



Curso de Tecnologia em Sistemas de Computação
Disciplina: Redes de Computadores II
AD1 - GABARITO - 2º semestre de 2010

1. [1 ponto] Quais são as duas principais funcionalidades da camada de redes? Explique cada uma delas de forma sucinta, indicando a diferença entre elas.

As duas principais funcionalidades da camada de redes são o encaminhamento e o roteamento. O encaminhamento determina como um pacote deve atravessar um roteador. Ou seja, dado um determinado pacote e sua porta de chegada, o encaminhamento define a porta de saída pela qual o pacote deve ser transmitido. O roteamento define a rota que os pacotes devem seguir desde a origem até o destino. O algoritmo de roteamento determina como o encaminhamento nos roteadores deve ser feito.

2. [0.5 pontos] Qual é a diferença entre encaminhamento e roteamento?

Roteamento tem a ver com a descrição e também determinação do caminho fim-a-fim a ser tomado por um pacote que atravessa a rede. Por exemplo, enumerando os enlaces da rede que devem ser atravessados desde o nó origem até o nó destino. Encaminhamento tem a ver com a determinação da porta de saída (enlace) de um roteador que um pacote determinado pacote deve tomar. Por exemplo, dada alguma informação no pacote (ex. endereço de destino), determinar a porta de saída pela qual o pacote deve ser transmitido.

3. [1 ponto] Descreva o que consta em uma tabela de roteamento de um roteador de uma rede orientada a conexão (circuito virtual) e de uma rede sem conexão (datagrama). Quais são as diferenças?

A tabela de roteamento em uma rede orientada a conexão consiste de números de circuitos virtuais e portas de saída. Esta informação é utilizada para encaminhar os pacotes da seguinte forma: ao chegar um pacote, o roteador utiliza o número do circuito virtual do pacote para determinar por qual porta de saída o pacote deve ser encaminhado. Além disso, o número do circuito virtual pode mudar, sendo atualizado pelo roteador antes do pacote ser transmitido.

A tabela de roteamento em uma rede sem conexão (como a Internet), consiste de endereços destinos e portas de saída. Esta informação é utilizada para encaminhar os pacotes da seguinte forma: ao chegar um pacote, o roteador utiliza o endereço destino do pacote para determinar por qual porta de saída o pacote deve ser encaminhado.

Um diferença entre os dois tipos de roteamento é que em redes com circuito virtual, o pacote precisa ser modificado pelo roteador (para escrever o número do novo circuito virtual), enquanto tal modificação não é necessária em redes sem conexão.

4. [0.5 pontos] O que consta nas tabelas de roteamento dos roteadores da Internet? Explique como tal informação é utilizada para rotear pacotes.

Os roteadores da Internet, por ser uma rede sem conexão, possui tabelas de roteamento que consistem de prefixos e portas de saída. Ao chegar um pacote, o roteador utiliza o endereço destino do pacote para fazer uma busca pela tabela de roteamento. O roteador busca pelo prefixo mais longo que coincide o endereço destino. O pacote então é encaminhado pela porta de saída correspondente a este prefixo.

5. [1 ponto] Descubra o endereço IP do servidor Web do CEDERJ (www.cederj.edu.br). Converta este número IP para sua representação binária (zeros e uns). Dica: para descobrir o endereço IP do servidor, utilize o comando *ping*.

O endereço IP do servidor é: **200.156.70.10**

Sua representação binária é: **11001000.10011100.01000110.00001010**

6. [1 ponto] Suponha que um aplicativo envia 70 bytes de dados através de um socket TCP. Tal informação é encapsulado em um segmento TCP e posteriormente em um pacote IP. Qual o tamanho do pacote depois destes dois encapsulamentos? Qual o percentual de *overhead* (tamanho dos cabeçalhos dividido pelo tamanho total)?

O cabeçalho do protocolo IP é de 20 bytes. O cabeçalho do protocolo TCP também é de 20 bytes. Logo, o pacote possui 40 bytes de cabeçalho. Como o pacote possui 70 bytes de dados, seu tamanho será de $40 + 70 = 110$ bytes. O overhead será $40/110 = 0.36$.

