



Visual C# .NET

Nível I - Básico

Autor-Compilador: Elzevir Jr.
Janeiro, 2007

SUMÁRIO

EXERCÍCIOS DO C#.NET	5
EXERCÍCIOS 01 – ENTRADA E SAÍDA DE DADOS E DECLARAÇÃO DE VARIÁVEIS	7
01.01 – <i>Volume de uma caixa retangular</i>	7
01.02 – <i>Converter temperatura em graus Fahrenheit para Celsius</i>	8
01.03 – <i>Calcula volume de um cilindro</i>	8
01.04 – <i>Curso de criação de coelhos</i>	10
01.05 – <i>Calcula consumo de combustível</i>	10
01.06 – <i>Calcula o quadrado da diferença entre números</i>	11
01.07 – <i>Propriedade distributiva entre 4 números</i>	12
01.08 – <i>Calcula o volume de uma esfera</i>	14
01.09 – <i>Conversão de temperaturas</i>	15
EXERCÍCIOS 02 – ENTRADAS E SAÍDAS DE DADOS E DECLARAÇÃO DE VARIÁVEIS	17
02.01 – <i>Calcula o produto e a adição de frações</i>	17
02.02 – <i>Calcula o salário de um vendedor</i>	18
02.03 – <i>Calcula o volume de uma lata de óleo</i>	19
02.04 – <i>Média harmônica</i>	20
02.05 – <i>Média ponderada</i>	21
02.06 – <i>Lê números e calcula operações</i>	22
02.07 – <i>Calcula potência de número</i>	23
02.08 – <i>Quadrado de um número</i>	24
02.09 – <i>Área de um quadrado</i>	25
EXERCÍCIOS 03 – ENTRADA E SAÍDA DE DADOS E DECLARAÇÃO DE VARIÁVEIS	27
03.01 – <i>Consumo de combustível</i>	27
03.02 – <i>Troca de valores entre números</i>	28
03.03 – <i>Entrada e saída de dados</i>	29
03.04 – <i>Conversão de temperaturas</i>	30
03.05 – <i>Prestação em atraso</i>	30
03.06 – <i>Soma do quadrado de números</i>	31
03.07 – <i>Salário de funcionário</i>	32
03.08 – <i>Salário mensal</i>	33
03.09 – <i>Velocidade de veículo</i>	35
EXERCÍCIOS 04 – ESTRUTURA DE SELEÇÃO IF E IF / ELSE	36
04.01 – <i>Verifica número informado</i>	36
04.02 – <i>Verificar o maior e o menor número dados</i>	37
04.03 – <i>Verificar se medidas informadas é um triângulo</i>	39
04.04 – <i>Ler dois números e verificar a diferença</i>	40
04.05 – <i>Aceita números e verifica seus divisores</i>	41
04.06 – <i>Lê dois número e verifica suas diferenças</i>	43
04.07 – <i>Calcula média de aluno</i>	44
04.08 – <i>Lê números e compara valores</i>	45
04.09 – <i>Verifica se aluno foi aprovado</i>	47
EXERCÍCIOS 05 – ESTRUTURA DE SELEÇÃO IF E IF / ELSE	51
05.01 – <i>Soma números e compara resultado</i>	51
05.02 – <i>Reajusta salário</i>	52
05.03 – <i>Lê número e o converte de + para -</i>	54
05.04 – <i>Acha intervalo de número</i>	55
05.05 – <i>Soma funções</i>	57
05.06 – <i>Recebe números e verifica divisores</i>	58
05.07 – <i>Verifica lados de um triângulo</i>	60
EXERCÍCIOS 06 – ESTRUTURA DE SELEÇÃO IF E IF / ELSE	63
06.01 – <i>Verifica nome e sexo de pessoa</i>	63
06.02 – <i>Compara soma de números</i>	64
06.03 – <i>Extraí raízes quadradas</i>	65
06.04 – <i>Compara valores</i>	66
06.05 – <i>Calcula Imposto de Renda</i>	69
06.06 – <i>Reajusta salário</i>	71
06.07 – <i>Verifica número</i>	72
EXERCÍCIOS 07 – ESTRUTURA DE SELEÇÃO WHILE E OUTRAS	74
07.01 – <i>Calcula tabuada de número</i>	74
07.02 – <i>Lê número e pede novos valores</i>	74
07.03 – <i>Quadrado entre um intervalo</i>	76
07.04 – <i>Determina idade e percentuais em pesquisa</i>	76
07.05 – <i>Lê valores e verifica-os</i>	79

07.06 – Gera números e verifica divisores.....	80
07.07 – Termos de PA.....	81
07.08 – Calcula média Geral de alunos.....	82
EXERCÍCIOS 08 – ESTRUTURA DE REPETIÇÃO WHILE E OUTRAS	84
08.01 – Apresenta valores em uma faixa de números	84
08.02 – Calcula média de idades	84
08.03 – Calcula população de um país	85
08.04 – Média de uma turma de alunos.....	86
08.05 – Soma dos primeiros números inteiros	88
08.06 – Resultado de pesquisa.....	88
08.07 – Potencias de um intervalo.....	91
08.08 – Lê números em um intervalo.....	92
EXERCÍCIOS 09 – ESTRUTURA DE REPETIÇÃO IF, ELSE, WHILE E OUTRAS	94
09.01 – Utilizando operadores lógicos	94
09.02 – Utilizando while, Match.Pow e cast in.....	94
09.03 – Estruturas de seleção simples e repetição while.....	96
09.04 – Estrutura de repetição while e variáveis.....	97
09.05 – Contadores e incremento e while	98
09.06 – Calcula média de notas de aluno	99
09.07 – Recebe números com e extrai cada dígito.....	100
09.08 – Aninhamento de estruturas de seleção	101
09.09 – Soma dez primeiros números.....	102
EXERCÍCIOS 10 – ESTRUTURA DE REPETIÇÃO WHILE E OUTRAS	105
10.01 – Calcula tabuada de um número	105
10.02 – Lê valores e pede novos valores	106
10.03 – Seleção composta encadeada e aninhada.....	107
10.04 – Lê valores e conta os números negativos	110
10.05 – Calcula média de idade	111
10.06 – Calcula tempo.....	112
10.07 – Gera números divisíveis	114
10.08 – Média geral de alunos.....	115
10.09 – Calcula média de turma de alunos.....	116
EXERCÍCIOS 11 – ESTRUTURA DE REPETIÇÃO WHILE, FOR E OUTRAS	119
11.01 – Calcula soma de números inteiros.....	119
11.02 – Calcula termos de uma série	120
11.03 – Calcula alunos reprovados e aprovados	121
11.04 – Resultado de uma pesquisa de aceitação	123
11.05 – Calcula média de uma turma de alunos	125
11.06 – Lê numero e soma entre uma série	127
11.07 – Calcula uma função.....	128
11.08 – Calcula termos de PA	129
11.09 – Calcula temperatura em graus diversos e com variações.....	131
EXERCÍCIOS 12 – ESTRUTURA DE REPETIÇÃO WHILE, FOR E OUTRAS	133
12.01 – Calcula a soma de termos de uma série.....	133
12.02 – Potencia de 3 com variações.....	134
12.03 – Pesquisa de satisfação	135
12.04 – Soma de numeros de um intervalo	138
12.05 – Média de uma turma	139
12.06 – Juros compostos com MessageBox	140
12.07 – Reajuste de salário.....	141
12.08 – Repetição controlada por contador	143
12.09 – Calcula bonus de uma faixa salarial	144
12.10 – Lê número de faz multiplicações	147
EXERCÍCIOS 13 – ESTRUTURA DE REPETIÇÃO FOR, WHILE E OUTRAS	149
13.01 – Faz pesquisa de preços por região.....	149
13.02 – Reajuste salarial de funcionário	151
13.03 – Quadrado de números.....	153
13.04 – Peso ideal de uma pessoa com base em dados pessoais.....	156
13.05 – Recebe informações de produtos: Preço, região, etc.....	158
13.06 – Verifica notas informadas	161
13.07 – Valor máximo de gastos de clientes	163
13.08 – Verifica dados com base em classificação	166
13.09 – Simples calculadora com MessageBox.....	171
13.10 – Comparação entre números	173
EXERCÍCIOS 14 – ESTRUTURA DE REPETIÇÃO DO / WHILE E OUTRAS	177
14.01 – Calcula uma série	177

