

## Definição do Trabalho Final

Desenvolver uma **aplicação** que simule a **troca de tabelas de roteamento** e o **envio de mensagens de texto** entre roteadores. A aplicação deve possuir duas partes, conforme descrito a seguir.

---

### PARTE 1

Quando um roteador inicia, ele deve ler um arquivo de configuração onde está especificado o endereço IP de quem são seus vizinhos. Neste momento, os endereços IP informados deverão ser cadastrados em uma tabela de roteamento com métrica 1 e saída com o IP do roteador vizinho. Três campos deverão estar presentes na tabela de roteamento: IP de Destino, Métrica e IP de Saída.

As tabelas de roteamento (apenas os campos IP de Destino e Métrica) serão trocadas entre os roteadores vizinhos a cada 15 segundos. Ao receber a tabela de roteamento de seus vizinhos, a aplicação deverá verificar as rotas recebidas e fazer as atualizações necessárias na tabela de roteamento local. Uma atualização deverá ser feita sempre que:

- for recebido um IP de Destino não presente na tabela local. Neste caso a rota deve ser adicionada, a Métrica deve ser incrementada em 1 e o IP de Saída deve ser o endereço do roteador que ensinou esta informação;
- for recebida uma Métrica menor para um IP Destino presente na tabela local. Neste caso, a Métrica e o IP de Saída devem ser atualizadas;
- um IP Destino deixar de ser divulgado. Neste caso, a rota deve ser retirada da tabela de roteamento.

Um roteador pode sair da rede a qualquer momento. Isso significa que seus vizinhos não receberão mais anúncios de rotas. Desta forma, depois de 35 segundos sem receber mensagens do roteador vizinho em questão, a **rota para ele e que passam por ele devem ser esquecidas** (ele está na saída de uma rota).

Periodicamente, a tabela de roteamento local deverá ser apresentada para o usuário. Além disso, alterações na tabela de roteamento deverão ser informadas para os usuários (através de prints na saída padrão).

## Protocolo de comunicação

A implementação deve respeitar fielmente o formato de mensagens descrito a seguir. A aplicação resultante deve ser **interoperável**, ou seja, **implementações de diferentes grupos devem ser capazes de se comunicar entre si**. Desta maneira, poderá ser construída uma topologia com roteadores implementados por diferentes grupos.

O protocolo de comunicação consiste em apenas duas mensagens, conforme descrito a seguir.

### Mensagem 1 - Anúncio de rotas

Esta mensagem deve ser enviada aos vizinhos a cada 15 segundos e consiste no envio da tabela de roteamento para os roteadores vizinhos. A mensagem consiste em uma lista de endereços IP em formato string. Por exemplo, se a tabela for:

IP	Métrica	Saída
192.168.1.2	1	192.168.1.2
192.168.1.3	1	192.168.1.3

A mensagem enviada será:

**@192.168.1.2-1@192.168.1.3-1**

Ou seja, “@” indica uma tupla, IP de Destino e Métrica. A métrica é separada do IP por um “-” (hífen).

O protocolo não prevê confirmação de recebimento de mensagens de roteamento, pois a tabela será reenviada a cada 15 segundos. Contudo, caso o recebimento de uma mensagem de anúncio de rotas cause a alteração da tabela de roteamento, o roteador deve enviar sua tabela imediatamente para seus vizinhos. **Os roteadores não devem incluir rotas para eles mesmos.**

### Mensagem 2 - Anúncio de roteador

Esta mensagem será enviada aos vizinhos (endereços IP que foram informados pelo usuário no arquivo de configuração) **somente quando o roteador entrar em uma rede já existente**. Neste caso, o roteador deverá anunciar-se para os vizinhos para que estes os incluam em suas tabelas de roteamento com métrica 1.

A mensagem enviada será:

