

**Redes de Computadores - Documentação TP0**  
**Aluno: Marcelo Costa Monteiro**  
**Universidade Federal de Minas Gerais**

Ambos os programas cliente-aluno.c e cliente-prof.c, iniciam a conexão com o servidor.c da mesma maneira, seguindo o fluxo: socket, connect, recebem a mensagem READY e enviam mensagem contendo a senha. Apenas após o envio da senha, temos as diferenças semânticas entre estes programas.

Em particular, o cliente-aluno.c, aguarda o recebimento de duas mensagens ("OK", "MATRICULA") e depois envia uma mensagem contendo a matrícula. O cliente-prof.c, por sua vez, apenas recebe a lista. Ambos encerram após receber uma última mensagem "OK" do servidor.

Uma estratégia adotada para deixar o código mais legível foi a utilização das funções sendMsg e recvMsg que executam o processo necessário para enviar e receber mensagens de forma padronizada, além de imprimi-las na tela, facilitando a depuração. No entanto, foram implementadas funções específicas para tratar mensagens específicas no caso do cliente-aluno e do cliente-prof: sendMatricula e recvLista, respectivamente.

Para tratar múltiplas conexões pelo programa servidor, este cria uma nova thread para cada conexão aceita. Desta forma, para continuar funcionando em caso de erro de comunicação, basta encerrar o cliente e a thread que apresentou o erro.

Houveram algumas dificuldades foram encontradas ao longo do desenvolvimento do trabalho, principalmente no que envolvia gerenciar arrays de caracteres e strings (que incluem o caractere '\0'). Essa e outras dificuldades foram contornadas. No entanto, a conversão da matrícula de string para bytes não foi implementada. Portanto a mesma está sendo enviada em formato string a fim de tornar a aplicação funcional.