14.02 – Reajuste salarial de acordo com critérios	178
14.03 – Calcula número “perfeito”	181
14.04 – Lê notas escolares e efetua cálculos	184
14.05 – Calcula uma expressão	185
14.06 – Calcula uma série numérica	187
14.07 – Volume de uma esfera.....	188
14.08 – Imprime série de 1 a 10.....	190
14.09 – Calcula média de um aluno.....	191
14.10 – Calcula uma expressão	195
EXERCÍCIOS 15 – ESTRUTURA DE REPETIÇÃO DO /WHILE E OUTRAS	198
15.01 – Soma de termos de uma série	198
15.02 – Pesquisa de Mercado	200
15.03 – Bônus salarial	203
15.04 – Menor número.....	207
15.05 – Maior número dentre assinantes	208
15.06 – Gera uma série Fibonacci.....	210
15.07 - Fatorial de um número	212
15.08 – Equação do 2º grau.....	214
15.09 – Conversão de temperaturas	215
15.10 – Termos de uma série.....	217
EXERCÍCIOS 16 - ESTRUTURA DE SELEÇÃO MÚLTIPLA SWITCH E OUTRAS	220
16.01 – Lê idade e classifica categoria.....	220
16.02 – Calcula medidas.....	222
16.03 – Verifica uma escolha	225
16.04 – Mostra como utilizar estruturas de seleção dupla e composta.....	228
16.05 – Testando a instrução break	231
16.06 – Compara números	233
16.07 – Bônus salarial	235
16.08 – Testando instrução break	241
16.09 – Cálculo de áreas de figuras geométricas	242
16.10 – Números de votos por candidatos	247

EXERCÍCIOS DO C#.NET

No Visual Studio.NET para criar seus programas, faça o seguinte:

- No Menu File escolha New Project
- Escolha o Tipo de Projeto como Visual C# Projects
- Depois escolha como Templates o modelo Console Application
- Dê um nome ao seu programa e escolha a pasta desejada e pronto, você poderá testar o código.

Estes programas têm a finalidade de fazer com que o aluno assimile conceitos fundamentais no desenvolvimento de aplicações com o C#, utilizando o Visual Studio .NET ou o framework SDK para desenvolver os exemplos.

Os referidos programas (exercícios) tem como objetivo ensinar o usuário iniciante a criar programas simples utilizando entrada e saída de dados (I/O), como também a declaração de variáveis, expressões algébricas e estruturas de repetição e seleção.

Se não tiver o Visual Studio, vá no .NET Framework SDK utilize o Notepad (Bloco de Notas) para criar os exemplos e salve com o nome que desejar com a extensão .cs. (se for utilizar o framework sdk).

Abra o *prompt* do MS-DOS e execute:

```
C:\windows\microsoft.net\framework\v1.1.4322
```

Digite na *prompt* CSC e o nome do programa.cs na linha de comando.

Se você não conseguir compilar seus programas copie eles para dentro da pasta v1.1.4322

Após a compilação será gerado um arquivo executável com o nome do programa compilado. Digite o nome do programa e seu programa aparecerá.

Exercícios 01 – Entrada e Saída de Dados e Declaração de Variáveis

01.01 – Volume de uma caixa retangular

Calcular o volume de uma caixa retangular mostra de forma simples como declarar variáveis e manipular dados de entrada e saída via console

```
using System;

namespace Caixaretangular
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Caixa
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            double volume, // Conterá informações sobre a medida do volume da caixa
            comprimento, // Conterá informações sobre o comprimento da
            largura, // Conterá informações sobre a largura da caixa
            altura; // Conterá informações sobre a altura da

            // Solicita e lê informações do usuário sobre o comprimento da caixa retangular
            Console.WriteLine("Por favor entre com a medida do comprimento da caixa em metros: ");
            comprimento = double.Parse(Console.ReadLine());

            // Solicita e lê informações do usuário sobre a largura da caixa retangular
            Console.WriteLine("Por favor entre com a medida da largura da caixa em metros: ");
            largura = double.Parse(Console.ReadLine());

            // Solicita e lê informações do usuário sobre a altura da caixa retangular
            Console.WriteLine("Por favor entre com a medida da altura da caixa em metros: ");
            altura = double.Parse(Console.ReadLine());

            // Calcula o volume da caixa retangular
            volume = comprimento * largura * altura;

            // Exibe o resultado do volume da caixa retangular
```

```

        Console.WriteLine("O volume da caixa retangular com {0} metros de
comprimento, {1} metros de largura e {2} metros de altura é de: {3} metros cúbicos", comprimento,
largura, altura, volume);

    } // fim do método Main

} // Fim da classe Caixa
}

```

01.02 – Converter temperatura em graus Fahrenheit para Celsius

Converter a temperatura em graus Fahrenheit para graus Celsius. Utiliza console de entrada e saída de dados.

```

using System;

namespace Celsius
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Temperatura
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double C,      // Temperatura na escala Celsius
                F;          // Temperatura na escla Fahrenheit

            // Solicita e lê a informação da temperatura fornecida pelo usuário

            Console.WriteLine("Informe a temperatura na escala Fahrenheit: ");
            F = Double.Parse(Console.ReadLine());

            // Efetua a conversão da temperatura para a escala Celsius
            C = ((F - 32) * 5/9);

            // Exibe o resultado da conversão entre as temperaturas
            Console.WriteLine("\tA temperatura de {0} graus Fahrenheit equivale a: {1} graus
Celsius", F, C);

        } // Fim do método Main
    } // Fim da Classe Temperatura
}

```

01.03 - Calcula volume de um cilindro

Calcular o volume de um Cilindro. Mostra de forma simples a declaração de variáveis e entrada e saída de dados

```
using System;
```

```
namespace Cilindro
```

```
{
```

```
    /// <summary>
```

```
    /// Summary description for Class1.
```

```
    /// </summary>
```

```
    class Cilindro
```

```
    {
```

```
        /// <summary>
```

```
        /// The main entry point for the application.
```

```
        /// </summary>
```

```
        [STAThread]
```

```
        static void Main(string[] args)
```

```
        {
```

```
            // Declaração de Variáveis
```

```
            double raio, // Armazenará informações sobre a medida do raio da base do
```

```
            altura, // Armazenará informações sobre a medida do altura do
```

```
            volume, // Armazenará informações sobre a medida do volume do
```

```
            areab, // Armazenará informações da área da base do cilindro
```

```
            pi; // Uma constante...
```

```
            //Inicialização de Variáveis
```

```
            pi = 3.1415;
```

```
            // Solicita e lê a informação do raio do Cilindro fornecido pelo usuário
```

```
            Console.WriteLine("Por favor informe a medida do raio da base do cilindro: ");
```

```
            raio = double.Parse(Console.ReadLine());
```

```
            // Solicita e lê a informação da altura do Cilindro fornecido pelo usuário
```

```
            Console.WriteLine("Por favor informe a medida da altura do cilindro: ");
```

```
            altura = double.Parse(Console.ReadLine());
```

```
            // Realiza o cálculo do Área da base do Cilindro
```

```
            areab = (pi * raio * raio);
```

```
            // Realiza o cálculo do Volume do Cilindro
```

```
            volume = (areab * altura);
```

```
            // Exibe o resultado do Volume do Cilindro
```

```
            Console.WriteLine("O volume do cilindro é: {0}", volume);
```

```
        } // Fim do método Main
```

```
    } // Fim da classe Cilindro
```

```
}
```