7. [3 pontos] Suponha que um roteador da Internet possua 4 portas, enumeradas de 0 a 4, onde os pacotes devem encaminhados de acordo com a tabela abaixo:

Faixa do Endereço Destino	Interface
11110000 00000000 00000000 00000000 à 11110000 00111111 11111111 11111111	0
11110010 00000000 11000000 00000000 à 11110010 00000000 11111111 11111111	1
11110010 00000000 11100111 00000000 à 11110010 00000000 11100111 11111111	2
11110110 10100000 11100000 10000000 à 11110110 10100000 11100000 11111111	2
caso contrário	3

- (a) Construa a tabela de roteamento com base nas informações acima. Sua tabela deve conter 4 entradas, utilizar prefixo mais longo e encaminhar os pacotes para as interfaces corretas.

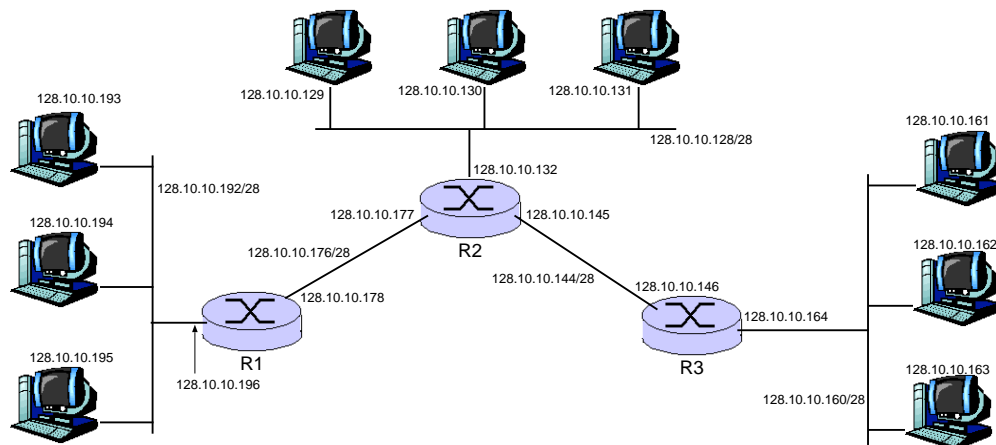
Prefixo	Interface
11110000 00	0
11110010 00000000 11	1
11110010 00000000 11100111	2
11110110 10100000 11100000 1	2
-	3

- (b) Escreva a tabela de roteamento do item acima utilizando a notação $a.b.c.d/x$ ao invés de utilizar números binários.

Prefixo	Interface
240.0.0.0/10	0
242.0.192.0/18	1
242.0.231.0/24	2
246.160.224.128/25	2
0.0.0.0/0	3

- (c) Descreva para qual porta de saída os pacotes com os seguintes endereços destino serão encaminhados:

Endereço Destino do Pacote	Porta de saída
11110000 11110110 11011011 11000111	3
11110010 00000001 11110001 10111010	3
11110010 00000000 11000111 00001111	1
11110110 10000000 11110000 11100011	3
11110110 10100000 11111100 11100011	3
11111000 10100000 11111100 11100011	3



8. [1 ponto] Considere a rede ilustrada na figura acima. Suponha que todos os endereços IPs desta rede precisam pertencer à classe 128.10.10.128/23. Configure a rede atribuindo endereços IPs a cada uma das interfaces e também a cada uma das sub-redes, utilizando nesta caso a notação $a.b.c.d/x$.

Repare que precisamos criar 5 redes locais diferentes: uma para cada segmento ilustrado na figura. Podemos dividir o endereço 128.10.10.128/23 em 5 redes, por exemplo, utilizando máscaras /28. Repare que ao usarmos /28 podemos ter até 8 redes (pois os bits 26, 27 e 28 serão usados para indentificar a rede). Além disso, ao usarmos /28 cada uma das sub-redes pode ter até 15 máquinas (pois os bits 29, 30, 31 e 32 serão usados para endereçar a máquina dentro da sub-rede). A figura abaixo ilustra uma possível configuração com base em redes /28, que não é a única solução correta.

