

**Lista - Camada de Transporte**  
Redes de Computadores  
Instituto de Ciência e Tecnologia - ICT  
Universidade Federal de São Paulo - UNIFESP  
2o semestre de 2021

Observações:

- Lista de exercícios para fixação de conteúdo.
- Livro texto onde se encontram as questões:
  - KUROSE, James F.; Ross, Keith W.. Redes de computadores e a internet: uma abordagem top- down. **5.ed.** São Paulo: Addison-Wesley, 2010.
- A resolução da lista é INDIVIDUAL.
- Para cada questão, crie uma caixa de texto e a utilize para responder.
- As respostas são dissertativas. Pondere completude e objetividade nas respostas.

Nome:

Igor Ribeiro Ferreira de Matos
--------------------------------

Exercícios de Fixação (pág. 215):

**3, 4, 6, 7, 8, 9, 10, 11, 14, 15, 17**

Problemas (pag. 216)

**5, 6, 7, 14, 21, 33, 39**

Questões Dissertativas (pag. 224)

**2, 4**

**PÁGINA 215**

3. Como a conexão TCP requer uma conexão com dois pares: <ip\_origem:porta\_origem> e <ip\_dest:porta\_dest> e cada conexão utiliza o mesmo socket, então, depois de estabelecida ela funcionará para as duas vias. Desta forma, o segmento que trafega do hospedeiro B para o A terá os mesmos números de porta e destino, porém invertidos entre si.

4. Isto pode ocorrer principalmente pelo fato da UDP ser mais “simples” e “direta”. Na UDP, não se tem estabelecimento de estados de conexão entre origem e destino (o que pode gerar atraso); o cabeçalho do datagrama é menor e pode se transmitir o mais rápido possível (por não ter controle de congestionamento). Portanto, é bastante comum de ser utilizada em casos onde se pode ser tolerante à perdas e sensível a taxas.

5. Além de não haver controle de taxas de transmissão, o UDP é bastante utilizado nestes casos principalmente pela característica dela ser tolerante à perdas, ou seja, a taxa de transmissão é mais importante do que sua integridade.

6. Sim. É possível aumentar a confiabilidade através da camada de aplicação, com recuperações de erro específicas da aplicação.

7. Sim, serão encaminhados para o mesmo socket. Este, por sua vez, diferencia a origem (e, conseqüentemente, o um potencial destino) através de funções primitivas. Ao utilizar a função `recvfrom` para escutar a mensagem, o servidor terá então o ip de origem em um formato `sockaddr`. Posteriormente, ele poderá usar este ip de origem como destino, para enviar com a função `sendto`.

8. O servidor poderia tratar todas as solicitações através do mesmo socket, delimitando uma fila. Desta forma, apenas um socket seria utilizado, sem a necessidade de tentar estabelecer dois sockets em uma mesma porta.

9. Os números de sequências são fundamentais para os casos em que os pacotes NAKs ou ACKs possam estar corrompidos. Com o número de sequência, é possível retransmitir o pacote atual caso o feedback esteja corrompido: caso seja um NAK, realmente seria necessário a retransmissão, caso seja um ACK teria o problema de pacote sendo duplicado; porém, com os números de sequência, o destinatário simplesmente não subiria este pacote duplicado após verificar sua existência com o bit alternante e aguardaria pelo próximo.

10. Para lidar com os casos em que os pacotes sequer chegaram ao destinatário. Nestes casos, devemos ter um tempo razoável para considerar uma retransmissão de pacote (sem muita preocupação com a questão de duplicar pacote, por conta dos números de sequência já lidarem com isto, mas que tenha um tempo considerável e provável em que ocorrerá a perda de pacotes).

11. Sim, ainda seriam necessários. No entanto, o temporizador seria o mais otimizado possível, uma vez que sabe-se o tempo constante. Ainda pode ocorrer perda no meio da transmissão, então, é necessário considerar um temporizador para a retransmissão.

14.

- a)
- b) V
- c) V
- d) F
- e) V
- f) F
- g) V

15.

- a) O primeiro segmento tem 20 bytes.
- b) B enviará a A um segmento com o número de reconhecimento "90".

17. Idealmente, o TCP daria uma divisão justa para cada. Neste caso, cada enlace receberia uma taxa de  $R/2$  bps.

## PÁGINA 216

5. Não, pois apesar da soma estar certo, pode ser apenas uma coincidência dos fatores. Assim como o número “20” pode ser formado pelas somas “8+12” e “15+5”.

6. Quando o receptor da figura 3.57 recebe um pacote OK, ele envia um “ACK” com a a verificação de somatória. O Emissor, no entanto, não esperaria receber este tipo de dado, mas sim apenas um “0” ou “1” como indicador do número de sequência.

7. Pois os ACK não tem uma importância na questão de sua consistência. Por ser apenas uma mensagem de verificação, não é necessário garantir completamente sua integridade de dados. Vemos que, caso ele seja corrompido ou atrasado, simplesmente ocorrerá a retransmissão do pacote e o destinatário descarta caso seja um pacote duplicado.

14.  $t_{trans} = L/R = 12000000 \text{ bits/pacote} / 1 \text{ gbits/segundo} = 0,012s$

A taxa de Utilização é definida da seguinte forma:

$$U_{\text{util}} = \frac{L/R}{RTT + L/R}$$

Tomando  $RTT = 30s$ , como no exemplo da figura 3.17, o tempo de utilização de cada pacote se daria por:

$$u = x * 0,012 / 30 + 0,012$$

onde x é o tamanho da janela

A janela deveria ter um tamanho de  $n=2376$  pacotes, para que a taxa de utilização do canal atinja 95%

21. Na repetição seletiva, vemos que o tamanho da janela pode/deve ser menor ou igual à metade do número de sequência.

33. Para otimizar o tempo de retransmissão, principalmente nos casos em que há perda de pacotes e queremos “pular” ou economizar um tempo do timeout, para evitar um atraso exponencial.

39. Pois a duplicação do intervalo de temporização não é um mecanismo muito abrangente e não lidaria com todos os casos possíveis. Este método lidaria bem com a questão de perda de pacotes, principalmente. O mecanismo de controle de congestionamento com janelas, no entanto, tenta otimizar quando há um congestionamento de fila em roteadores, ou perda de pacotes por tráfego excessivo.

## PÁGINA 224

2. Poderia ter, de forma hipotética, apenas um tipo de fluxo em toda a rede de internet. Como o protocolo de TCP aplica justiça, e o UDP não - o fator de injustiça vem quando há um fluxo UDP consumindo mais taxa do que de TCP.

4. Eventualmente teria uma situação de injustiça, onde um enlace poderia ter mais conexões paralelas do que o outro, conseguindo garantir uma fração a mais de taxa de transmissão.