01.04 – Curso de criação de coelhos

Calcular o custo com a criação de coelhos. Mostra de forma simples como manipular dados via console e declarar variáveis, e expressões algébricas:

```
using System;

namespace Animais
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Coelhos
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double custo;           // Armazenará os custos com a criação de
coelhos                             int nr_coelhos;           // Armazenará o número de coelhos

            // Solicita e lê o número de coelhos da criação
            Console.WriteLine("Por favor informe o número total de coelhos da criação: ");
            nr_coelhos = Int32.Parse(Console.ReadLine());

            // Calcula o custo com a criação de coelhos
            custo = (nr_coelhos * 0.70)/28;

            // Exibe o resultado do custo com os coelhos
            Console.WriteLine("O custo total com a criação de {0} coelhos, foi de: {1}",
nr_coelhos, custo);

            } // Fim do método Main

        } // Fim da classe Coelhos
    }
}
```

01.05 – Calcula consumo de combustível

Calcular o consumo de combustível por Km. Mostra de forma simples conceitos de entrada e saída de dados e declaração de variáveis:

```
using System;

namespace Combustível
```

```

{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Cálculo
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double KI,           // Armazenará a Kilometragem Inicial do Percurso do
Veículo
                                KF,           // Armazenará a Kilometragem Final do Percurso
do Veículo
                                CC,           // Armazenará o Consumo de Combustível
                                DP,           // Armazenará a Distância Percorrida
                                GC;           // Armazenará o Gasto Total do veículo com
Combustível

            // Solicita e lê a Kilometragem Inicial do Percurso
            Console.WriteLine("Por favor informe a Kilometragem Inicial: ");
            KI = double.Parse(Console.ReadLine());

            // Solicita e lê a Kilometragem Final do Percurso
            Console.WriteLine("Por favor informe a Kilometragem Final: ");
            KF = double.Parse(Console.ReadLine());

            // Efetua o cálculo da Distância Percorrida
            DP = (KF - KI);

            // Solicita e lê o Consumo de Combustível durante o Percurso realizado pelo
veículo Final do Percurso
            Console.WriteLine("Por favor informe o Consumo de Combustível em litros: ");
            CC = double.Parse(Console.ReadLine());

            // Efetua o cálculo do Gasto de Combustível
            GC = (DP/CC);

            // Exibe o total de Consumo de Combustível por Km
            Console.WriteLine("A cada Litro consumido o veículo percorreu: {0} Km ", GC);

        } // Fim do método Main

    } // Fim da classe Cálculo
}

```

01.06 – Calcula o quadrado da diferença entre números

Calcular o quadrado da diferença entre dois números quaisquer. Demonstra de forma simples o uso de consoles, entrada e saída de dados e o método pow:

```

using System;

namespace Diferença
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Numeros
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double N1,          // Número qualquer fornecido pelo usuário
                N2,             // Número qualquer fornecido pelo usuário
                DIF;           // Diferença do quadrado entre dois números

            // Solicita e lê dois números fornecidos pelo usuário

            Console.Write("Informe o 1º número: ");
            N1 = Double.Parse(Console.ReadLine());

            Console.Write("Informe o 2º número: ");
            N2 = Double.Parse(Console.ReadLine());

            // Calcula o Quadrado da Diferença entre os números fornecidos pelo usuário
            DIF = Math.Pow(N1,2) - Math.Pow(N2,2);

            // Exibe o resultado do quadrado da diferença entre os números fornecidos pelo
usuário
            Console.WriteLine("O quadrado da diferença entre os números {0} e {1} é de:
{2}", N1, N2, DIF);

        } // Fim do Método Main

    } // Fim da Classe Numeros
}

```

01.07 – Propriedade distributiva entre 4 números

Utilizar a propriedade distributiva entre 4 números nos quais devem ser somados e multiplicados entre si. Simples utilização de entrada, saída e declaração de variáveis:

```

using System;

namespace Distributiva
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Atributos
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double A,           // Primeiro número fornecido pelo usuário
                B,           // Segundo número fornecido pelo usuário
                C,           // Terceiro número fornecido pelo usuário
                D,           // Quarto número fornecido pelo usuário

                S_AB, // Soma do número A com o número B
                S_AC, // Soma do número A com o número C
                S_AD, // Soma do número A com o número D
                S_BC, // Soma do número B com o número C
                S_BD, // Soma do número B com o número D
                S_CD, // Soma do número C com o número D

                M_AB, // Multiplicação do número A com o número B
                M_AC, // Multiplicação do número A com o número C
                M_AD, // Multiplicação do número A com o número D
                M_BC, // Multiplicação do número B com o número C
                M_BD, // Multiplicação do número B com o número D
                M_CD; // Multiplicação do número C com o número D

            // Solicita e lê o 1º número a ser fornecido pelo usuário
            Console.WriteLine("\t\tPor favor entre com o 1º número: ");
            A = Double.Parse(Console.ReadLine());

            // Solicita e lê o 2º número a ser fornecido pelo usuário
            Console.WriteLine("\t\tPor favor entre com o 2º número: ");
            B = Double.Parse(Console.ReadLine());

            // Solicita e lê o 3º número a ser fornecido pelo usuário
            Console.WriteLine("\t\tPor favor entre com o 3º número: ");
            C = Double.Parse(Console.ReadLine());

            // Solicita e lê o 4º número a ser fornecido pelo usuário
            Console.WriteLine("\t\tPor favor entre com o 4º número: ");
            D = Double.Parse(Console.ReadLine());

            // Efetua a Propriedade distributiva da Soma
            S_AB = A + B;

```

```

S_AC = A + C;
S_AD = A + D;
S_BC = B + C;
S_BD = B + D;
S_CD = C + D;

// Efetua a Propriedade distributiva da Multiplicação
M_AB = A * B;
M_AC = A * C;
M_AD = A * D;
M_BC = B * C;
M_BD = B * D;
M_CD = C * D;

// Exibe o resultado da Propriedade distributiva da Soma e Multiplicação
Console.WriteLine();
Console.WriteLine("\t\tO valor da soma {0} + {1} = {2}",A, B, S_AB);
Console.WriteLine();
Console.WriteLine("\t\tO valor da soma {0} + {1} = {2}",A, C, S_AC);
Console.WriteLine();
Console.WriteLine("\t\tO valor da soma {0} + {1} = {2}",A, D, S_AD);
Console.WriteLine();
Console.WriteLine("\t\tO valor da soma {0} + {1} = {2}",B, C, S_BC);
Console.WriteLine();
Console.WriteLine("\t\tO valor da soma {0} + {1} = {2}",B, D, S_BD);
Console.WriteLine();
Console.WriteLine("\t\tO valor da soma {0} + {1} = {2}",C, D, S_CD);
Console.WriteLine();
Console.WriteLine("\t\tO valor da multiplicação {0} * {1} = {2}",A, B, M_AB);
Console.WriteLine();
Console.WriteLine("\t\tO valor da multiplicação {0} * {1} = {2}",A, C, M_AC);
Console.WriteLine();
Console.WriteLine("\t\tO valor da multiplicação {0} * {1} = {2}",A, D, M_AD);
Console.WriteLine();
Console.WriteLine("\t\tO valor da multiplicação {0} * {1} = {2}",B, C, M_BC);
Console.WriteLine();
Console.WriteLine("\t\tO valor da multiplicação {0} * {1} = {2}",B, D, M_BD);
Console.WriteLine();
Console.WriteLine("\t\tO valor da multiplicação {0} * {1} = {2}",C, D, M_CD);
Console.WriteLine();

} // Fim do método Main

} // Fim da Classe Atributos
}

```

01.08 – Calcula o volume de uma esfera

Calcular o volume de uma esfera. Mostra como declarar variáveis e utilizar console e o método pow.

```
using System;
```

```

namespace Esfera
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Vesfera
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double raio, // Raio do diâmetro da Esfera
            volume, // Volume da Esfera
            pi; // Uma constante..

            // Inicialização de Variáveis
            pi = 3.14159;

            // Solicita e lê as informações sobre a Esfera
            Console.WriteLine("\tInforme a medida do comprimento do raio da esfera em metros:");

            raio = Double.Parse(Console.ReadLine());
            Console.WriteLine();

            // Realiza o cálculo do volume da Esfera
            volume = (( 4 * pi * Math.Pow(raio,3)/3));

            // Exibe o resultado do volume da Esfera
            Console.WriteLine("\tO volume da esfera é: {0} metros cúbicos", volume);
            Console.WriteLine( );

        } // Fim do método Main

    } // Fim da classe Vesfera
}

