

Aluno: _____

Assinatura: _____

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
 2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
 3. Você pode usar lápis para responder as questões.
 4. Ao final da prova devolva as folhas de questões e as de respostas.
 5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
-

1. A camada de transporte da arquitetura TCP/IP fornece dois tipos de serviço.

- a. Cite quais são esses dois tipos de serviço. (0,5 pontos)

Resposta: O serviço de entrega não-confiável (ou não-orientado a conexão) (0,2 pontos) e o serviço de entrega confiável (ou orientado a conexão) (0,3 pontos).

- b. Cite dois protocolos de transporte que implementam cada um desses serviços e diferencie o princípio de funcionamento desses dois protocolos. (1,0 ponto)

Resposta: O *User Datagram Protocol* (UDP) e o *Transmission Control Protocol* (TCP) implementam, respectivamente, os serviços de entrega não-confiável e entrega confiável (0,2 pontos). O UDP é um protocolo não orientado a conexão, ou seja, não há conexão entre remetente e o receptor antes do envio dos dados. Assim sendo, segmentos UDP podem ser perdidos e entregues à aplicação fora de ordem, uma vez que cada segmento é tratado de forma independente. O UDP oferece os serviços mínimos da camada de transporte: multiplexação e demultiplexação e verificação de integridade (0,4 pontos). O TCP é um protocolo orientado a conexão, ou seja, antes do envio dos dados, segmentos de sinalização são trocados entre transmissor e receptor para definir parâmetros e estabelecer a conexão lógica entre os sistemas finais. É um protocolo ponto-a-ponto e *full-duplex*. O TCP oferece outros serviços além dos mínimos, como entrega confiável, controle de fluxo e controle de congestionamento (0,4 pontos).

- c. Cite dois protocolos da camada de aplicação que executam sobre cada um dos protocolos citados no Item b e explique o porquê desse uso. (1,0 ponto)

Resposta: O DNS (*Domain Name System*), o SNMP (*Simple Network Management Protocol*) e o NFS (*Network File System*) são exemplos de protocolos da camada de aplicação que usam o UDP (citar um vale 0,2 pontos). O DNS, por exemplo, é um protocolo que usa o UDP, pois precisa de baixo tempo de resposta para não prejudicar a interatividade da navegação web (0,3 pontos pela explicação correta). O HTTP (*HyperText Transfer Protocol*) e o SMTP (*Simple Mail Transfer Protocol*) são exemplos de protocolos que executam sobre o TCP (citar um vale 0,2 pontos), pois ambos necessitam de transferência confiável de dados (0,3 pontos pela explicação correta).

2. Sobre os serviços de multiplexação e demultiplexação oferecidos pela camada de transporte e sobre os *sockets* UDP (*User Datagram Protocol*) e TCP (*Transmission Control Protocol*):

- a. Quais as funções dos serviços de multiplexação e demultiplexação? (0,5 pontos).

Resposta: As funções dos serviços de multiplexação e demultiplexação são (i) identificar a qual processo pertence um segmento e adicionar um cabeçalho (0,2 pontos) e (ii) encaminhar um segmento para o processo correto com base nos dados desse cabeçalho (0,3 pontos).

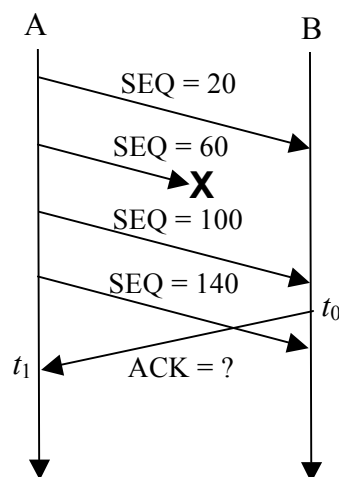
- b. Considere que um sistema final C possui um *socket* UDP com número de porta 12345. Considere ainda que outros dois sistemas finais, A e B, enviam um segmento UDP cada para C número de porta de destino 12345. Esses dois segmentos serão encaminhados para o mesmo *socket* em C? Se sim, como o processo em execução em C consegue identificar que os dois segmentos foram enviados por sistemas finais diferentes? Justifique sua resposta explicando como funciona o serviço de demultiplexação com o UDP. (1,0 ponto)

Resposta: Sim, os dois segmentos serão encaminhados para o mesmo *socket* identificado pela porta 12345 (0,2 pontos). Para cada segmento recebido pelo *socket*, o sistema operacional fornece ao processo em execução em C os endereços IP de A e B para que este processo seja capaz de determinar a origem de cada segmento (0,8 pontos).