9. [1 ponto] Considere um único roteador que interconecta três redes locais, L_1 , L_2 , e L_3 . Suponha que todas as interfaces (i.e., números IPs) destas redes precisam obrigatoriamente pertencer a rede 100.23.44.0/22. Você deseja alocar 100 números IPs para a rede local L_1 , 80 números IPs para a rede local L_2 e 60 para a rede local L_3 . Forneça os endereços das redes locais na forma $a.b.c.d/x$ para atender tais requerimentos (repare que há várias soluções corretas).

Repare que temos um endereço /22, o que nos permite ter até 1023 ($2^{10} - 1$, pois temos 10 bits) endereços IPs distintos. Entretanto, tais IPs precisam estar divididos em três redes locais distintas. Vamos definir então três diferentes endereços de rede local, utilizando para cada uma delas, uma máscara /24. Isto nos dá até 4 redes locais diferentes, onde cada uma pode endereçar até 255 endereços distintos. Para a rede local L_1 temos 100.23.45.0/24, o que nos dá um total de 255 endereços IPs. Para a rede local L_2 temos 100.23.46.0/24, o que nos dá um total de 255 endereços IPs. Para a rede local L_3 temos 100.23.47.0/24, o que nos dá um total de 255 endereços IPs. Repare que esta solução atende os requerimentos acima. Repare ainda que existem outras possíveis soluções.

10. [1 ponto] Suponha que um aplicativo envia de uma só vez 20KB (1KB = 1000 bytes) de dados que irão atravessar um link que possui um MTU de 1460 bytes (que é o caso do link Ethernet). Quantos pacotes na camada de rede serão gerados? Suponha que o primeiro fragmento IP possua número de identificação 3218. Quais os números de identificação e *offset* que serão utilizados em cada pacote?

Ignorando o cabeçalho introduzido pelos protocolos de transporte e de redes, temos um total de 14 pacotes transmitidos, pois $\lceil 20000/1460 \rceil = 14$. O primeiro pacote possui número de identificação 3218 e offset 0, o segundo pacote identificação 3218 e offset 1460, o terceiro pacote identificação 3218 e offset 2920, até o último pacote, que possui identificação 3218 e offset 18980. Repare que todos os pacotes possuem o mesmo número de identificação. Os treze primeiros pacotes são do mesmo tamanho (1460 bytes de dados), enquanto o último pacote possui 1020 bytes de dados.

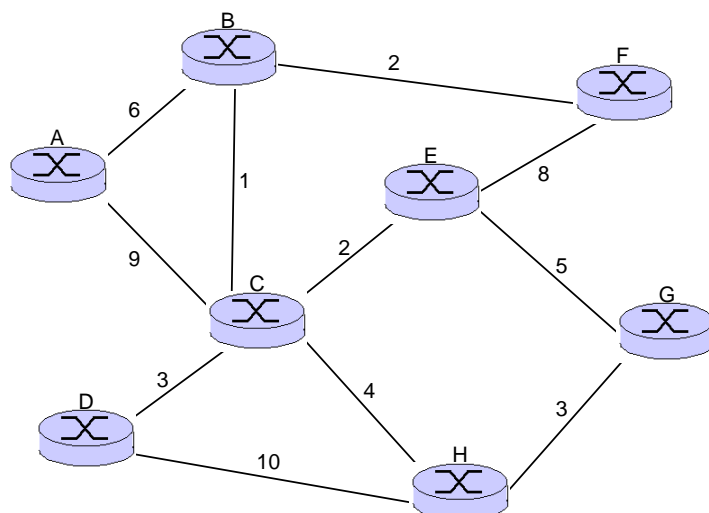
11. [1 ponto] Suponha que você irá baixar um arquivo MP3 da Internet que possui 4.8MB (1MB = 10^6 bytes) de tamanho. Se você está ligado a uma rede local tipo Ethernet, cujo MTU é 1450 bytes, quantos pacotes IP serão necessários (no mínimo) para efetuar esta transferência? Faça as contas ignorando os cabeçalhos dos protocolos TCP e IP, e depois refaça as contas considerando estes dois cabeçalhos.

