

**Atividade 02 - Programação com Sockets C (UDP)**

Wesley Miguel Bizerra de Mello

**Orientação:**

Prof. Dr. Rodrigo Mitsuo Kishi

Três Lagoas - MS

2021

# RELATÓRIO

## Servidor

Na implementação do algoritmo do servidor foram usadas 6 bibliotecas e dessas 6 a mais relevante para citar é a **arpa/inet.h**, essa biblioteca nos permite usar sockets no nosso algoritmo, não me aprofundei muito sobre todas suas funcionalidades e funções, apenas aprofundei o suficiente para conseguir usar as funções ligadas aos sockets. Mais um ponto a se destacar é o fato dessa biblioteca ser, pelo que eu entendi pesquisando, nativa para o Linux, o que complicou um pouco a criação desse algoritmo visto que tive que pensar em uma forma de emular um terminal Linux para testar se o código estava funcionando ou não. Ainda no servidor, foi criada uma função para salvar os dados do arquivo do cliente no servidor, dentro dela é usado a função **recvfrom()** para receber os pacotes UDP vindos do cliente e também é usado a função **fprintf()** para guardar esses dados no arquivo servidor.txt.

Dentro da **main()** nós temos a parte de definição de valores de IP e porta, a criação do socket UDP, a definição do endereço ip e porta UDP que serão utilizadas, e a associação desse endereço com o socket criado.

## Cliente

No algoritmo do cliente as mesmas 6 bibliotecas foram utilizadas. Além disso temos uma função para enviar os dados do arquivo do cliente para o servidor.

Na **main()** nós também temos a parte de definição de valores de IP e porta, a criação do socket UDP, a definição do endereço ip e porta UDP (do servidor) que serão utilizadas, e a associação desse endereço com o socket. Além disso, temos a criação de um ponteiro FILE para que possamos ler os dados do arquivo cliente.txt e enviar para o servidor.

# COMO USAR A APLICAÇÃO

A aplicação é simples, no arquivo **cliente.txt** digite o que deve ser enviado para o servidor. Após isso inicie o servidor e depois inicie o cliente. Os dados que estiverem no arquivo **cliente.txt** serão enviados para o servidor e serão salvos no arquivo **servidor.txt**.