



**CST em Sistemas para a Internet
Programação Orientada a Objetos
Prof. Frederico Costa Guedes Pereira**

Roteador IPv4

Documento de Especificação

1 Descrição do problema

Deseja-se implementar um simulador de roteador IP utilizando-se a linguagem de programação Java. O simulador deve ser capaz de configurar sua tabela de rotas e, recebendo um número IP, indicar por qual interface o datagrama seria entregue.

Uma rota é definida como sendo um registro das seguintes informações: endereço de destino, endereço do *gateway*, máscara de sub-rede e interface. Uma tabela de rotas é um conjunto destes registros. Exemplo de tabela de rotas:

Destination	Gateway	Mask	Interface
127.0.0.1	127.0.0.1	255.0.0.0	lo0
10.0.2.0	10.0.2.2	255.255.255.0	eth0
10.0.3.0	10.0.3.2	255.255.255.0	eth1
10.0.1.0	10.0.2.1	255.255.255.0	eth0
10.0.4.0	10.0.3.3	255.255.255.0	eth1
0.0.0.0	10.0.3.3	0.0.0.0	eth1
10.0.3.22	10.0.3.2	255.255.255.255	eth0

A primeira coluna é o endereço de destino, a segunda é o endereço do gateway para onde o pacote deve ser enviado, a quarta é a máscara de sub-rede desta rota e a última é o nome da interface física (ethernet) a ser usada. Existem alguns endereços de destinos especiais em toda tabela de rotas, como o endereço *localhost* (127.0.0.1) e de rota *default* (0.0.0.0). A rota default é aquela rota a ser usada quando nenhuma outra foi encontrada na tabela pelo algoritmo de *longest match*.

Uma vez configurada a tabela, o roteador está pronto para realizar a simulação do roteamento. O usuário deve fornecer um número IP a ser roteado. O programa deve guardar também os números IPs das interfaces do roteador. Após receber o número a ser roteado, o programa deve imprimir qual foi a rota escolhida, utilizando o algoritmo de roteamento para o protocolo IP. Exemplos de respostas:

200.129.68.2 → (0.0.0.0, 10.0.3.3, 0.0.0.0, eth1)

10.0.4.2 → (10.0.4.0, 10.0.3.3, 255.255.255.0, eth1)

10.0.3.22 → (10.0.3.22, 10.0.3.2, 255.255.255.255, eth0)

2 Requisitos Funcionais do Sistema

UC01 – Usuário cadastrada interface física
Pré-condição: Não existir interface física com o mesmo nome
1) Usuário solicita cadastro de interface física 2) Sistema pede o nome da interface e endereço IP 3) Sistema verifica se não existe interface com este nome 4) Se não existir, sistema cadastrada interface
Pós-condição: Interface cadastrada no sistema

UC02 – Usuário cadastrada rota
Pré-condição: Não há
1) Usuário solicita cadastro de rota 2) Sistema pede o endereço de destino, máscara de subrede, roteador (gateway) de destino e interface física 3) Usuário informa dados 4) Sistema verifica se já há uma rota idêntica, se houver informa erro, senão, registra a rota
Pós-condição: Rota cadastrada no sistema

UC03 – Usuário visualiza tabela de rotas
Pré-condição: Não há
1) Usuário solicita estado atual da tabela de rotas (listagem) 2) Sistema imprime na interface a tabela completa ou a mensagem "Tabela de rotas vazia"
Pós-condição: Nenhuma

UC04 – Usuário altera rota
Pré-condição: Há rota cadastrada para ser alterada
1) Usuário solicita alteração de rota na tabela 2) Sistema imprime a tabela e pede para usuário identificar rota a ser alterada 3) Usuário informa que rota quer alterar 4) Sistema trás dados da rota e pede novamente todos os dados, dando a opção de manter os atuais ou ler um novo valor (se o usuário apenas teclar enter num dos dados da rota, permanece o atual). 5) Sistema informa que rota foi atualizada
Pós-condição: Rota alterada no sistema

UC05 – Usuário exclui rota
Pré-condição: Há rota cadastrada para ser excluída
1) Usuário solicita exclusão de rota na tabela 2) Sistema imprime a tabela e pede para usuário identificar rota a ser excluída 3) Usuário informa que rota quer excluir

- | |
|--|
| 4) Sistema pede confirmação da exclusão |
| 5) Usuário confirma exclusão |
| 6) Sistema informa que rota foi excluída |

Pós-condição: Rota excluída no sistema

UC06 – Usuário configura exibição das rotas

Pré-condição: Nenhuma

- | |
|---|
| 1) Usuário solicita configuração da exibição dos endereços de destino das rotas |
| 2) Sistema apresenta 2 opções: 'Máscara de subrede' ou 'Notação CIDR ¹ ' |
| 3) Usuário escolhe opção |
| 4) Sistema informa que configuração escolhida |

Pós-condição: A exibição das rotas passa a usar a notação escolhida doravante

UC07 – Usuário pede roteamento para IP

Pré-condição: Tabela de rotas ter rotas cadastradas

- | |
|---|
| 1) Usuário solicita a opção de rotear datagrama |
| 2) Sistema pede o IP a ser roteado |
| 3) Usuário informa o IPv4 |
| 4) Sistema exibe a melhor rota a ser usada pelo algoritmo de <i>longest match</i> |