Ignorando o cabeçalho introduzido pelos protocolos TCP e IP, temos um total de 3000 pacotes (no mínimo devido a possíveis retransmissões), pois $\lceil 4800000/1450 \rceil = 3311$. O cabeçalho dos protocolos TCP e IP, juntos, adicionam 40 bytes (20 bytes de cada protocolo). Assim sendo, a aplicação consegue enviar somente 1460 bytes por pacote, pois $1450 - 40 = 1410$. Logo serão necessários 3405 pacotes para efetuar a transmissão, pois $\lceil 4800000/1410 \rceil = 3405$.

12. [1.5 pontos] Explique como funciona o mecanismo NAT. Dê um exemplo para facilitar sua explicação onde um *host* dentro da sub-rede do NAT abre uma conexão com um *host* na Internet. Cite ainda as vantagens e desvantagens deste mecanismo.

O NAT é um mecanismo que permite criar uma rede local privada, com endereços IPs que não são vistos diretamente por hosts na Internet pública. O NAT-box é um computador especial que permite que um host na rede local privada faça uma conexão com um host na rede pública da Internet, traduzindo os endereços. Ao abrir uma conexão TCP com um servidor Web na Internet, por exemplo, um host na rede privada atravessa o NAT-box de forma transparente para ambos (host e servidor Web). O NAT-box anota o endereço e porta de origem do pacote vindo da rede local privada, e muda a porta e o endereço do pacote antes de transmiti-lo pelo link de saída (que dá acesso a Internet). O NAT-box mantém uma tabela com este mapeamento. O novo endereço de origem deste pacote é o endereço IP público do NAT-box. Ao receber um pacote vindo da Internet, o NAT-box consulta sua tabela de mapeamento e faz o procedimento reverso.

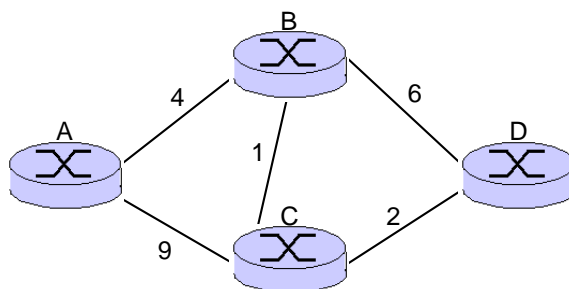
Algumas vantagens do mecanismo NAT é ter um endereçamento dos hosts independente de endereços IPs válidos na Internet pública. Isto permite um melhor gerenciamento da rede, pois não restringe a alocação de endereços. Outra vantagem é a segurança, uma vez que hosts na rede local privada não podem ser acessados diretamente de um host na Internet pública. Uma desvantagem é exatamente esta vantagem, pois devido a esta característica, aplicativos que se comunicam diretamente com os hosts podem não funcionar (como aplicativos peer-to-peer, por exemplo). Outra desvantagem é o possível gargalo que pode ser formado no NAT-box, degradando o desempenho da rede.



13. [2.0 pontos] Considere a rede ilustrada acima, onde os enlaces estão anotados com seus respectivos custos. Utilize o algoritmo de Dijkstra para calcular o caminho mais curto do nó *A* para todos os outros nós da rede. Construa uma tabela igual a mostrada em aula que demonstra o funcionamento do algoritmo de forma iterativa.

Resultado do processo iterativo está ilustrado na tabela abaixo:

Passo	N'	d(B),p(B)	d(C),p(C)	d(D),p(D)	d(E),p(E)	d(F),p(F)	d(G),p(G)	d(H),p(H)
0	A	6,A	9,A	∞	∞	∞	∞	∞
1	AB		7,B	∞	∞	8,B	∞	∞
2	ABC			10,C	9,C	8,B	∞	11,C
3	ABCF			10,C	9,C		∞	11,C
4	ABCFE			10,C			14,E	11,C
5	ABCFED						14,E	11,C
6	ABCFEDH						14,E	
7	ABCFEDHG							

