

# Sistema de Envio de Mensagens Utilizando Rede em Anel

## CI058 - Trabalho II

Cauê Rodrigues  
Universidade Federal do Paraná  
Curitiba, Brasil  
cogr13@inf.ufpr.br

Arianne Bortolan  
Universidade Federal do Paraná  
Curitiba, Brasil  
apb14@inf.ufpr.br

### 1. INTRODUÇÃO

Relatório referente ao segundo trabalho da matéria de Redes (CI058) da Universidade Federal do Paraná, o mesmo tem como objetivo transmitir os objetivos do projeto, os desafios encontrados e as decisões tomadas a respeito da atividade fornecida pelo professor Luiz Carlos Pessoa Albini.

### 2. PROJETO

#### 2.1 Implementação

O projeto foi implementado em C e visava construir uma rede em anel entre 4 servidores afim de tornar possível o envio de mensagens entre eles utilizando passagem de bastão temporizada, fila de prioridades e *timeout*.

#### 2.2 Desafios

Em primeira instância, o maior desafio foi lidar com códigos e bibliotecas nunca antes usados pela dupla, utilizamos os códigos expostos na página do professor Elias P. Duarte Jr. para auxiliar na criação dos métodos de envio e recepção de mensagens.

Obtivemos certa dificuldade para efetuar a confirmação do envio de uma mensagem, fazendo com que o *"ack"* se perdesse, gerando *timeout*.

Não logramos êxito na criação da temporização do bastão e no envio do mesmo, embora utilizando uma abordagem simples não conseguimos localizar o problema.

Além dos problemas acima, também enfrentamos certas dificuldades ao criar as prioridades das mensagens.

#### 2.3 Decisões

Como já citado, utilizamos os códigos "cliente" e "servidor" do professor Elias para construir o messageiro, transformamos ambos os programas em funções e englobamos eles em um programa maior que, estando com o bastão ativava a função cliente e não estando com o bastão ficava aguardando uma mensagem como servidor.

O bastão foi criado como uma mensagem qualquer, para isso, na estrutura havia um campo destinado a verificar se a mensagem em questão seria uma mensagem comum ou o bastão, que decidia em qual "modo" deveria entrar, não conseguimos porém resolver a temporização do bastão e a sincronização das mensagens.

Utilizamos o *timeout* da biblioteca "sys/socket.h" por facilidade e simplicidade, bastando apenas referenciar as funções, o tempo escolhido foi de 20 segundos e o número de tentativas é 10 (dez), mostrando após isso uma mensagem de falha de conexão ao usuário.

Para seleção de envio e recebimento de mensagens foi criado uma definição com os nomes dos servidores do departamento, relacionando-os aos respectivos ips, foi utilizada um porta padrão definida no código para o uso do messageiro (7171).

### 3. CONCLUSÃO

Conseguimos enviar as mensagens entre os servidores com sucesso, no entanto enfrentamos maiores problemas com o envio do bastão, que gerava falha de segmentação, embora o corpo da mensagem utilizasse a mesma estrutura das demais mensagens, seu comportamento era adverso às demais. Não foi possível localizar o problema em tempo hábil.

A aplicação do *timeout* funcionou perfeitamente, inclusive auxiliando durante os testes do programa, principalmente nas falhas de confirmação de recebimento das mensagens.

### 4. TUTORIA

Professor Luiz Carlos Pessoa Albini, doutor em ciência da computação pela Universidade de Pisa, atuando como professor titular na Universidade Federal do Paraná.