```

01.09 – Conversão de temperaturas

Efetuar a conversão de temperatura da escala Celsius para a escala Fahrenheit. Demonstra de forma simples como declarar variáveis e utilizar uma console com o framework ou VS.NET.

```

using System;

namespace Fahrenheit
{
    /// <summary>

```

```

/// Summary description for Class1.
/// </summary>
class Temperatura
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main(string[] args)
    {
        // Declaração de Variáveis

        // Declaração de Variáveis

        double C,      // Temperatura na escala Celsius
               F;      // Temperatura na escla Fahrenheit

        // Solicita e lê a informação da temperatura fornecida pelo usuário

        Console.WriteLine("\tInforme a temperatura na escala Celsius: ");
        C = Double.Parse(Console.ReadLine());

        // Efetua a conversão da temperatura para a escala Celsius
        F = (((9 * C)+ 160)/5);

        // Exibe o resultado da conversão entre as temperaturas
        Console.WriteLine("\tA temperatura de {0} graus Celsius equivale a: {1} graus
Fahrenheit", C, F);

    } // Fim do método Main

} // Fim da Classe Temperatura
}

```


Exercícios 02 – Entradas e Saídas de Dados e Declaração de Variáveis

02.01 – Calcula o produto e a adição de frações

Calcular o produto e a adição de duas frações. Mostra de forma simples como utilizar variáveis e manipular dados via console

```
using System;
```

```
namespace Frações  
{
```

```
    /// <summary>  
    /// Summary description for Class1.  
    /// </summary>  
    class Fração  
    {  
        /// <summary>  
        /// The main entry point for the application.  
        /// </summary>  
        [STAThread]  
        static void Main(string[] args)  
        {
```

```
            // Declaração de variáveis
```

```
            double NUM1,          // Numerador1 de uma fração qualquer  
                  NUM2,          // Numerador2 de uma fração qualquer  
                  DEN1,          // Denominador1 de uma fração qualquer  
                  DEN2,          // Denominador1 de uma fração qualquer  
                  FRAC1,          // Armazenará o resultado do  
numerator1 e denominator1      FRAC2,          // Armazenará o resultado do  
numerator2 e denominator2  
                                 SOMA,          // Armazenará a soma das duas frações  
                                 PRODUTO;      // Armazenará o produto das duas frações
```

```
            //Solicita e lê o número informado pelo usuário para o 1ºNumerador  
            Console.Write("Por favor informe o 1º numerador da fração: ");  
            NUM1 = double.Parse(Console.ReadLine());
```

```
            //Solicita e lê o número informado pelo usuário para o 1ºDenominador  
            Console.Write("Por favor informe o 1º denominador da fração: ");  
            DEN1 = double.Parse(Console.ReadLine());
```

```
            //Solicita e lê o número informado pelo usuário para o 2ºNumerador  
            Console.Write("Por favor informe o 2º numerador da fração: ");  
            NUM2 = double.Parse(Console.ReadLine());
```

```
            //Solicita e lê o número informado pelo usuário para o 2ºDenominador  
            Console.Write("Por favor informe o 2º denominador da fração: ");  
            DEN2 = double.Parse(Console.ReadLine());
```

```

        // Calcula as frações
        FRAC1 = (NUM1/DEN1);
        FRAC2 = (NUM2/DEN2);

        // Calcula a soma das frações
        SOMA = (FRAC1 + FRAC2);
        PRODUTO = (FRAC1 * FRAC2);

        // Exibe o resultado da adição e multiplicação das frações
        Console.WriteLine("A soma das frações {0} e {1} foi de: {2}", FRAC1, FRAC2,
SOMA);
        Console.WriteLine("O produto das frações {0} e {1} foi de: {2}", FRAC1, FRAC2,
PRODUTO);

    } // Fim do método Main
} // Fim da classe Fração
}

```

02.02 – Calcula o salário de um vendedor

Calcular o salário total de um vendedor. Define pontos importantes de como se utilizar console de entrada e saída de dados

```

using System;

namespace Vendedor
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Empregado
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            int            cod_funcionario;        // Número que identifica o funcionário da
empresa
            double  salario_fixo,                // Salário definido que o funcionário receberá
todo mês (o fixo)
            total_vendas,                // Total das Vendas efetuadas pelo
funcionário
            percentual_ganho,            // Percentual ganho sobre o total das
vendas efetuadas pelo funcionário
            salario_total;                // É a soma do salário fixo com o
percentual de ganho sobre as vendas

            // Solicita e lê as informações do Empregado

```



```

        // Solicita e lê as informações da lata de óleo

        Console.WriteLine("\tPor favor informe a medida do raio: ");
        raio = Double.Parse(Console.ReadLine());
        Console.WriteLine();

        Console.WriteLine("\tPor favor informe a medida da altura: ");
        altura = Double.Parse(Console.ReadLine());
        Console.WriteLine();

        // Calcula o volume da lata de óleo
        volume = ((3.14159 * raio)-(2 * altura));

        // Exibe o resultado do volume da lata de óleo
        Console.WriteLine("\tO volume da lata de óleo é: {0}", volume);
        Console.WriteLine();

    } // Fim do método Main

} // Fim da Classe Cálculo
}

```

02.04 – Média harmônica

Calcular a média harmônica das notas de um aluno. Define como utilizar uma console manipulando dados de entrada e saída e declarando variáveis

```

using System;

namespace MediaHarmônica
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Harmônica
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {

            // Declaração de Variáveis
            double N1,           // Primeira nota do aluno
                    N2,           // Segunda nota do aluno
                    N3,           // Terceira nota do aluno
                    N4,           // Quarta nota do aluno
                    MEDIAH;       // Média Harmônica

            // Solicita e lê as notas de um aluno

```

```

        Console.WriteLine("Por favor informe a 1ª nota do aluno: ");
        N1 = Double.Parse(Console.ReadLine());

        Console.WriteLine("Por favor informe a 2ª nota do aluno: ");
        N2 = Double.Parse(Console.ReadLine());

        Console.WriteLine("Por favor informe a 3ª nota do aluno: ");
        N3 = Double.Parse(Console.ReadLine());

        Console.WriteLine("Por favor informe a 4ª nota do aluno: ");
        N4 = Double.Parse(Console.ReadLine());

        // Efetua o cálculo da Média Harmônica
        MEDIAH = (4/((1/N1) + (1/N2) + (1/N3) + (1/N4)));

        // Exibe a Média Harmônica
        Console.WriteLine("O valor da Média Harmônica é: {0}", MEDIAH);

    } // Fim do método Main

} // Fim da classe Harmônica
}

```

02.05 – Média ponderada

Calcular a Média Ponderada de duas provas realizadas por um aluno. Mostra de forma simples como declarar variáveis e manipular dados de entrada e saída

```

using System;

namespace Média_Ponderada
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Média
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de variáveis
            double N1,          // Primeira nota da prova do aluno
                   N2,          // Segunda nota da prova do aluno
                   P1,          // Peso da primeira prova do aluno
                   P2,          // Peso da segunda prova do aluno
                   MEDIAP;      // Armazenará o resultado da Média Ponderada

            do Aluno

```

```

// Solicita e lê a primeira nota de prova do aluno
Console.Write("Por favor informe a 1º nota do aluno: ");
N1 = double.Parse(Console.ReadLine());

// Solicita e lê a segunda nota de prova do aluno
Console.Write("Por favor informe a 2º nota do aluno: ");
N2 = double.Parse(Console.ReadLine());

// Solicita e lê o peso da 1ª prova do aluno
Console.Write("Por favor informe o Peso da 1º prova do aluno: ");
P1 = double.Parse(Console.ReadLine());

// Solicita e lê o peso da 2ª prova do aluno
Console.Write("Por favor informe o Peso da 2º prova do aluno: ");
P2 = double.Parse(Console.ReadLine());

// Efetua o cálculo da Média Ponderada do Aluno
MEDIAP = (N1*P1 + N2*P2)/(P1 + P2);

// Exibe o resultado obtido da Média Ponderada
Console.WriteLine(" O resultado da média do aluno foi de: {0}", MEDIAP);

} // Fim da método Main

} // Fim da classe Média
}

