelas refletirem: recuração...
- Além disso: caro mandar tabelas por rodada
  - Possível melhorar com atualizações apenas

# Mas Bellman-Ford é muito usado!

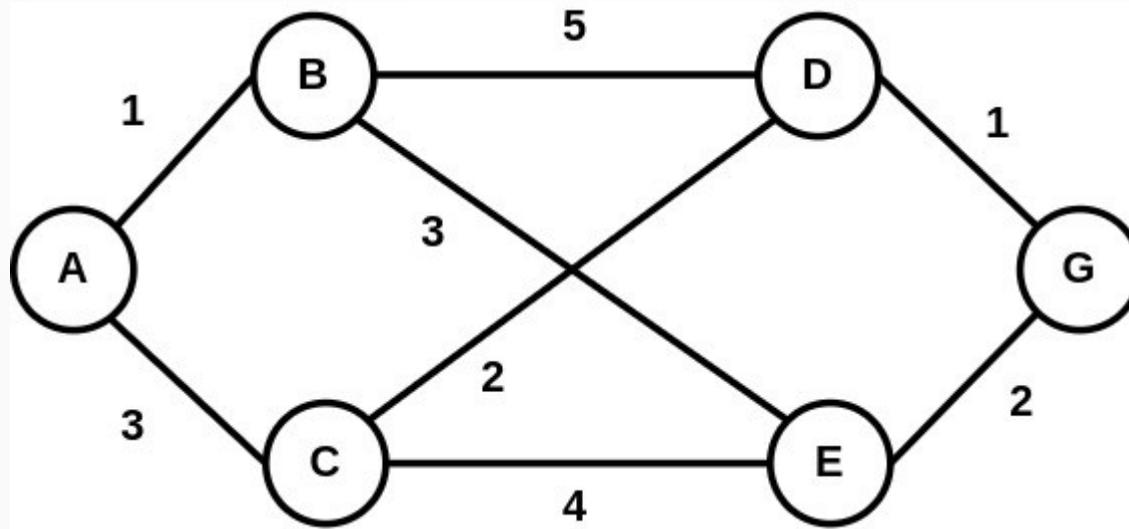
- A grande vantagem é que não precisa da topologia
- No roteamento externo da Internet é a única opção
  - Usado também em roteamento interno (RIP p.ex.)
- Entre diferentes Sistemas Autônomos não se divulga a topologia, apenas destinos alcançáveis e custo
- Mesmo que divulgassem: pense em manter a topologia da Internet inteira!

# O Algoritmo de Dijkstra

- Muito mais eficiente que Bellman-Ford mas...
- Precisa da topologia: nodos já inicializam com a topologia completa da rede
- Nodos apenas mantêm e comunicam informações sobre os estados dos enlaces (Roteamento “Link-State”)
- Durante séculos um problema considerado NP-Completo, até que Dijkstra enxergou!

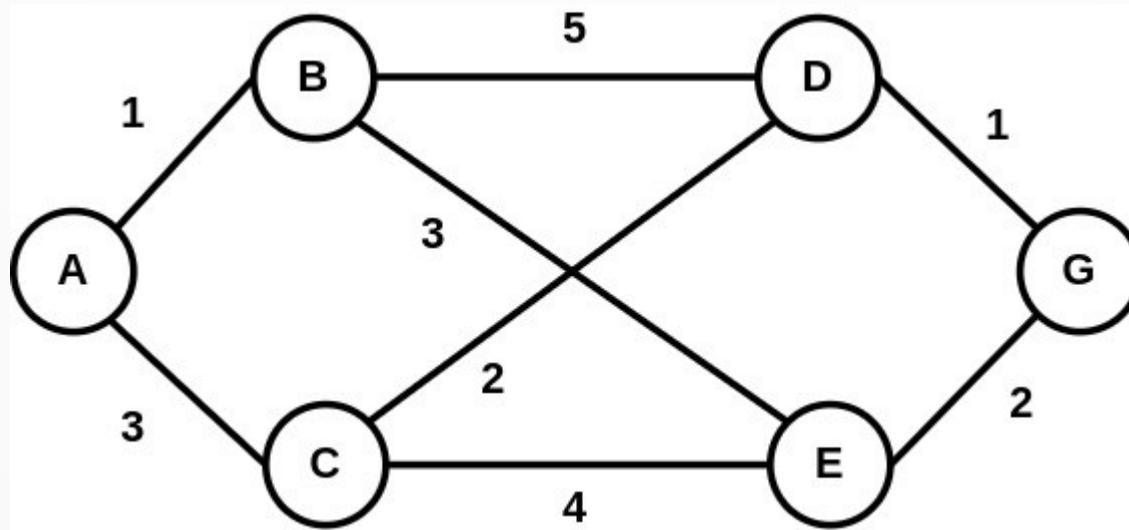
# O que Dijkstra enxergou?