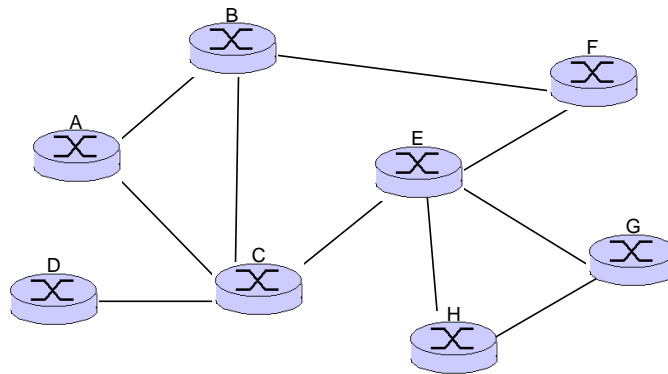


14. [2.0 pontos] Considere a rede ilustrada acima, onde os enlaces estão anotados com seus respectivos custos. Utilize o algoritmo *distance vector* para calcular as tabelas de distância de cada um dos nós. Assuma uma versão síncrona do algoritmo, e mostre a evolução das tabelas em cada nó, conforme visto em aula.

Legenda: oo - infinito

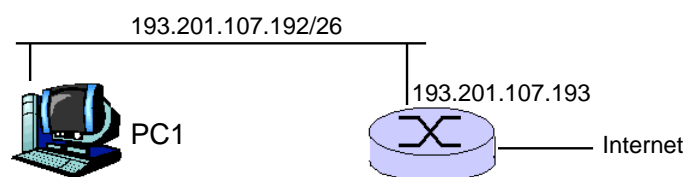
Passo 0					Passo 1					Passo 2				
tabela do nó A					tabela do nó A					tabela do nó A				
		custo até					custo até					custo até		
		B	C	D			B	C	D			B	C	D
via	B	4	oo	oo	via	B	4	5	10	via	B	4	5	7
	C	oo	9	oo		C	10	9	11		C	10	9	11
tabela do nó B					tabela do nó B					tabela do nó B				
		custo até					custo até					custo até		
		A	C	D			A	C	D			A	C	D
via	A	4	oo	oo	via	A	4	13	oo	via	A	4	9	14
	C	oo	1	oo		C	10	1	3		C	6	1	3
	D	oo	oo	6		D	oo	8	6		D	16	8	6
tabela do nó C					tabela do nó C					tabela do nó C				
		custo até					custo até					custo até		
		A	B	D			A	B	D			A	B	D
via	A	9	oo	oo	via	A	9	13	oo	via	A	9	13	19
	B	oo	1	oo		B	5	1	7		B	5	1	4
	D	oo	oo	2		D	oo	8	2		D	12	5	2
tabela do nó D					tabela do nó D					tabela do nó D				
		custo até					custo até					custo até		
		A	B	C			A	B	C			A	B	C
via	B	oo	6	oo	via	B	10	6	7	via	B	10	6	7
	C	oo	oo	2		C	11	3	2		C	7	3	2

Evolução das tabelas do algoritmo distance vector.



15. [1.5 pontos] Considere a rede ilustrada acima. Suponha que o nó A deseja enviar uma determinada informação a todos os outros nós da rede. Utilizando o algoritmo de *flooding*, mostre os pacotes que serão enviados por cada nó da rede, indicando também o instante de transmissão de cada um deles (a ser decidido por você). Por exemplo, $A \rightarrow B, t = 1$.

Mensagens enviadas
$A \rightarrow B, t = 1$
$A \rightarrow C, t = 1$
$B \rightarrow C, t = 2$
$B \rightarrow F, t = 2$
$C \rightarrow B, t = 2$
$C \rightarrow D, t = 2$
$C \rightarrow E, t = 2$
$E \rightarrow F, t = 3$
$E \rightarrow G, t = 3$
$E \rightarrow H, t = 3$
$F \rightarrow E, t = 3$
$G \rightarrow H, t = 4$
$H \rightarrow G, t = 4$



16. [1 ponto] Considere a rede local ilustrada na figura acima. Suponha que o computador PC1 está rodando Linux. Configure o endereço IP e o gateway de saída do PC1, dando os comandos necessários (vistos na aula prática). Escolha qualquer endereço IP válido da rede local para o PC1.

Comandos:

```
ifconfig eth0 193.201.107.194 netmask 255.255.255.192 up
route add default gw 193.201.107.193
```

Qualquer endereço IP na faixa 193.201.107.193 - 193.201.107.255 pode ser usado nesta subrede (num total de 63 endereços).