```

02.06 – Lê números e calcula operações

Ler dois reais e efetuar as quatro operações entre eles. Demonstra de forma simples como efetuar cálculos, declarar variáveis e utilizar a console para manipular dados

```

using System;

namespace Operações
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Contas
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            double N1, N2, // Números a ser fornecidos pelo usuário
                SOM, // Armazenará o resultado da soma entre os dois
                números

```

```

SUB, // Armazenará o resultado da subtração entre os
dois números
MUL, // Armazenará o resultado da multiplicação entre
os dois números
DIV; // Armazenará o resultado da divisão entre os
dois números

// Solicita e lê os números

Console.WriteLine("\tEntre com o 1º número: ");
N1 = Double.Parse(Console.ReadLine());
Console.WriteLine();

Console.WriteLine("\tEntre com o 2º número: ");
N2 = Double.Parse(Console.ReadLine());
Console.WriteLine();

// Realiza as operações com os números fornecidos

SOM = (N1 + N2);
SUB = (N1 - N2);
MUL = (N1 * N2);
DIV = (N1 / N2);

// Exibe o resultado das operações

Console.WriteLine("\tA soma dos números {0} e {1} é de: {2}", N1, N2, SOM);
Console.WriteLine();
Console.WriteLine("\tA subtração dos números {0} e {1} é de: {2}", N1, N2, SUB);
Console.WriteLine();
Console.WriteLine("\tA multiplicação dos números {0} e {1} é de: {2}", N1, N2,
MUL);

Console.WriteLine();
Console.WriteLine("\tA divisão dos números {0} e {1} é de: {2}", N1, N2, DIV);
Console.WriteLine();

} // Fim do método Main

} // Fim da classe Contas
}

```

02.07 – Calcula potência de número

Calcular a potência de um número informado. Simples utilização de entrada, saída e declaração de variáveis e o método Math.Pow

```

using System;

namespace Potência
{
    /// <summary>
    /// Summary description for Class1.

```

```

/// </summary>
class Exponencial
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main(string[] args)
    {
        // Declaração de Variáveis

        double N,      // Um número fornecido pelo usuário
                EXP,    // O expoente no qual o número será elevado
                POT;    // Armazenará a potência do número

        // Solicita e lê o número fornecido pelo usuário
        Console.Write("Entre com um número: ");
        N = Double.Parse(Console.ReadLine());

        // Solicita e lê o expoente que o número será elevado
        Console.Write("Entre com o expoente: ");
        EXP = Double.Parse(Console.ReadLine());

        // Calcula a Potência do Número informado
        POT = Math.Pow(N, EXP);

        // Exibe o resultado do Número elevado ao expoente
        Console.WriteLine("A potência do número {0} com expoente {1} é de: {2}", N,
EXP, POT);

    } // Fim do método Main

} // Fim da classe Exponencial
}

```

02.08 – Quadrado de um número

Calcular o quadrado de um número qualquer. Simples utilização de entrada e saída de dados via console e o método Math.Pow

```

using System;

namespace Elevado
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Expoente

```



```

{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main(string[] args)
    {
        // Declaração de Variáveis

        double N,      // Número qualquer fornecido pelo usuário
                X;      // Armazenará o do número elevado ao quadrado

        // Solicita e lê um número qualquer informado pelo usuário
        Console.Write("Por favor informe um número qualquer: ");
        N = Double.Parse(Console.ReadLine());

        // Efetua o cálculo de um número qualquer ao quadrado
        X = Math.Pow(N,2);

        // Exibe o resultado do número elevado ao quadrado
        Console.WriteLine("O número {0} elevado a potência 2 é: {1}", N, X);
    } // Fim do método Main
} // Fim da Classe Expoente
}

```

02.09 – Área de um quadrado

Calcular a área de um quadrado. Demonstra a simples utilização de entrada e saída de dados

```

using System;

namespace Quadrado
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Cálculo
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double lado,      // Medida do lado do quadrado
                    area;     // Armazenará o cálculo da área do quadrado

```

```
// Solicita e lê a informação da medida do lado do quadrado
Console.WriteLine("Por favor informe a medida do lado do quadrado: ");
lado = double.Parse(Console.ReadLine());

// Efetua o cálculo da Área do Quadrado
area = lado * lado;

// Exibe o resultado
Console.WriteLine(" A área do quadrado é: {0}", area);

} // Fim da método Main

} // Fim da classe Cálculo

}
```

Exercícios 03 – Entrada e Saída de Dados e Declaração de Variáveis

03.01 Consumo de combustível

Calcular o consumo de combustível, ou seja a quantidade de litros consumidos em uma viagem de veículo. Utiliza de forma simples e clara conceitos de declaração de variáveis e console

```
using System;

namespace Consumo
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Combustível
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            double          qtde_litros,    // Quantidade de litros consumidos (gastos) na
viagem
                                                tempo,          // Tempo gasto na viagem (em
horas)
                                                velocidade,      // Velocidade média do veículo
durante o percurso
                                                km_litros,       // Quantidade de kilometros por
litro que o veículo realiza
                                                distancia;       // Distância do percurso da
viagem

            // Solicita e lê as informações da viagem
            Console.WriteLine("\tInforme o tempo em horas do percurso: ");
            tempo = Double.Parse(Console.ReadLine());
            Console.WriteLine();

            Console.WriteLine("\tInforme a velocidade média em Km/h do veículo: ");
            velocidade = Double.Parse(Console.ReadLine());
            Console.WriteLine();

            Console.WriteLine("\tInforme a quantidade de Kilometros que o veículo faz com 1
litro: ");

            km_litros = Double.Parse(Console.ReadLine());
            Console.WriteLine();

            // Efetua o cálculo da distância percorrida e da quantidade de litros consumida na
viagem
            distancia = (velocidade * tempo);
            qtde_litros = (distancia / km_litros);
```

```

        // Exibe a distância percorrida da viagem
        Console.WriteLine("\tA distância do percurso foi de: {0} Km", distancia);
        Console.WriteLine();

        // Exibe o resultado da quantidade de litros de combustível gasto na viagem
        Console.WriteLine("\tA quantidade de combustível gasto na viagem foi de: {0}
litros", qtde_litros);
        Console.WriteLine();

    } // Fim do método Main

} // Fim da Classe Combustível
}

```

03.02 – Troca de valores entre números

Efetuar a troca dos valores de dois números. Utiliza conceitos de declaração de variáveis, entrada e saída de dados

```

using System;

namespace Trocas
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Troca
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de variáveis
            double N1,          // Primeiro número a ser fornecido pelo usuário
                N2,             // Segundo número a ser fornecido pelo usuário
                T;              // Variável auxiliar para efetuar a troca dos
números

            // Solicita e lê os números a serem fornecidos pelo usuário
            Console.Write("Por favor entre com o 1º número X: ");
            N1 = Double.Parse(Console.ReadLine());

            Console.Write("Por favor entre com o 2º número Y: ");
            N2 = Double.Parse(Console.ReadLine());

            // Efetua a troca dos valores fornecidos

            T = N1;
            N1 = N2;
            N2 = T;

            // Exibe os números fornecidos pelo usuário de forma invertida
            Console.WriteLine("O número fornecido X = {0}, agora é: {1}", N2,N1);
            Console.WriteLine("O número fornecido Y = {0}, agora é: {1}", N1,N2);

```

```

        } // Fim do método Main

    } // Fim da classe Troca
}

```

03.03 – Entrada e saída de dados

Apresentar conceitos de entrada e saída de dados. Realizar o cálculo da área de um triângulo.

```

using System;

namespace Triângulo
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Cálculo
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double baset, // Armazena informações da base do triângulo
                altura, // Armazena informações da altura do triângulo
                area; // Armazenará o resultado do cálculo da área do triângulo

            // Solicita e lê a informação da base do triângulo
            Console.Write("Por favor informe a medida da Base do Triângulo: ");
            baset = Double.Parse(Console.ReadLine());

            // Solicita e lê a informação da altura do triângulo
            Console.Write("Por favor informe a medida da Altura do Triângulo: ");
            altura = Double.Parse(Console.ReadLine());

            // Efetua o cálculo da área do triângulo
            area = (baset * altura)/2;

            // Exibe o resultado
            Console.WriteLine("A Área do triângulo é: {0}", area);

        } // Fim da método Main

    } // Fim da classe Cálculo
}