- Que conclusão definitiva é possível tirar sobre a rede abaixo?



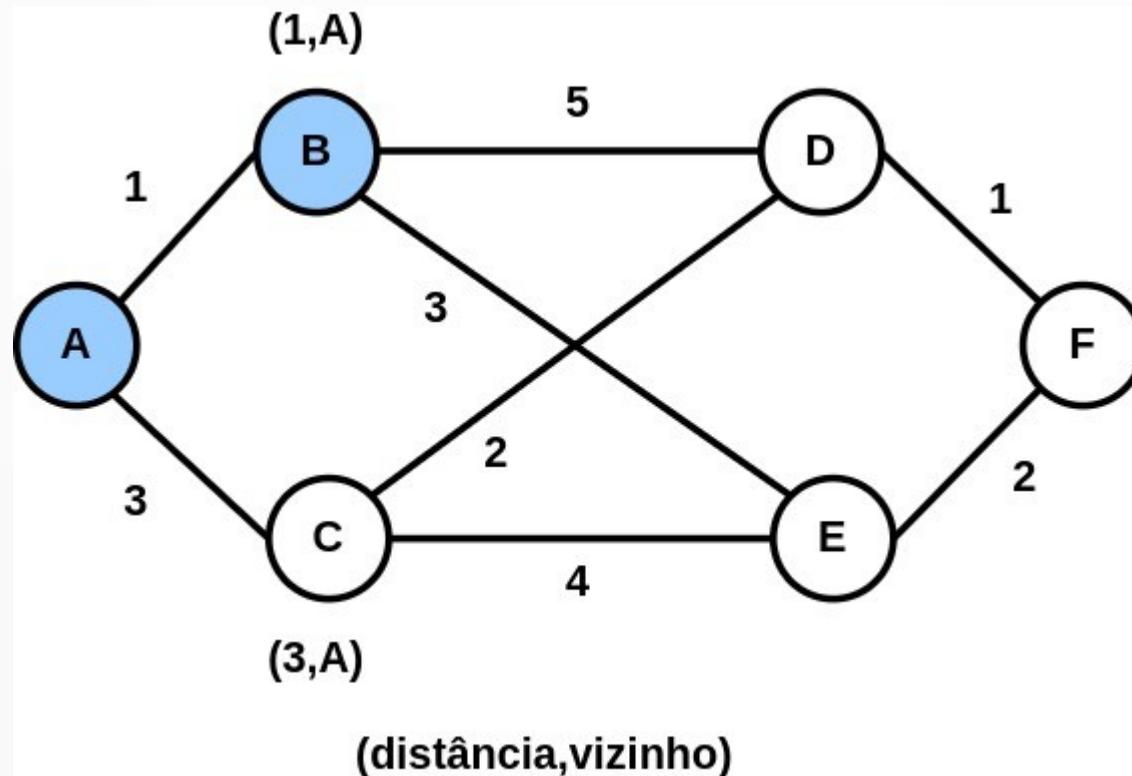
# O que Dijkstra enxergou?

- Que conclusão definitiva é possível tirar sobre a rede abaixo?
- A distância entre A e B é 1, e sempre será, nada mudará isso!



