



# UFRJ

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO  
INSTITUTO DE COMPUTAÇÃO

PEDRO RENATO MELLO  
DRE:115060367

TELEPROCESSAMENTO E REDES  
RIO DE JANEIRO, OUTUBRO DE 2022  
Professor: Daniel Sadoc

# Relatório

1.

Para o resultado desejado só foi necessário adicionar um loop que engloba os comandos desde a digitação da mensagem até o recebimento da mesma pelo servidor. Como o servidor está ouvindo todas as requisições em loop, não foi necessário fazer nenhuma alteração no mesmo.

2.

A vantagem na execução persistente é a velocidade em que as threads conseguem fazer suas várias chamadas para o servidor. Apesar de não ter conseguido rodar como eu queria, das poucas vezes que a função persistente não deu timeout, deu para notar a sua velocidade comparado com as threads não persistentes.

3.

Para essa questão foram utilizados 10 threads, sendo 5 TCP e 5 UDP e 100 Clientes, sendo 50 TCP (não persistente) e 50 UDP. Ao realizar a execução notamos uma predominância muito grande das chamadas UDP, o que faz sentido já que elas não precisam realizar o 3-way handshake antes de enviarem o único pacote.

5.

A aplicação foi feita utilizando o protocolo UDP.

6.

A questão foi feita, porém o python não conseguiu entender que a variável state é global. E aí dentro da função nextState ele está reclamando que a variável é referenciada antes de ser definida. Não consegui resolver esse erro. Mas a lógica está toda pronta tanto do servidor como do client.