

Departamento de Ciência da Computação - UDESC
Disciplina: Redes de Computadores – TADS
Professora: Janine Kniess, 2017/1

Sockets em C (Utilizar o Sistema Operacional Linux)

1- Compilar os programas (cliente e servidor) em C.

1.1. No Linux, abra o aplicativo “Terminal” e compile os programas em C

```
gcc -o servidor Servidor.c  
gcc -o client Client.c
```

2- Executar os programas (cliente e servidor) em C

2.1. No Linux, abra uma segunda sessão do aplicativo “Terminal” (mantenha a primeira aberta).

2.2. Em cada janela separadamente execute os programas. Primeiro deverá ser executado o servidor.

```
./servidor
```

2.4. Na outra sessão do terminal aberta execute o cliente.

```
./client
```

No termina será apresentado as seguintes informações:

```
Socket created  
Connected
```

```
Enter message : Sockets
```

```
Server reply :
```

```
Sockets
```

```
Enter message : Teste2
```

Observação: Pode compilar e executar o cliente com apenas um comando:

```
gcc Client.c && ./a.out
```

Exercícios:

1) Altere os programas, cliente ou/e servidor, sendo que, tanto o cliente quanto o servidor devem lidar com uma mensagem formada por um conjunto de palavras (e.g., Aula REC Sockets). No momento, o programa somente aceita uma palavra como mensagem.

2) Altere os programas cliente ou/e servidor, sendo que, o cliente envia uma mensagem composta por um número maior ou igual a dois de palavras. O servidor deve traduzir os caracteres para seu correspondente em ASCII e entregar para o cliente o somatório dos decimais em ASCII. Por exemplo, com a mensagem: Oi Ana, o servidor deve apresentar o valor $79(O) + 105(i) + 65(A) + 110(n) + 97(a) = 456$.

3) Os programas cliente e servidor foram implementados com Socket TCP ou UDP?

4) Suponha que o cliente seja executado primeiro que o servidor. O que acontece? Por quê?

5) Qual a função do comando “server.sin_family” nos programas cliente e servidor.

6) No programa servidor, qual a função da linha de comando: server.sin_addr.s_addr = INADDR_ANY;