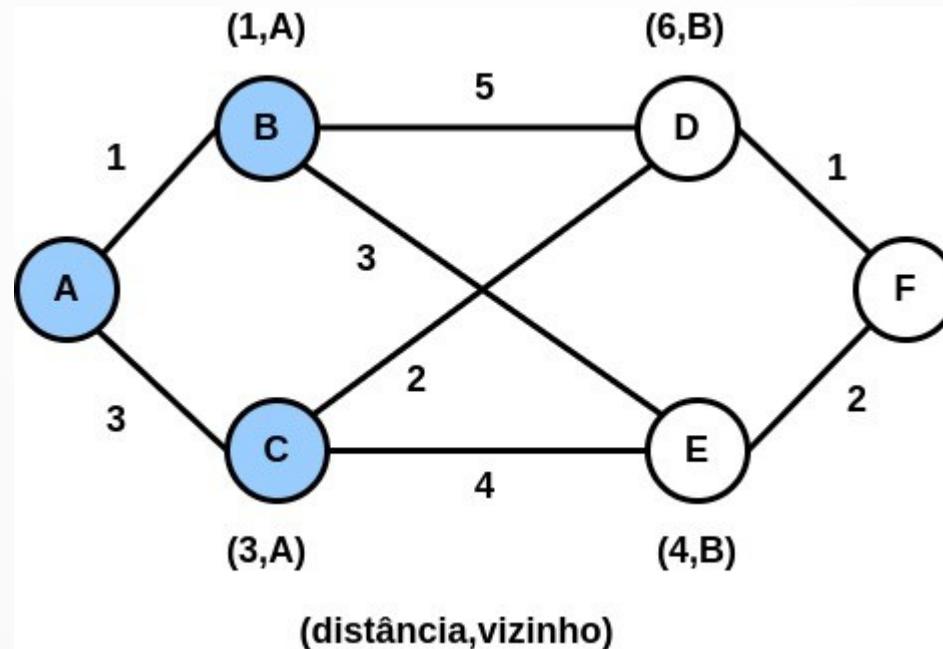
# Passo a Passo: Marca Definitivos

- Qual o menor caminho de A para F?
- Inicialmente A definitivo, distância 0 (zero)
- Depois tem duas opções: B ou C → vizinhos dos nodos “definitivos” → distância mínima novo definitivo