```

03.04 - Conversão de temperaturas

Fazer a conversão de temperatura na escala Fahrenheit para a escala Celsius. Utiliza os simples conceitos de declaração de variáveis e manipulação de dados via console.

```
using System;

namespace Conversão
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Temperatura
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double F,          // Temperatura na escala Fahrenheit
                  C;          // Temperatura na escala Celsius

            //Solicita e lê a informação da temperatura na escala Fahrenheit
            Console.WriteLine("\t\tPor favor informe a temperatura na escala Fahrenheit: ");
            F = Double.Parse(Console.ReadLine());
            Console.WriteLine();

            // Efetua a conversão de temperatura
            C = ((F-32)*5)/9);

            // Exibe o resultado da Conversão de temperatura
            Console.WriteLine("\t\tA temperatura na escala Celsius é: {0} ", C);
            Console.WriteLine();

        } // Fim do método Main
    } // Fim da classe Temperatura
}
```

03.05 – Prestação em atraso

Calcular o valor de uma prestação em atraso. Mostra de forma simples como utilizar uma console com entrada e saída de dados e declarar variáveis.

```
using System;

namespace Taxas
{
    /// <summary>
```

```

/// Summary description for Class1.
/// </summary>
class Taxa
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main(string[] args)
    {
        // Declaração de Variáveis
        int        dias;           // Dias em atraso
        double valor,             // Valor da prestação em atraso
            taxa,                 // Valor da taxa de juros
            prestacao;           // Valor da prestação a pagar

        // Solicita e lê o valor da prestação em atraso
        Console.Write("Informe o valor da prestação em atraso: ");
        valor = Double.Parse(Console.ReadLine());

        // Solicita e lê a taxa de juros definida pelo funcionário
        Console.Write("Informe a taxa de juros: ");
        taxa = Double.Parse(Console.ReadLine());

        // Solicita e lê a informação dos dias em atraso
        Console.Write("Informe a quantidade de dias em atraso: ");
        dias = Int32.Parse(Console.ReadLine());

        // Calcula o valor da nova prestação a ser paga
        prestacao = (valor + (valor * (taxa/100) * dias));

        // Exibe o novo valor da prestação a ser paga
        Console.WriteLine("O valor a ser paga será de: {0}", prestacao);

    } // Fim do método Main

} // Fim da classe Taxa
}

```

03.06 – Soma do quadrado de números

Calcular a soma dos quadrados de 3 números. Usa conceitos simples de entrada e saída de dados e declaração de variáveis.

```

using System;

namespace Squadrados
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Quadrados
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)

```

```

        {
            // Declaração de Variáveis

            double N1,    // Primeiro número a ser fornecido pelo usuário
                N2,    // Segundo número a ser fornecido pelo usuário
                N3,    // Terceiro número a ser fornecido pelo usuário
                S;      // Soma dos quadrados dos números fornecidos pelo
usuário

            // Solicita e lê os números fornecidos pelo usuário

            Console.WriteLine("Por favor informe o 1º número: ");
            N1 = Double.Parse(Console.ReadLine());

            Console.WriteLine("Por favor informe o 2º número: ");
            N2 = Double.Parse(Console.ReadLine());

            Console.WriteLine("Por favor informe o 3º número: ");
            N3 = Double.Parse(Console.ReadLine());

            // Calcula a soma dos quadrados dos números fornecidos
            S = (Math.Pow(N1,2) + Math.Pow(N2,2) + Math.Pow(N3,2));

            // Apresenta o resultado da soma dos quadrados
            Console.WriteLine(" A soma dos quadrados dos números {0}, {1} e {2} é de: {3}",
N1, N2, N3, S);

        } // Fim do método Main

    } // Fim da classe Quadrados
}

```

03.07 – Salário de funcionário

Calcular o salário de um funcionário. Mostra como declarar variáveis, e utilizar a console com entrada e saída de dados.

```

using System;

namespace Salário
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Funcionário
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de variáveis

            double n_horas,    // Número de horas trabalhadas
                v_hora,    // Valor da hora trabalhada

```



```

        v_sf, // Valor do salário família
        salario_final; // Valor do salário do funcionário

int        n_func, // Número de identificação do funcionário
        n_filhos; // Número de filhos do funcionário

// Solicita e lê o Código do Funcionário
Console.Write("Entre com a matrícula do funcionário: ");
n_func = Int32.Parse(Console.ReadLine());

// Solicita e lê a Quantidade de Horas Trabalhadas do Funcionário
Console.Write("Informe a quantidade de horas trabalhadas do funcionário: ");
n_horas = Double.Parse(Console.ReadLine());

// Solicita e lê o valor da Hora Trabalhada do Funcionário
Console.Write("Informe o valor da Hora de Trabalho do funcionário: ");
v_hora = Double.Parse(Console.ReadLine());

// Solicita e lê a Quantidade de filhos menores de 14 anos do Funcionário
Console.Write("Informe a quantidade de filhos do funcionários com idade menor
que 14 anos: ");
n_filhos = Int32.Parse(Console.ReadLine());

// Solicita e lê o Valor do Salário Família por criança
Console.Write("Informe o valor do salário família por filho: ");
v_sf = Double.Parse(Console.ReadLine());

// Calcula o valor do Salário do Funcionário
salario_final = ((n_horas * v_hora) + (n_filhos * v_sf));

// Exibe o valor do Salário do Funcionário
Console.WriteLine("O valor do salário do funcionário de matrícula {0} é de: {1}",
n_func, salario_final);

// TODO: Add code to start application here
//
} // Fim do método Main

} // Fim da classe Funcionário
}

```

03.08 – Salário mensal

Calcular o salário mensal de um vendedor de uma revendedora de carros. Utiliza declaração de variáveis e manipulação de dados entrada/saída.

```

using System;

namespace Revendedora
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Vendedor
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>

```

```

[STAThread]
static void Main(string[] args)
{
    // Declaração de Variáveis

    int          cod_vend,          // Número de identificação do vendedor
    vendedor     car_vend;          // Número de carros vendidos pelo

    double sal_fixo,                // Salário Fixo do vendedor
    vendedor     percent,           // Percentual sobre o total das vendas do
    vendedor     total_vend,        // Total das vendas com carros do
    vendedor     val_pcar,          // Valor de comissão a ser ganha por
    carro vendido sal_mensal;       // Salário Mensal do Vendedor

    // Inicialização de Variáveis
    percent = 0.05;                // Cinco por cento

    // Solicita e lê as informações do vendedor
    Console.Write("Informe a matrícula do vendedor: ");
    cod_vend = Int32.Parse(Console.ReadLine());
    Console.WriteLine();

    // Solicita e lê o valor do salário fixo do vendedor
    Console.Write("Entre com o valor do salário fixo do vendedor: ");
    sal_fixo = Double.Parse(Console.ReadLine());
    Console.WriteLine();

    // Solicita e lê a quantidade de carros vendidos pelo vendedor
    Console.Write("Entre com o número de carros vendidos pelo vendedor: ");
    car_vend = Int32.Parse(Console.ReadLine());
    Console.WriteLine();

    // Solicita e lê o valor ganho por carro vendido
    Console.Write("Entre com o valor de comissão a ser ganho por carro vendido
    pelo vendedor: ");

    val_pcar = Double.Parse(Console.ReadLine());
    Console.WriteLine();

    // Solicita e lê o valor total das vendas do vendedor
    Console.Write("Entre com o valor das vendas do vendedor: ");
    total_vend = Double.Parse(Console.ReadLine());
    Console.WriteLine();

    // Efetua o cálculo do salário total do vendedor
    sal_mensal = sal_fixo + (val_pcar * car_vend) + (total_vend * percent);

    // Exibe o valor do Salário Total do Vendedor
    Console.WriteLine("O Vendedor {0} tem salário total de: {1}", cod_vend,
    sal_mensal);

    Console.WriteLine();

} // Fim do método Main

} // Fim da classe Funcionário
}

