

Curso de Bacharelado em Ciência da Computação

Disciplina: Introdução à Redes de Computadores

Avaliação 03/2015-1

Prof Mauro Oliveria

Aluno (a): _____

1) Considere o EXEMPLO 01 na folha seguinte.

- a. Identifique que tipo de programa o exemplo implementa:
 Cliente Java Servidor TCP Cliente UDP Servidor UDP
 Nível de Aplicação Aplicação Usuário Transporte Rede
- b. Que serviço o EXEMPLO implementa (ou faz parte) ?
- c. Preencha os espaços laterais do programa com as expressões a seguir
Cria SOCKET para Datagrama / Aloca Memória para receber Datagrama
Cria Fluxo de Saída ligado ao SOCKET / Envia Linha ao Servidor
Cria Fluxo de Entrada / Cria SOCKET de Cliente, conexão ao Servidor
Cria Fluxo de Entrada ligado ao SOCKET / Lê linha do Servidor
Obtém endereço IP / Escreve Datagrama no SOCKET
Cria SOCKET para recepção / Aguarda contato do Cliente no SOCKET de recepção
Lê linha do SOCKET / Escreve Linha ao SOCKET / Cria SOCKET do Cliente
Traduz nome do hospedeiro do endereço IP usando DNS / Cria Datagrama
Envia Datagrama ao Servidor / Lê Datagrama do Servidor / Recebe Datagrama
Cria Fluxo de Entrada ligado ao SOCKET / Cria Fluxo de Saída ligado ao SOCKET

2) Considere o EXEMPLO 02 na folha seguinte.

- d. Identifique que tipo de programa o exemplo implementa:
 Cliente Java Servidor TCP Cliente UDP Servidor UDP
 Nível de Aplicação Aplicação Usuário Transporte Rede
- e. Que serviço o EXEMPLO implementa (ou faz parte) ?
- f. Preencha os espaços laterais do programa com as expressões acima (EX 1)

3) Responda as seguintes perguntas

- g. Discuta Multiplexação/Demux no contexto da funcionalidade Transporte/Rede
- h. Compare detalhadamente os protocolos TCP e UDP
- i. Conceitue SOCKET. Qual a relação deste conceito com a arquitetura de redes?
- j. Associe o conceito SOCKET com o handshake three-way do TCP.
- k. Quais as principais diferenças conceituais entre os 2 exemplos apresentados?

4. Desenhe as Máquina de Estados Finitos do TCP que ilustre os seguintes casos:

- l. Meio ideal e receptor com buffer infinito & com buffer finito.
- m. Mecanismo para tratamento de perda de pacote
- n. Idem para a perda do reconhecimento (protocolo do Bit Alternante)

5. Discuta detalhadamente o protocolo GO-BACK-N (GBN)

EXEMPLO 01

```
import java.io.*;
import java.net.*;

class clienteUDP {
    public static void main(String args[]) throws Exception
    {
        BufferedReader doUsuario=
            new BufferedReader(new InputStreamReader(System.in));
        DatagramSocket socketCliente = new DatagramSocket();
        InetAddress IPAddress = InetAddress.getByName("nomeHosp");

        byte[] dadosEnvio = new byte[1024];
        byte[] dadosRecebidos = new byte[1024];

        String frase = doUsuario.readLine();
        dadosEnvio = frase.getBytes();

        DatagramPacket pacoteEnviado =
            new DatagramPacket(dadosEnvio, dadosEnvio.length,
                IPAddress, 9876);
        socketCliente.send(pacoteEnviado);

        DatagramPacket pacoteRecebido =
            new DatagramPacket(dadosRecebidos, dadosRecebidos.length);
        socketCliente.receive(pacoteRecebido);

        String fraseModificada =
            new String(pacoteRecebido.getData());

        System.out.println("Do Servidor:" + fraseModificada);
        socketCliente.close();
    }
}
```

EXEMPLO 02

```
import java.io.*;
import java.net.*;

class servidorTCP {

    public static void main(String argv[]) throws Exception
    {
        String fraseCliente;
        String FraseMaiusculas;

        ServerSocket socketRecepcao = new ServerSocket(6789);

        while(true) {
            Socket socketConexao = socketRecepcao.accept();

            BufferedReader doCliente =
                new BufferedReader(new
                    InputStreamReader(socketConexao.getInputStream()));

            DataOutputStream paraCliente =
                new DataOutputStream(socketConexao.getOutputStream());

            fraseCliente= doCliente.readLine();

            fraseEmMaiusculas= fraseCliente.toUpperCase() + "\n";

            paraCliente.writeBytes(fraseEmMaiusculas);
        }
    }
}
```