



Universidade Federal do Ceará Campus de Russas

Disciplina Redes de Computadores Trabalho prático - Programação de sockets

Introdução

Este é um trabalho simples, introdutório, apenas para permitir um primeiro contato com a programação em sockets. Neste trabalho iremos desenvolver duas aplicações. Uma aplicação cliente que se conecta a um servidor e uma aplicação servidor.

Trabalho realizado em dupla ou trio

Objetivos

- Introduzir a interface de programação de sockets;
- Introduzir os conceitos de aplicação cliente e servidor.

Regras do Trabalho:

1. **Construção do Servidor e Cliente:** Cada dupla/trio deve criar tanto um servidor quanto um cliente utilizando a biblioteca de sockets do Python. O servidor deve ser capaz de lidar com múltiplas conexões de cliente de forma concorrente. (2,0 pontos)
2. **Funcionalidade Básica:** O servidor deve ser capaz de aceitar conexões de clientes e responder a solicitações deles. O cliente deve ser capaz de se conectar ao servidor, enviar mensagens e receber respostas. (2,0 pontos)
3. **Troca de Mensagens:** Cada dupla/trio deve implementar um protocolo simples de troca de mensagens entre o cliente e o servidor. Isso pode envolver mensagens de "comando" do cliente e respostas correspondentes do servidor. (1,0 pontos)
4. **Comandos Mínimos:** Um conjunto mínimo de comandos que o cliente pode enviar para o servidor. Por exemplo, o cliente pode enviar um comando de "consulta" e o servidor responde com uma mensagem contendo informações fictícias. (2,0 pontos)

Comandos Disponíveis:

CONSULTA: Solicita informações ao servidor.

HORA: Solicita a hora atual do servidor.

ARQUIVO <nome>: Solicita ao servidor o envio de um arquivo específico.

LISTAR: Solicita ao servidor a lista de arquivos disponíveis.

SAIR: Encerra a conexão com o servidor.

Respostas do Servidor:

DADOS <informações>: Resposta à consulta do cliente.

HORA_ATUAL <hora>: Resposta com a hora atual no formato HH:MM:SS.

ARQUIVO <nome> <conteúdo>: Resposta contendo o conteúdo do arquivo solicitado.

LISTA_ARQUIVOS <arquivo1> <arquivo2> ...: Resposta com a lista de arquivos disponíveis.

ARQUIVO_NAO_ENCONTRADO: Resposta quando o arquivo solicitado não é encontrado.

ADEUS: Confirma o encerramento da conexão após receber o comando SAIR.

COMANDO_DESCONHECIDO: Resposta quando o servidor não reconhece o comando enviado pelo cliente.

5. **Tratamento de Erros**: Cada dupla/trio deve incorporar tratamento de erros em suas implementações para lidar com desconexões inesperadas, erros de conexão e outros problemas que possam surgir durante a execução. (1,0 ponto)
6. **Interface Amigável**: Embora o foco principal seja a funcionalidade de rede, podem criar uma interface amigável, exibindo mensagens de status e interações para facilitar a compreensão do funcionamento do sistema. (1,0 ponto)
7. **Documentação**: Cada dupla/trio deve fornecer um documento descrevendo a implementação, explicando as decisões, os desafios encontrados e as soluções adotadas. Além disso, detalhes sobre como executar o servidor e cliente devem ser incluídos. (1,0 ponto)