```

03.09 – Velocidade de veículo

Calcular a velocidade média de um veículo qualquer. Mostra de forma simples a utilização de entrada e saída de dados via console e como declarar variáveis.

```
using System;

namespace VelocidadeM
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Velocidade
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double velocidade, // Armazenará informações da velocidade do veículo
            distancia, // Armazenará informações da distância
            percorrida pelo veículo
            tempo; // Armazenará informações do tempo gasto no
            percurso

            // Solicita e lê a distância percorrida pelo veículo
            Console.WriteLine("Por favor informe a distância percorrida pelo veículo em
            Kilômetros: ");

            distancia = double.Parse(Console.ReadLine());

            // Solicita e lê o tempo gasto no percurso pelo veículo
            Console.WriteLine("Por favor informe o tempo gasto na viagem em horas: ");
            tempo = double.Parse(Console.ReadLine());

            // Efetua o cálculo da Velocidade Média
            velocidade = (distancia/tempo);

            // Exibe o resultado da Velocidade Média
            Console.WriteLine("A velocidade Média do veículo é: {0} Km/h", velocidade);

        } // Fim do método Main

    } // Fim da classe Velocidade
}
```

Exercícios 04 – Estrutura de Seleção IF e IF / ELSE

04.01 – Verifica número informado

Este programa utiliza conceitos de estruturas de seleção, operadores relacionais e utilização de blocos de instruções. Verificar em que faixa está localizada um número informado pelo usuário.

```
using System;

namespace Verificação
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Faixa
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double N;          // Número a ser fornecido pelo usuário

            // Solicita e lê um número do usuário
            Console.WriteLine("\tInforme um número: ");
            N = Double.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            // Verifica se o número está na faixa de 20 a 90

            // Estrutura de Seleção Composta

            if ((N >= 20) && (N <= 90)) // Utilize chaves para mais de uma instrução
            (blocos de instruções)
            {
                Console.WriteLine("\tO número está na faixa de 20 a 90");
                // Exibe uma linha em branco
                Console.WriteLine();
            }

            else
            {
                Console.WriteLine("\tO número não está na faixa de 20 a 90");
                // Exibe uma linha em branco
                Console.WriteLine();
            }
        } // Fim do método Main
    } // Fim da Classe Faixa
}
```

04.02 – Verificar o maior e o menor número dados

Identificar o maior e o menor número dentre 5 números fornecidos pelo usuário

```
using System;

namespace Valores
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Compara
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int N1, N2, N3, N4, N5, MAIOR, MENOR;

            // Números que serão fornecidos pelo usuário
            // Armazenará o maior valor fornecido pelo usuário
            // Armazenará o menor valor fornecido pelo usuário

            // Solicita e lê os números fornecidos pelo usuário
            Console.WriteLine("Informe o valor de N1: ");
            N1 = Int32.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            Console.WriteLine("Informe o valor de N2: ");
            N2 = Int32.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            Console.WriteLine("Informe o valor de N3: ");
            N3 = Int32.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            Console.WriteLine("Informe o valor de N4: ");
            N4 = Int32.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            Console.WriteLine("Informe o valor de N5: ");
            N5 = Int32.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco
            Console.WriteLine(); // Adiciona uma linha em branco

            // Verifica qual o maior dentre os números fornecidos
            // Utilize chaves para mais de uma instrução (blocos de instruções)
            if ((N1 >= N2) && (N1 >= N3) && (N1 >= N4) && (N1 >= N5))
            {
                MAIOR = N1;
                Console.WriteLine("O número N1 é o maior: {0}", MAIOR);
                Console.WriteLine();
            }
            else
```

```

{
    if (( N2 >= N1) && (N2 >= N3) && (N2 >= N4) && (N2 >= N5))
    {
        MAIOR = N2;
        Console.WriteLine("\t\tO número N2 é o maior: {0}", MAIOR);
        Console.WriteLine();
    }
    else
    {
        if (( N3 >= N1) && (N3 >= N2) && (N3 >= N4) && (N3 >= N5))
        {
            MAIOR = N3;
            Console.WriteLine("\t\tO número N3 é o maior: {0}",
                MAIOR);

            Console.WriteLine();
        }
        else
        {
            if (( N4 >= N1) && (N4 >= N2) && (N4 >= N3) && (N4 >=
                N5))
            {
                MAIOR = N4;
                Console.WriteLine("\t\tO número N4 é o maior:
                    {0}", MAIOR);

                Console.WriteLine();
            }
            else
            {
                MAIOR = N5;
                Console.WriteLine("\t\tO número N5 é o maior:
                    {0}", MAIOR);

                Console.WriteLine();
            }
        }
    }

    // Verifica qual o menor número dentre os números fornecidos
    if (( N1 <= N2) && ( N1 <= N3) && ( N1 <= N4) && ( N1 <= N5))
    {
        MENOR = N1;
        Console.WriteLine("\t\tO número N1 é o menor: {0}", MENOR);
        Console.WriteLine();
    }
    else
    {
        if (( N2 <= N1) && ( N2 <= N3) && ( N2 <= N4) && ( N2 <= N5))
        {
            MENOR = N2;
            Console.WriteLine("\t\tO número N2 é o menor: {0}",
                MENOR);

            Console.WriteLine();
        }
        else
        {
            if (( N3 <= N1) && ( N3 <= N2) && ( N3 <= N4) && ( N3
                <= N5))
            {
                MENOR = N3;
                Console.WriteLine("\t\tO número N3 é o menor:
                    {0}", MENOR);
            }
        }
    }
}

```

```

        Console.WriteLine();
    }
    else
    {
        if (( N4 <= N1) && ( N4 <= N2) && ( N4 <= N3)
        {
            MENOR = N4;
            Console.WriteLine("\t\tO número N4 é o
            Console.WriteLine();
        }
        else
        {
            MENOR = N5;
            Console.WriteLine("\t\tO número N5 é o
            Console.WriteLine();
        }
    }
}
}
}
} // Fim do método Main
} // Fim da Classe Comparar
}

```

04.03 – Verificar se medidas informadas é um triângulo

Verificar se as medidas informadas formam um triângulo e se for, qual o seu tipo.

```

using System;

namespace VTriangulos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Tipos
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double A, B, C; // Lados de um triângulo qualquer

            // Solicita e lê as medidas para um triângulo qualquer
            Console.Write("\t\tInforme o 1º lado: ");
            A = Double.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

```

```

        Console.WriteLine("\t\tInforme o 2º lado: ");
        B = Double.Parse(Console.ReadLine());
        Console.WriteLine();    // Adiciona uma linha em branco

        Console.WriteLine("\t\tInforme o 3º lado: ");
        C = Double.Parse(Console.ReadLine());
        Console.WriteLine();    // Adiciona uma linha em branco

        // Condições de existência

        /*
        * (A < B + C) ou (B < A + C) ou (C < A + B)
        * (A = B) ou (A = C) ou (B = C) ---> Triângulo Isósceles
        * (A ? B) e (B ? C) ---> Triângulo Escaleno
        * (A = B) e (B = C) ---> Triângulo Equilátero
        */

        // Verifica a condição de existência para ser um triângulo
        // Utilize chaves para mais de uma instrução (blocos de instruções)
        if ((A < B + C) && (B < A + C) && (C < A + B))
        {
            if ((A == B) && (B == C))
                Console.WriteLine("\t\tTriângulo Equilátero!");
            else
            {
                if ((A == B) || (A == C) || (B == C))
                    Console.WriteLine("\t\tTriângulo Isósceles!");
                else
                    Console.WriteLine("\t\tTriângulo Escaleno!");
            }
        }
        else
            Console.WriteLine("\t\tAs medidas fornecidas não formam um
triângulo!");

        Console.WriteLine();

    } // Fim do método Main

} // Fim da classe Tipos
}

```

04.04 – Ler dois números e verificar a diferença

Ler dois valores numéricos e apresentar a diferença do maior pelo menor.

```

using System;

namespace Doisnúmeros
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Diferença
    {
        /// <summary>
        /// The main entry point for the application.