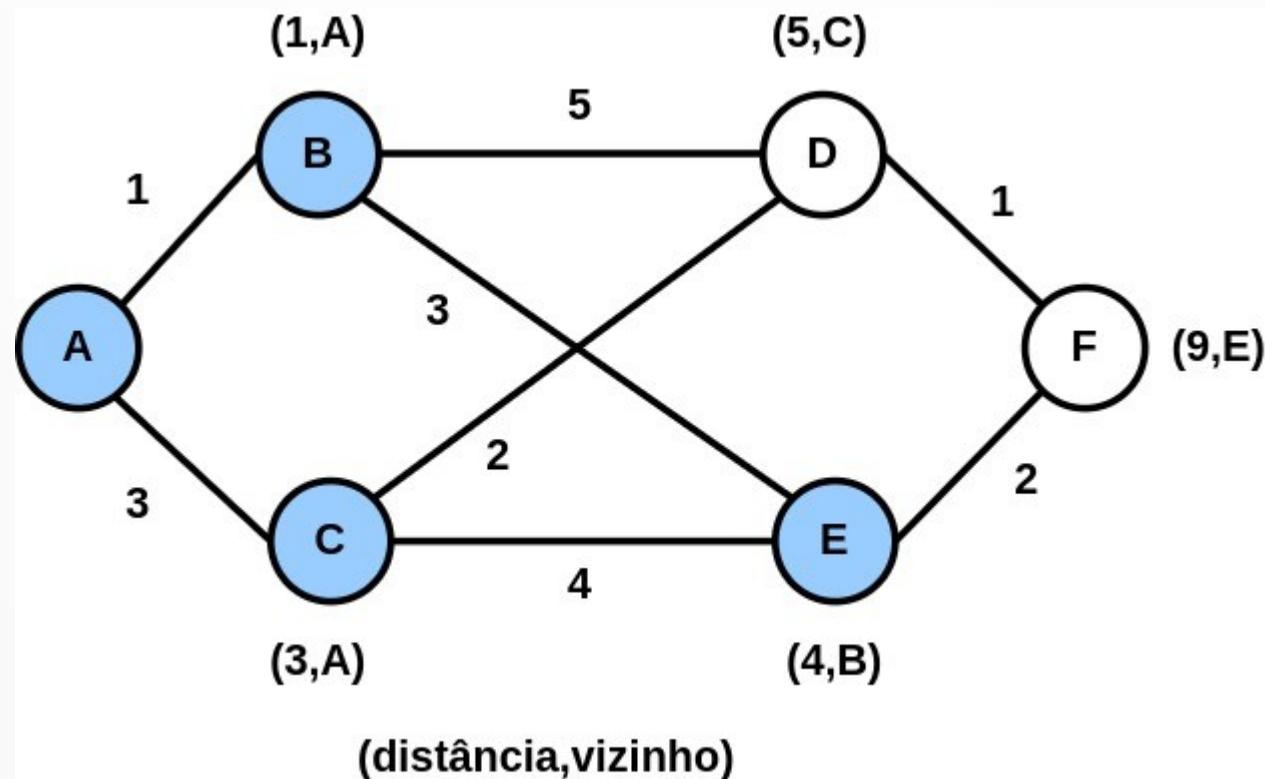
# Dijkstra

- Agora considera os vizinhos dos definitivos A e B
- Novo definitivo: C



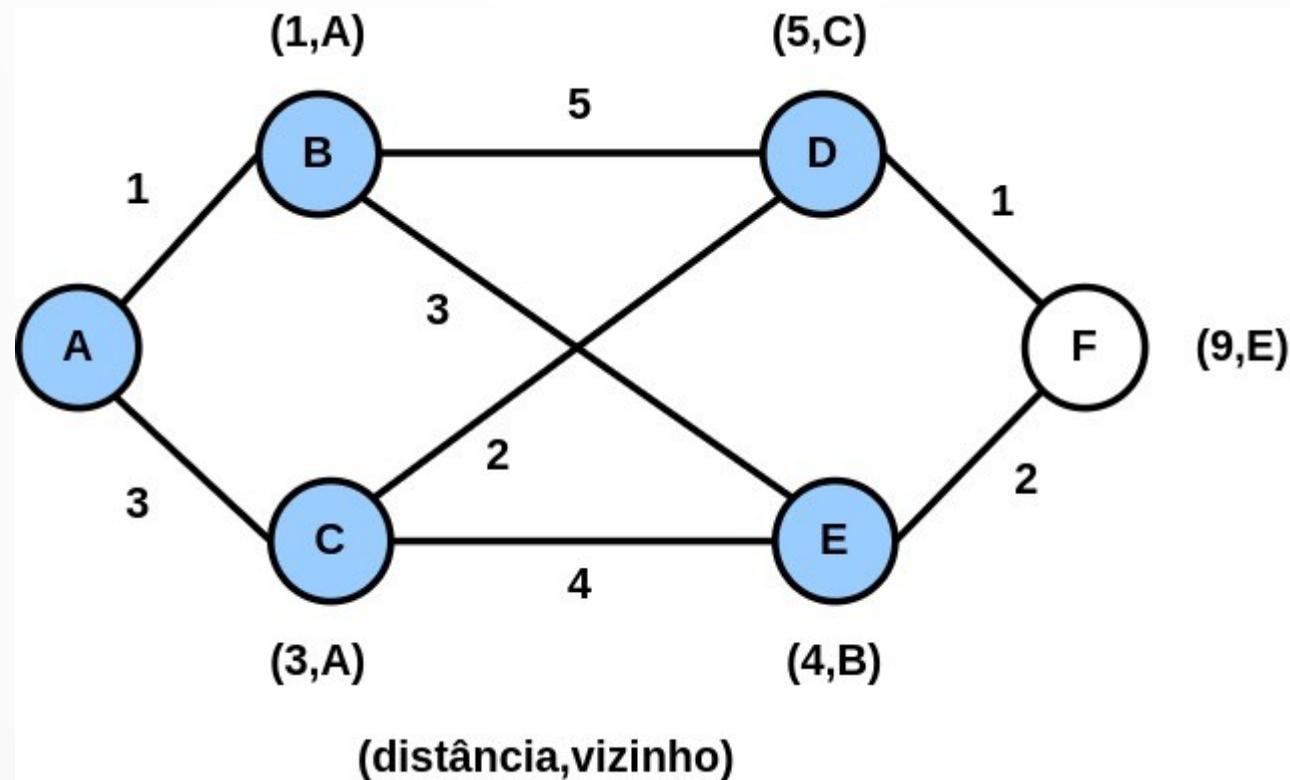
# Dijkstra

- Agora considera dos vizinhos dos definitivos A, B, e C: novo definitivo E vindo de B