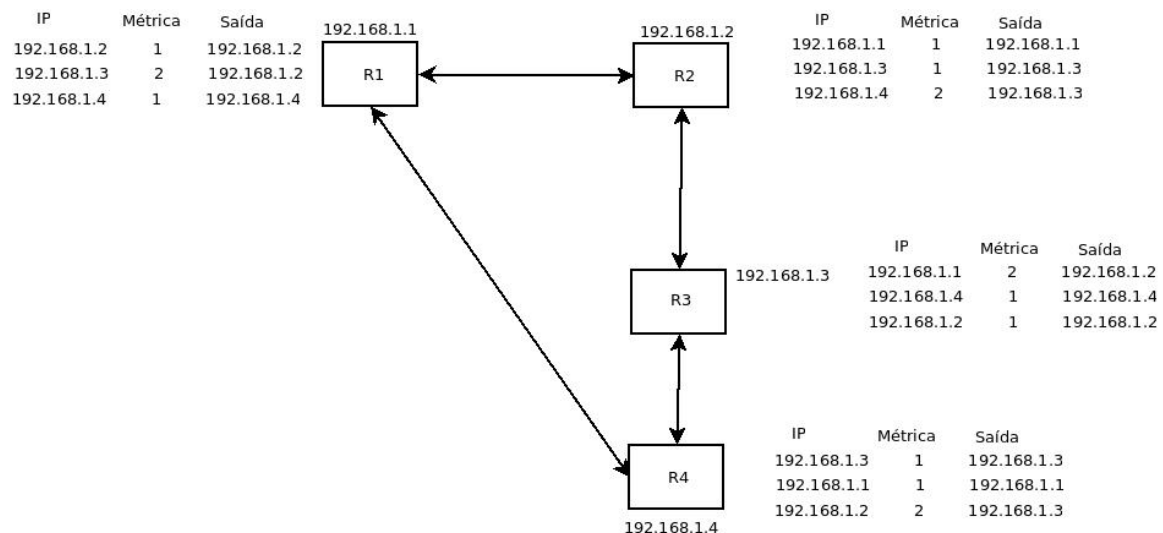
**\*192.168.1.1**

Ou seja, um \* (asterisco) seguido do próprio endereço IP do roteador que entrou na rede.

Após o seu anúncio, o roteador que entrou na rede já existente deve criar a sua tabela de roteamento com os endereços IP informados pelo usuário e seguir o funcionamento do protocolo já especificado.

### Exemplo de topologia com 4 roteadores

A figura abaixo ilustra 4 roteadores e suas respectivas tabelas de roteamento depois de algumas iterações para troca de tabelas. As flechas indicam roteadores vizinhos.



---

## PARTE 2

A segunda parte da implementação consiste na possibilidade do **envio de mensagens de texto entre os roteadores**, sendo que as mensagens devem ser roteadas de acordo com as tabelas de roteamento das máquinas.

A qualquer momento o usuário da aplicação pode enviar **mensagens de texto** para os roteadores. Neste caso, será necessário informar o **endereço IP do roteador destino e a mensagem de texto**.

O roteador deve aceitar apenas mensagens de texto direcionadas para ele. Caso a mensagem não seja para ele, será necessário roteá-la para o próximo roteador do caminho ou para o destino, conforme está especificado em sua tabela de roteamento.

Ao receber uma mensagem de texto, o roteador deve **imprimir a mensagem recebida, bem como o endereço IP da origem e o IP do destino e indicar se a mensagem está sendo repassada adiante ou se chegou ao destino**.

Uma mensagem de texto deve ter o seguinte formato:

**!192.168.1.2;192.168.1.1;Oi tudo bem?**

Ou seja, “!” indica que uma mensagem de texto foi recebida. O primeiro endereço é o IP da origem, o segundo é o IP de destino e a seguir vem a mensagem de texto. Cada informação é separada um “;” (ponto e vírgula).

---

### **Considerações sobre a implementação**

Os endereços IP dos roteadores vizinhos devem ser informados no arquivo denominado **roteadores.txt**, sendo especificado um endereço IP por linha.

A interoperabilidade é uma questão fundamental em redes de computadores. Ela permite que equipamentos de diferentes fabricantes operem em harmonia na rede. Para isso, as **especificações dos protocolos devem ser rigidamente implementadas**. Espera-se neste trabalho, que os diversos grupos apresentem implementações coerentes com a especificação e que permita a interoperabilidade com as implementações dos outros grupos.

A implementação deve utilizar socket **UDP** e as mensagens devem ser enviadas para a porta **9000** do roteador vizinho.

## Regras Gerais

---

**Grupos:** Até 4 componentes

**Data de entrega:** 13/11

**Obs.: Todos os participantes devem estar presentes**

### **Visualização dos Resultados:**

- A demonstração deverá acontecer, no mínimo, em 4 máquinas. Os grupos serão convidados a colocar seus roteadores em uma topologia definida pelo professor e que envolverá diversos grupos.

**IMPORTANTE:** Não serão aceitos trabalhos entregues fora do prazo. Trabalhos que não compilam ou que não executam não serão avaliados. Todos os trabalhos serão analisados e comparados. Caso seja identificada cópia de trabalhos, todos os trabalhos envolvidos receberão nota ZERO.