Pós-condição: Nenhuma

UC08 – Usuário pede para resetar tabela de rotas

Pré-condição: Nenhuma

- | |
|--|
| 1) Usuário solicita a opção de resetar tabela de rotas |
| 2) Sistema pede confirmação da operação |
| 3) Usuário confirma reset |
| 4) Sistema reseta a tabela de rotas |

Pós-condição: Tabela de rotas vazia

O sistema pretendido deverá ser capaz de aceitar as seguintes configurações:

- Número de interfaces físicas do roteador (portas eth, lp, sl, etc.)
- Endereço IP destas interfaces físicas
- Inclusão/remoção/alteração de rotas no formato especificado acima
- Listagem atual da tabela de rotas
- Escolha da rota mais adequada (utilizando o algoritmo de *longest match*)

O sistema deve receber um número IP do suposto datagrama a ser roteado. Após recebê-lo, deve informar qual a melhor rota a ser tomada por ele, dentre as possíveis na tabela.

¹ CIDR significa Classless Inter-Domain Routing. É a notação mais compacta que mostra os endereços de destino das rotas com IP/bits. Exemplo: 192.168.1.2/16. A opção máscara de subrede exibiria este mesmo endereço assim: 192.168.1.2 255.255.0.0.

3 Requisitos não funcionais

Você deve observar as seguintes recomendações para a implementação do software:

- Modelagem orientada a objetos das entidades do domínio do problema
- Indentação do código fonte
- Comentários (apenas o necessário)
- Nomenclatura Java para os identificadores de classes, métodos, variáveis etc.
- Facilidade de uso da interface em geral
- Utilizar Git como sistema de versionamento

Quaisquer dúvidas sobre os requisitos funcionais e não funcionais devem ser perguntadas ao professor. Se você encontrar um erro ou inconsistências neste documento, envie para fred@ifpb.edu.br para eu fazer a correção.

4 Algoritmo de *Longest Match*

Quando um roteador recebe um pacote, ele verifica o endereço IP de destino e precisa decidir qual rota usar em sua tabela de roteamento.

Cada entrada na tabela de roteamento é um prefixo de rede (como 192.168.1.0/24), que significa “todos os IPs que compartilham os primeiros n bits”. Este endereço também pode ser especificado como o IP e sua máscara de rede. Assim, uma máscara de rede com 24 bits (exemplo) seria 255.255.255.0 (cada 255 tem 8 bits 1, totalizando 24 bits 1 nos três desta máscara. Assim, tanto faz dizer 192.168.1.0/24 como 192.168.1.0 e máscara 255.255.255.0. A forma com a quantidades de bits 1 é mais resumida e usada.

O roteador encontra todas as entradas que correspondem ao IP de destino.

Se múltiplas rotas corresponderem, o roteador escolhe aquela com o prefixo mais longo (ou seja, a rota mais específica, com o maior valor). Isso é chamado de regra de correspondência de prefixo mais longa (LPM).

Imagine um roteador com as seguintes rotas:

Destination Prefix	Next Hop
0.0.0.0/0	ISP gateway (default route)
10.0.0.0/8	Router A
10.1.0.0/16	Router B
10.1.2.0/24	Router C
192.168.1.0/24	Router D

Exemplo 1: Qual rota seria escolhida para rotear um datagrama IP com o endereço de destino 10.1.2.45? Note que vários endereços de destino são parecidos com este IP que tem o valor 10 no primeiro octeto (byte).

Quando verificássemos qual é a rota mais adequada teríamos o seguinte cenário:

Vamos ver quais rotas correspondem:

- 0.0.0.0/0 → corresponde a tudo (padrão).
- 10.0.0.0/8 → corresponde porque 10.1.2.45 começa com 10.
- 10.1.0.0/16 → corresponde porque os primeiros 16 bits são 10.1.
- 10.1.2.0/24 → corresponde porque os primeiros 24 bits são 10.1.2.

- 192.168.1.0/24 → sem correspondência.

A rota default é a que tem mais casamentos, mas ela só será usada se nenhuma das demais rotas tiver nenhuma correspondência. Temos correspondências em outras 3 rotas (para Router A, Router B e Router C). Das 3, a rota que teve mais correspondência foi a para o Router C, pois 24 bits eram iguais nos dois endereços.

Exemplo 2: se pedíssemos para rotar o endereço 10.3.5.7 teríamos o seguinte cenário:

- 0.0.0.0/0 → corresponde a tudo.
- 10.0.0.0/8 → casa com 8 bits (10.3.5.7 começa com 10).
- 10.1.0.0/16 → sem correspondência.
- 10.1.2.0/24 → sem correspondência.
- 192.168.1.0/24 → sem correspondência.

Ou seja, seria roteado para o Router A.

Exemplo 3: A rota para 8.8.8.8 casaria para a rota default (0.0.0.0/0), pois este IP não teria correspondência com nenhuma das demais rotas. Por isso ela é chamada rota *default*. De alguma maneira um datagrama é rotado, mesmo que este roteador não saiba uma boa rota para ele, adota como estratégia remetê-lo para outro roteador com mais condições de rotear o datagrama (o default gateway da rede).

Por que *Longest Match*? Se voltarmos ao exemplo 1 acima, o IP 10.1.2.45 seria roteado para o Router A porque esta entrada seria a primeira a ser verificada, mas ela casa com apenas 8 bits deste IP. A melhor rota não é esta, mas sim a que leva ao Router C cujo casamento é de 24 bits entre o endereço de destino desta rota e o IP a ser roteado (ambos começam com 10.1.2)