



Universidade de Brasília - UnB
Faculdade do Gama - FGA
Fundamentos de Redes de Computadores
Prof. Fernando William Cruz
Giovanna Borges Bottino Matr:17/0011267

Relatório Individual
Projeto de pesquisa - Parte 1

Dupla: Felipe Boccardi Silva Agustini Matr: 180119818

Objetivo do trabalho:

O objetivo desse trabalho é a implementação de um pseudo-protocolo de enlace de dados contendo diálogo ponto-a-ponto, stop-and-wait e orientado à conexão em modo simplex.

Diário de atividades:

Antes de começar a desenvolver a dupla se reunião para definir como iríamos contribuir para o projeto. Utilizamos C como definido pelo professor, o Git para o versionamento do código e rastreabilidade de interações.

O git pode ser encontrado no seguinte link: <https://github.com/giovannabbottino/frc>

Nossas contribuições e reuniões podem ser observadas pela contribuição no git em <<https://github.com/giovannabbottino/frc/commits/main>>. Para avaliar nosso projeto recomendo uma leitura do nosso *server.c* e do *client.c* que tentamos deixar comentado para o entedimento. Além disso, também recomendo seguir o README enviado. Se não pretender usar o Makefile não esqueça de incluir o *-I./inc*. De forma que fique da seguinte maneira:

```
gcc src/server.c -o bin/server -I./inc
```

```
gcc src/client.c -o bin/client -I./inc
```

Um de nossos maiores problemas foi a interpretação do texto, diferíamos bastante sobre o que acreditávamos ser o requisitado pelo professor. Mas após decidir qual interpretação seguir avançamos na programação.

Decidimos utilizar de TCP não UDP, isso também foi um problema, já que não tínhamos conhecimento e experiências o suficiente. A internet foi um grande auxiliar nessa questão.

Ainda há algumas melhorias a serem feitas no nosso trabalho como por exemplo tornar duplex. Para fazer isso iremos fazer como no exemplo do professor, mas ainda há pesquisas a serem feitas.

Para fazer a simplex usamos as seguintes bibliotecas:

#include <netinet/in.h> para a struct sockaddr_in

#include <sys/socket.h> para as funções connect() bind() send() recv()

#include <arpa/inet.h> para as funções htons(), inet_addr()

#include <sys/types.h> para os tipos AF_INET, SOCK_STREAM

Em nosso projeto tentamos adicionar alguns tratamentos de erros, como na entrada pelo argumento da PDU, se não for inserido ou não for um número o client não continua. Nesse caso, o server continua, um problema a ser arrumado ou pensado para a próxima entrega.

Outro caso é quando há mais de duas vezes erro na resposta da server, nesse momento a tentativa de envio da mensagem é terminado, para não entrar em loop infinito. Para testar basta enviar a mensagem “de erro”.

Observação Pessoal:

Acredito que esse trabalho foi bastante desafiador. Ele traz um novo ponto de vista, mais prático de redes. Penso que para a próxima entrega eu não tenha mais tantas dúvidas de interpretação de texto.

