

# Trabalho de Redes 1 - ERE2

Diogo Paris Kraut - GRR20166365

Departamento de Informática – Universidade Federal do Paraná (UFPR)

`dpk16@inf.ufpr.br`

***Resumo.** Relatório inclui manual das funções utilizadas na implementação do trabalho e a descrição das escolhas feitas durante o desenvolvimento do mesmo.*

## 1. Detalhes de Implementação

A seguir são relatadas alguns dos detalhes da implementação do trabalho.

### 1.1 Mensagens

Mensagens são armazenadas em uma estrutura com campos de tamanho fixo, incluindo o campo de dados. Ambos cliente e servidor mandam e recebem essa estrutura, o que significa que sempre escaneiam o soquete por uma mensagem com o tamanho desta.

### 1.2 Comandos

Os comandos são chamados conforme determinado na especificação do trabalho. Nomes de arquivos passados por um comando (argumento) para o servidor não podem passar de 15 caracteres.

Para o comando “edit”: se a linha especificada não existe nada acontece ao arquivo, mas também não é retornado um erro.

Cada arquivo retornado por “ls” é terminado por ‘\n’. Caso o tamanho do nome seja múltiplo de 15 ‘\n’ é mandado em uma mensagem separada. O limite de tamanho para um nome de arquivo ser retornado é o próprio limite definido pelo sistema operacional.

## 2. Manual

Manual descrevendo as funções mais complexas do projeto.

### 2.1 FUNÇÃO `sendPacket`

```
int sendPacket(int socket, tMessage *mS, tMessage *mR, int TYPE);
```

Manda a mensagem **mS** e espera a resposta. A resposta é então copiada para **mR**. Se a resposta for do tipo NACK, ou se não receber a resposta até o TIMEOUT **mS** é reenviada.

### RETORNO

Retorna 1 se a resposta tiver tipo **TYPE** e 0 se tiver tipo ERR.

### 2.2 FUNÇÃO `runCommand`

```
void runCommand(int socket, tMessage *mS, tMessage *mR, char *arg, int  
CMD_TYPE, int R_TYPE, int src, int dest, char *s);
```

Manda um comando do tipo **CMD TYPE**, com o campo de dados preenchido com **arg**. Processa (mostra na tela) todas as respostas com tipo **R TYPE** até receber uma mensagem de EOTX ou ERR. Se **s** não for NULL, concatena o conteúdo das mensagem em **s** ao invés de mostrar na tela.