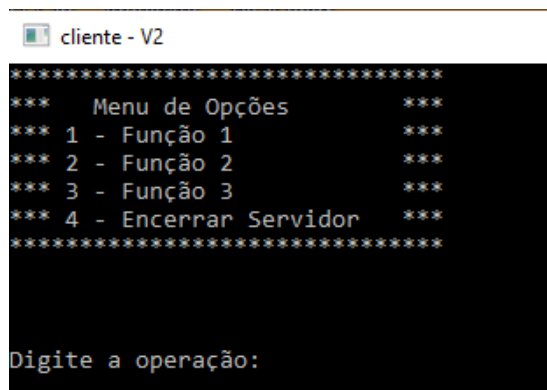
 PUC Minas	PROGRAMAÇÃO COM SOCKETS		
	Curso: Jogos – Praça da Liberdade		
	Turno: T	Período:5	Data: 12/03/2020
	LABORATÓRIO: 1	Valor: 5	Nota:
	Disciplina: Jogos em		
	Professor: JÚLIO CONWAY		

APLICAÇÃO CLIENTE-SERVIDOR

Neste trabalho deverá ser desenvolvido em **C#** uma aplicação **CLIENTE-SERVIDOR**, em modo Console, para transferência de comandos do Servidor para o Cliente. Assim, o Cliente envia comandos para o Servidor, que ecoa estes comandos. Futuramente estes comandos poderão ser ações a serem tomadas pelo Servidor, como por exemplo tratar comandos de um Jogo, ou enviar o comando para um outro jogador. Está disponibilizado no SGA um código básico de Cliente-Servidor, que poderá servir de base para o trabalho.

A aplicação Cliente deverá ter um Menu de Opções, conforme a Figura 1.



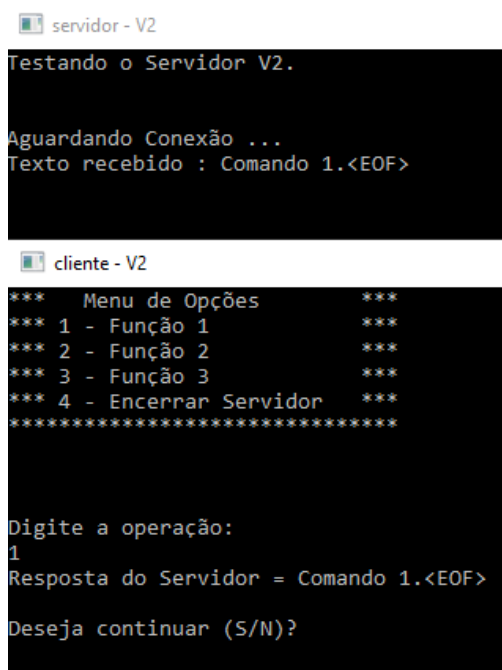
```

cliente - V2
*****
***  Menu de Opções      ***
***  1 - Função 1        ***
***  2 - Função 2        ***
***  3 - Função 3        ***
***  4 - Encerrar Servidor ***
*****

Digite a operação:
  
```

Figura 1

Quando o Servidor receber um comando, ele deverá ecoar o comando recebido, conforme a Figura 2.



```

servidor - V2
Testando o Servidor V2.

Aguardando Conexão ...
Texto recebido : Comando 1.<EOF>

cliente - V2
***  Menu de Opções      ***
***  1 - Função 1        ***
***  2 - Função 2        ***
***  3 - Função 3        ***
***  4 - Encerrar Servidor ***
*****

Digite a operação:
1
Resposta do Servidor = Comando 1.<EOF>

Deseja continuar (S/N)?
  
```

Figura 2

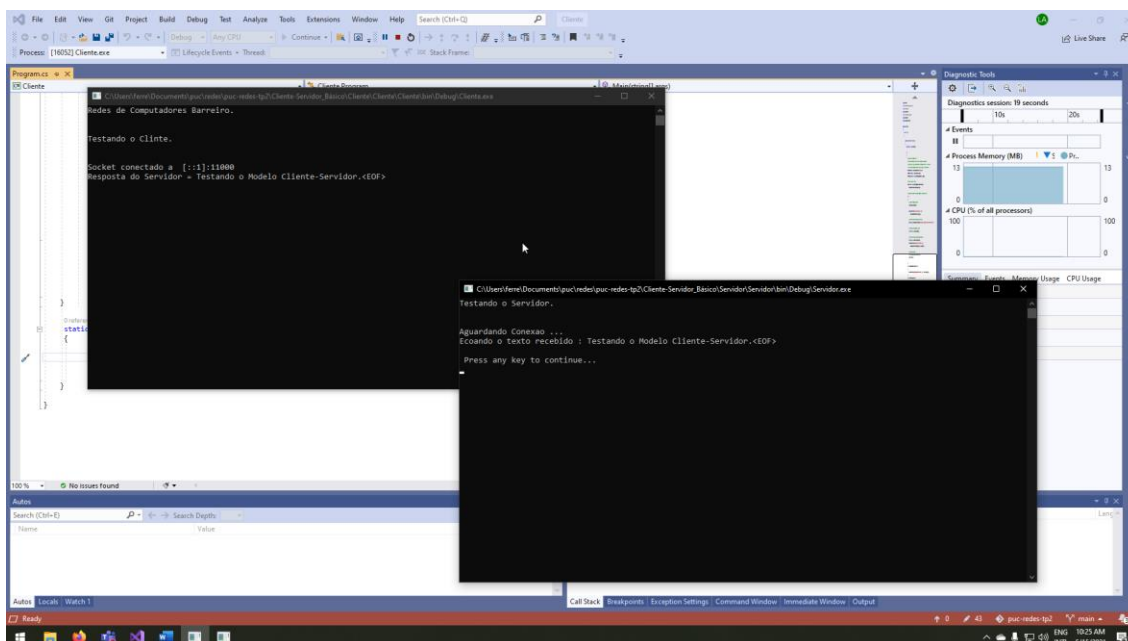
Como pode ser visto na Figura 2, o usuário no lado Cliente escolhe uma opção e envia esta opção ao Servidor. Este recebe a opção, ecoa de volta para o Cliente, que a exibe. A aplicação Cliente deve possuir então um Menu de Opções conforme a Figura 2. Quando o usuário escolhe a opção 4, ambos os Cliente e Servidor devem ser encerrados.