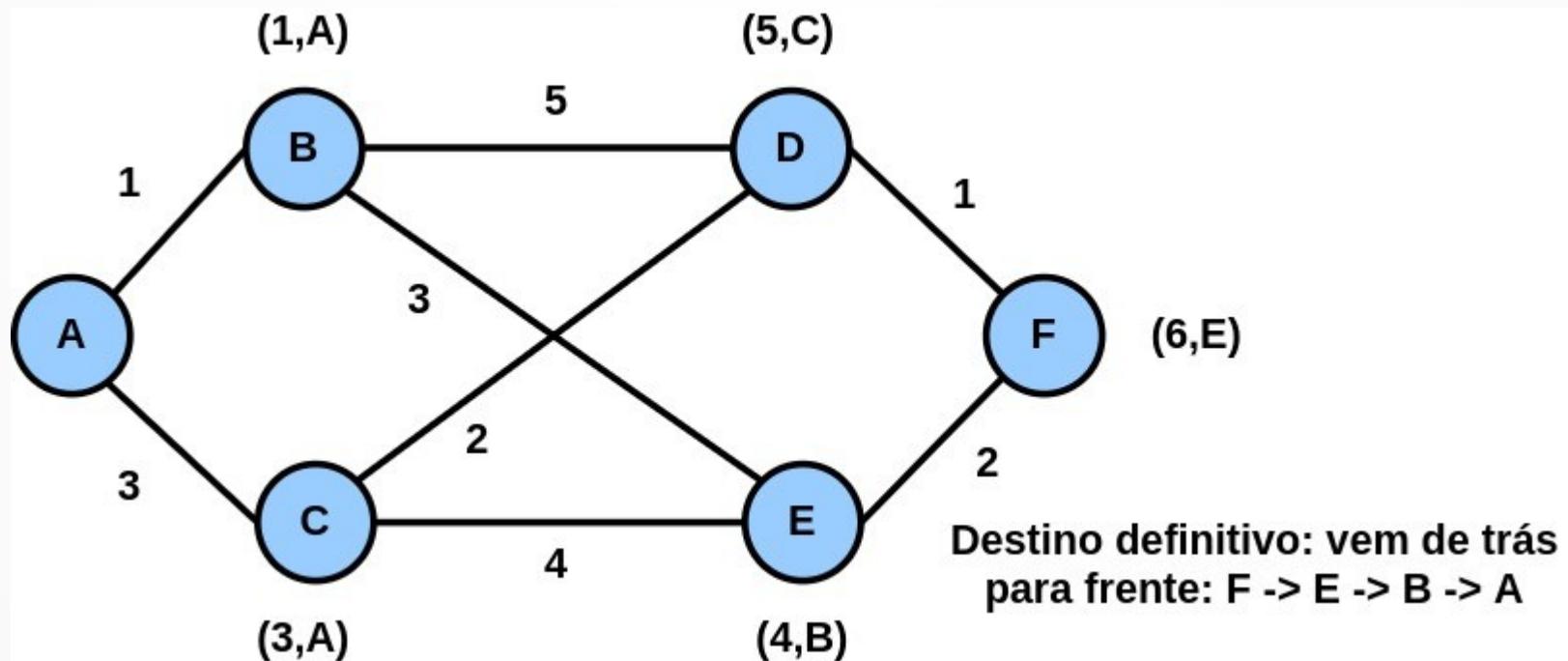
# Dijkstra

- Em seguida E pode ser marcado como definitivo



# Dijkstra

- Por fim chegamos ao destino final



# Protocolos & Algoritmos

- Na Internet alguns protocolos de roteamento implementam o algoritmo de Bellman-Ford, outros implementam Dijkstra
- Roteamento externo: tem que ser Bellman-Ford, assim o protocolo BGP implementa Bellman-Ford
- Roteamento interno: tem protocolo de roteamento que implementa
  - Bellman-Ford, por ex. RIP (*Routing Information Protocol*)
  - Dijkstra: protocolo OSPF (*Open Shortest Path First*)

# Conclusão

- Nesta aula começamos estudando a arquitetura de roteamento da Internet
- Definição importante: núcleo e borda da rede
- Definição importante: AS - Sistema Autônomo
- Roteamento interno & externo
- Protocolos & Algoritmos de Roteamento
- Algoritmos de Bellman-Ford & Dijkstra
- Próx aula: mais roteamento e MPLS

**Obrigado!**

Lembrando: a página da disciplina é:  
<https://www.inf.ufpr.br/elias/redes>