- c. Considere agora que um sistema final C executa em um servidor Web na porta 80. Esse servidor emprega conexões persistentes e recebe simultaneamente requisições de outros dois sistemas finais, A e B. As requisições de A e B estão sendo enviadas para o servidor através do mesmo *socket* em C? Caso estejam sendo enviadas por *sockets* diferentes, estes *sockets* possuem a mesma identificação de porta, no caso 80? Justifique suas respostas explicando como funciona o serviço de demultiplexação com o TCP. (1,0 ponto)

Resposta: Não, as requisições de A e B estão sendo enviadas para o servidor através de *sockets* diferentes (0,2 ponto). Para cada conexão persistente, o servidor Web em C cria um *socket* diferente identificado pela quádrupla: endereço IP de origem, número de porta de origem, endereço IP de destino, número de porta de destino (0,2 pontos). Quando, C recebe um pacote IP, ele examina esses quatro campos contidos nos cabeçalhos da camada de transporte e rede para determinar para qual *socket* ele deve encaminhar os dados contidos no segmento (0,2 pontos). Como as requisições de A e B possuem, ao menos, endereços IP e origem diferentes, elas serão encaminhadas para *sockets* diferentes (0,2 pontos). Os *sockets* em C associados a esse servidor Web possuem o mesmo identificador de porta de destino, 80, mas as quádruplas que identificam os sockets são diferentes, uma vez que os endereços IP de origem são diferentes (0,2 pontos).

3. Dois processos A e B executando em estações diferentes se comunicam usando o TCP (*Transmission Control Protocol*), como o diagrama abaixo. Nesse diagrama, o tempo cresce de cima para baixo e as setas diagonais representam segmentos TCP enviados de A para B ou de B para A, dependendo da orientação da seta. Os números de sequência dos dados de aplicação enviados de A para B estão indicados sobre as setas. O número de sequência do primeiro byte enviado através da conexão de A para B é 20. Dos quatro segmentos enviados de A para B, o segundo segmento foi perdido pela rede e não alcançou o destino.



Com base na situação ilustrada no diagrama:

- a. Defina o número de confirmação (ACK) enviado de B para A no instante t_0 . Justifique sua resposta (1,0 ponto). Questão adaptada do ENADE 2014 - Engenharia de Computação

Resposta: ACK = 60 (0,5 pontos), porque B já recebeu 40 bytes corretamente a partir do byte 20. Isso pode ser inferido pelo número de sequência do segundo segmento enviado. Logo, o próximo byte esperado por B é o byte 60 (0,5 pontos).

Lembre-se que tanto o número de reconhecimento quanto o de sequência no TCP são orientados a bytes.

b. Qual o tamanho em bytes dos segmentos enviados de A para B? Justifique sua resposta. (0,5 pontos)

Resposta: Os três primeiros segmentos enviados de A para B possuem 40 bytes (0,2 pontos). Dado que o número de sequência no TCP é orientado a bytes, determina-se o tamanho de um segmento pela diferença entre o número de sequência do segmento seguinte e o número de sequência do segmento o qual se deseja saber o tamanho (0,3 pontos). Por exemplo, o tamanho do primeiro segmento enviado de A para B é dado por $60 - 20 = 40$ bytes. Vale ressaltar que não se pode determinar o tamanho do quarto segmento enviado de A para B com os dados do enunciado.

c. Qual a medida tomada por A logo após o instante t_1 ? Justifique sua resposta. Assuma que não existe um *buffer* em B para armazenar segmentos recebidos fora de ordem, que o valor do temporizador é muito maior do que t_1 e que A não possui mais segmentos para serem enviados para B. (1,0 pontos)

Resposta: Ao receber o ACK=60 em t_1 , A confirma a boa recepção do primeiro segmento por B, desliza a sua janela de transmissão e atualiza o contador de ACKs (0,5). Como A não possui mais segmentos para serem enviados para B, A ficará esperando pelo estouro do temporizador do segundo segmento enviado para então retransmitir esse segmento (0,5 pontos).

4. Sobre o controle de fluxo e de congestionamento do TCP:

a. Diferencie os objetivos dos dois mecanismos. (0,5 pontos)

Resposta: O objetivo do controle de fluxo é não sobrecarregar o receptor com mais dados do que ele pode receber (0,2 pontos). Por outro lado, o controle de congestionamento tem como objetivo inferir um congestionamento e não sobrecarregar ainda mais a rede nessa situação (0,3 pontos).

b. Descreva sucintamente o funcionamento do mecanismo de controle de fluxo do TCP. (1,0 ponto)

Resposta: O controle de fluxo funciona da seguinte forma. O receptor anuncia o espaço livre em seu buffer para o transmissor através do campo janela de recepção (RcvWindow) presente no cabeçalho de cada segmento enviado (0,5 pontos). Ao receber essa informação, o transmissor limita sua janela de transmissão, ou seja, a quantidade de dados ainda não reconhecidos, ao tamanho informado no campo (0,5 pontos). Dessa forma, o receptor não é afogado pelo transmissor.

c. Descreva sucintamente o funcionamento do mecanismo de controle de congestionamento do TCP. (1,0 ponto)

Resposta: O controle de congestionamento também limita a quantidade de dados ainda não reconhecidos, porém, de acordo com o nível de congestionamento da

rede. Para tanto, perdas são inferidas através da recepção de três ACKs duplicados e pelo estouro do temporizador de espera de um ACK (0,4 pontos). Nos dois casos, a medida tomada após o evento de perda é reduzir a janela de congestionamento pela metade ou para 1 segmento e redefinir o valor do `ssthresh` (0,4 pontos). Também são definidas diferentes formas de crescimento da janela em função do valor atual da janela de congestionamento: partida lenta (crescimento exponencial) e prevenção de congestionamento (crescimento linear) (0,2 pontos).