A aplicação Cliente deve ficar em loop mostrando a mensagem “Deseja continuar (S/N)?”. A opção “N” também encerra ambos Cliente e Servidor.

Obs: As atividades abaixo deverão ser feitas em grupos de dois alunos ou individualmente. Deverá ser postado no SGA, um arquivo para o grupo. Os relatórios deverão ser postados em PDF e deverão conter prints de todas as atividades pedidas.

Atividade 1: Executar os códigos Cliente e Servidor disponibilizados no SGA, comprovando o funcionamento do código.

Primeiramente, o código foi testado com a execução do template oferecido. Primeiro o servidor foi inicializado e uma mensagem de “Aguardando conexão foi exibida”. Em seguida o cliente foi executado. Com isso, conseguimos conformar a conexão entre os serviços conforme a imagem abaixo onde uma mensagem do cliente é enviada ao servidor e uma resposta é enviada de volta ao cliente.



```
Testando o Servidor.  
  
Aguardando Conexao ...  
Ecoando o texto recebido : Testando o Modelo Cliente-Servidor.<EOF>  
  
Press any key to continue...
```

```
Redes de Computadores Barreiro.  
  
Testando o Clinte.  
  
Socket conectado a [::1]:11000  
Resposta do Servidor = Testando o Modelo Cliente-Servidor.<EOF>
```

Atividade 2: Modificar o código disponibilizado para atender as especificações conforme descrito acima e mostrado na Figura 2.

O código foi adaptado de acordo com a especificação. Detalhes do código em si podem ser encontrado no repositório <https://github.com/ferreiraalves/puc-redes-tp2>. Abaixo vamos demonstrar as mensagens no cliente e servidor em dois casos. A primeira é a chamada de um dos comandos pelo cliente:

```
C:\Users\ferre\Documents\puc\redes\puc-redes-tp2\Cliente-Servidor_Básico\Cliente\Cliente\bin\Debug\Cliente.exe
Redes de Computadores Barreiro.

Testando o Cliente.

Socket conectado a [::1]:11000
Menu de opcoes:
1 - Funcao 1
2 - Funcao 2
3 - Funcao 3
4 - Encerrar Servidor

Escolha a opção:
1
Resposta do Servidor = Comando 1.<EOF>
Deseja Continuar? (S/N)

C:\Users\ferre\Documents\puc\redes\puc-redes-tp2\Cliente-Servidor_Básico\Servidor\Servidor\bin\Debug\Servidor.exe
Testando o Servidor V2.

Aguardando Conexao ...
Texto recebido : Comando 1.<EOF>
_
```

Neste caso a opção 1 é escolhida. A mensagem é enviada para o servidor, lida e retornada para o cliente. Para validar que os retornos estão corretos, segue um exemplo de execução ao selecionarmos a alternativa 3:

```
Menu de opcoes:  
1 - Funcao 1  
2 - Funcao 2  
3 - Funcao 3  
4 - Encerrar Servidor  
  
Escolha a opcao:  
3  
Resposta do Servidor = Comando 3.<EOF>  
Deseja Continuar? (S/N)
```

O caso 4 é um pouco destino. Nele, o servidor tem uma resposta de acordo com a mensagem recebida. O exemplo de execução segue abaixo:

```
Redes de Computadores Barreiro.  
  
Testando o Ciente.  
  
Socket conectado a [::1]:11000  
Menu de opcoes:  
1 - Funcao 1  
2 - Funcao 2  
3 - Funcao 3  
4 - Encerrar Servidor  
  
Escolha a opcao:  
4  
Resposta do Servidor = Encerra Servidor<EOF>  
Servidor Desligado  
Desligando cliente...  
client shutdown connection  
  
C:\Users\ferre\Documents\puc\redes\puc-redes-tp2\Cliente-Servidor_Básico\Servidor\Servidor\bin\Debug\Servidor.exe  
Testando o Servidor V2.  
  
Aguardando Conexao ...  
Texto recebido : Encerra Servidor<EOF>  
Encerrando o Servidor...  
  
Press any key to continue...
```

Ou seja, o servidor recebe a mensagem, dá o retorno e encerra em seguida. O cliente, por sua vez, recebe a mensagem de encerramento e também é desligado. Com este caso validado, temos todos os requisitos da proposta atendidos.