**Faculdade de Engenharia da Universidade do Porto**

C:\Users\-DIOGO\Downloads\Cores oficiais\logótipo com cores oficiais.tif

**Rede de Computadores**

**Trabalho 1**

**27/04/2015**

**Turma:** *3MIEIC5*

**Autores:**

Ana Catarina Amaral [@fe.up.pt](mailto:up201306082@fe.up.pt)

Pedro Miguel Vieira da Câmara [up201304073@fe.up.pt](mailto:up201304073@fe.up.pt)

Pedro Oliveira da Silva [up201306095@fe.up.pt](mailto:up201306095@fe.up.pt)

Índice

[Introdução 3](#_Toc434015549)

[Arquitetura 4](#_Toc434015550)

[Estrutura do código 4](#_Toc434015551)

[Casos de uso principais 4](#_Toc434015552)

[Protocolo de ligação lógica 4](#_Toc434015553)

[Protocolo de aplicação 4](#_Toc434015554)

[Validação 4](#_Toc434015555)

[Elementos de valorização 5](#_Toc434015556)

[Conclusão 5](#_Toc434015557)

[Bibliografia 6](#_Toc434015558)

# Sumário

O seguinte relatório serve para apresentar a implementação de um protocolo, estudado nas aulas teóricas, que permite a transmissão de ficheiros entre computadores, com o uso de um cabo de série.

Quanto à implementação em si, o código está dividido em 2 camadas, uma que representa a aplicação para comunicar com o utilizador e uma outra, mais baixo nível, a camada de link que trata do funcionamento interno e implementação do protocolo em si.

# Introdução

Como descrito no Sumário, este trabalho serviu para pôr em prática a matéria estudada nas aulas teóricas sobre o mecanismo de comunicação entre computadores com o uso de uma porta série. Este mecanismo está secionado em várias camadas de modo a ter uma estrutura organizada e de fácil interpretação. Para uma melhor clarificação, este relatório irá servir para explicar com melhor clareza todo este funcionamento.

Em seguida, irá ser apresentado um conjunto de tópicos que permite explicar como está organizado o código, desde a comunicação com o utilizador ao próprio envio dos pacotes de informação.

# Arquitetura

Como descrito anteriormente, todo o código foi divido em camadas.

A cama da aplicação é responsável ligação entre o utilizador e o programa em si. Tem como principal função obter as informações necessárias para a configuração do protocolo, como por exemplo, o número de tentativas no caso de uma falha de leitura ou até mesmo o tempo para ativar o alarme. Para além disso, contém as funções mais alto nível para o envio e receção de pacotes de dados relativo ao ficheiro a enviar e pacotes de controlo para avisar o início e fim do envio.

A outra camada, *linkLayer*, trata de todo o funcionamento mais baixo nível do programa. Após o envio de um pacote cabe a esta processá-lo e determinar que tipo de informação contém, ou seja, se é dados ou controlo. Para qualquer um dos casos é sempre verificado na função llread(), se a informação enviada contém erros, se é informação repetida ou se simplesmente falta informação.

Foi criada em cada camada algumas estruturas para melhor gerir os dados. Na primeira era essencial guardar o descritor da porta série usada e em que modo foi acedida, emissor ou recetor.

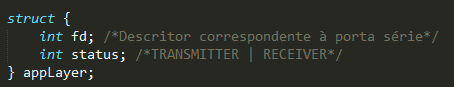


Figura : Estrutura que representa a camada da aplicação.

Para além desta estrutura foram guardadas diversas constantes, desde *flags* usadas a valores guardados por defeito no caso de o utilizador não os inserir.

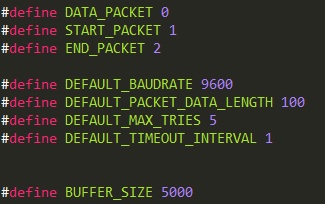


Figura : Constantes usadas na cama de aplicação.

No caso da segunda camada, foi criada uma estrutura de dados para gerir a estatística, as configurações definidas e uma última para representar a informação enviada em cada trama.

# Estrutura do código

# Casos de uso principais

# Protocolo de ligação lógica

# Protocolo de aplicação

# Validação

# Elementos de valorização

# Conclusão

O presente relatório elaborado em dispor da Unidade Curricular de Concepção e Análise de Algoritmos da Faculdade de Engenharia da Universidade do Porto reflete o trabalho realizado pelo grupo para a formalização de um problema aplicado às exigências bem como a sua correta formulação e aplicação numa situação real, tendo em conta as suas limitações.

Destacou-se como maior dificuldade a compreensão do enunciado, registando a complexidade da transformação da ideia no projeto a desenvolver.

Considera-se o trabalho exequível e corretamente formulado para a continuação do projeto e respetiva próxima iteração.