```



```

/// </summary>
[STAThread]
static void Main(string[] args)
{
    // Declaração de Variáveis
    double N1, N2,          // Números que serão fornecidos pelo usuário
    SUBT;                   // Diferença entre o maior e o menor
número

    // Solicita e lê os números
    Console.WriteLine("\t\tInforme o valor de N1: ");
    N1 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme o valor de N2: ");
    N2 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    // Compara qual número é o maior e o menor

    // Utilize chaves para mais de uma instrução (blocos de instruções)
    if (N1 > N2)
    {
        SUBT = N1 - N2;
        Console.WriteLine("\t\tA diferença de {0} " + "-" + " {1} é: {2}", N1, N2,
SUBT);

        Console.WriteLine();    // Adiciona uma linha em branco
    }
    else
    {
        SUBT = N2 - N1;
        Console.WriteLine("\t\tA diferença de {0} " + "-" + " {1} é: {2}", N2, N1,
SUBT);

        Console.WriteLine();    // Adiciona uma linha em branco
    }

    } // Fim do Método Main

} // Fim da Classe Diferença
}

```

04.05 – Aceita números e verifica seus divisores

Aceitar quatro números do usuário e verificar quais são divisíveis por 2 ou 3.

```

using System;

namespace Divisíveis
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>

```

```

class Números
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main(string[] args)
    {
        // Declaração de Variáveis
        int N1, N2, N3, N4; // Números fornecidos pelo usuário

        // Solicita os números do usuário
        Console.WriteLine("\t\tInforme o 1º número: ");
        N1 = Int32.Parse(Console.ReadLine());
        Console.WriteLine();

        Console.WriteLine("\t\tInforme o 2º número: ");
        N2 = Int32.Parse(Console.ReadLine());
        Console.WriteLine();

        Console.WriteLine("\t\tInforme o 3º número: ");
        N3 = Int32.Parse(Console.ReadLine());
        Console.WriteLine();

        Console.WriteLine("\t\tInforme o 4º número: ");
        N4 = Int32.Parse(Console.ReadLine());
        Console.WriteLine();

        // Executa o teste para verificar se N1 é divisível por 2
        // Estruturas de Seleção Simples If

        // Utilize chaves para mais de uma instrução (blocos de instruções)
        if (N1 % 2 == 0)
        {
            Console.WriteLine("\t\tO número {0} é divisível por 2", N1);
            Console.WriteLine();
        }

        // Executa o teste para verificar se N1 é divisível por 3
        if (N1 % 3 == 0)
        {
            Console.WriteLine("\t\tO número {0} é divisível por 3", N1);
            Console.WriteLine();
        }

        // Executa o teste para verificar se N2 é divisível por 2
        if (N2 % 2 == 0)
        {
            Console.WriteLine("\t\tO número {0} é divisível por 2", N2);
            Console.WriteLine();
        }

        // Executa o teste para verificar se N2 é divisível por 3
        if (N2 % 3 == 0)
        {
            Console.WriteLine("\t\tO número {0} é divisível por 3", N2);
            Console.WriteLine(); // Insere uma linha em branco
        }

        // Executa o teste para verificar se N3 é divisível por 2
    }
}

```

```

        if (N3 % 2 == 0)
        {
            Console.WriteLine("\t\tO número {0} é divisível por 2", N3);
            Console.WriteLine();    // Insere uma linha em branco
        }

        // Executa o teste para verificar se N3 é divisível por 3
        if (N3 % 3 == 0)
        {
            Console.WriteLine("\t\tO número {0} é divisível por 3", N3);
            Console.WriteLine();    // Insere uma linha em branco
        }

        // Executa o teste para verificar se N4 é divisível por 2
        if (N4 % 2 == 0)
        {
            Console.WriteLine("\t\tO número {0} é divisível por 2", N4);
            Console.WriteLine();    // Insere uma linha em branco
        }

        // Executa o teste para verificar se N4 é divisível por 3
        if (N4 % 3 == 0)
        {
            Console.WriteLine("\t\tO número {0} é divisível por 3", N4);
            Console.WriteLine();    // Insere uma linha em branco
        }

    } // Fim do método Main

} // Fim da Classe Números
}

```

04.06 – Lê dois número e verifica suas diferenças

Ler dois número e apresentar a diferença do maior pelo menor

```

using System;

namespace Diferença
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Numeros
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    N1, N2, // Números a serem fornecidos pelo usuário
                RESULTADO; // O Resultado da diferença do maior número pelo menor
        }
    }
}

```

```

// Solicita e lê os números do usuário
Console.WriteLine("\t\tInforme o 1º número: ");
N1 = Int32.Parse(Console.ReadLine());
Console.WriteLine(); // Adiciona uma linha em branco

Console.WriteLine("\t\tInforme o 2º número: ");
N2 = Int32.Parse(Console.ReadLine());
Console.WriteLine(); // Adiciona uma linha em branco

// Converte os números se negativos para positivos

// Estruturas de seleção Simples
if (N1 < 0)
    N1 = N1 * (-1);

if (N2 < 0)
    N2 = N2 * (-1);

// Estrutura de Seleção Composta
if (N1 > N2) // Utilize chaves para mais de uma instrução (blocos de
instruções)
{
    RESULTADO = N1 - N2;
    // Exibe o resultado da diferença do Maior pelo Menor
    Console.WriteLine("\t\tA diferença de {0} por {1} é: {2}", N1, N2,
RESULTADO);
    Console.WriteLine(); // Adiciona uma linha em branco
}
else
{
    RESULTADO = N2 - N1;
    // Exibe o resultado da diferença do Maior pelo Menor
    Console.WriteLine("\t\tA diferença de {0} por {1} é: {2}", N2, N1,
RESULTADO);
    Console.WriteLine(); // Adiciona uma linha em branco
}

} // Fim do método Main

} // Fim da Classe Números
}

```

04.07 – Calcula média de aluno

Calcular a Média de um aluno e verificar se o mesmo foi aprovado ou reprovado.

```

using System;

namespace Condição
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Alunos
    {
        /// <summary>

```

```

/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main(string[] args)
{
    // Declaração de Variáveis

    double N1, N2, N3, N4, MEDIA; // Notas escolares de um aluno e sua média

    // Solicita e lê as notas escolares de um aluno
    Console.WriteLine("\t\tInforme a 1º nota do aluno: ");
    N1 = Double.Parse(Console.ReadLine());
    Console.WriteLine(); // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme a 2º nota do aluno: ");
    N2 = Double.Parse(Console.ReadLine());
    Console.WriteLine(); // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme a 3º nota do aluno: ");
    N3 = Double.Parse(Console.ReadLine());
    Console.WriteLine(); // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme a 4º nota do aluno: ");
    N4 = Double.Parse(Console.ReadLine());
    Console.WriteLine(); // Adiciona uma linha em branco

    // Calcula a Média do Aluno

    MEDIA = (N1 + N2 + N3 + N4)/4;

    // Verifica se o aluno foi aprovado ou não

    if (MEDIA >=7) // Estrutura de Seleção Composta
        Console.WriteLine("\t\tO aluno foi aprovado e sua média é: {0}", MEDIA);
    else
        Console.WriteLine("\t\tO aluno foi reprovado e sua média é: {0}",
MEDIA);

    } // Fim do método Main

} // Fim da classe Alunos
}

```

04.08 – Lê números e compara valores

Efetuar a leitura de 3 números e compara o valor do 1º número com uma tabela e conforme a situação altera o novo valor dos outros 2 números conforme as condições de existência.

```

using System;

namespace Comparação
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>

```

```

class Números
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main(string[] args)
    {
        // Declaração de Variáveis

        double N,      X,      Y; // Números a ser fornecido pelo usuário
                                   // Double é o tipo da variável como (real,
ponto flutuante).....

        // Solicita e lê os números fornecidos pelo usuário
        Console.WriteLine("\t\tInforme o 1º número: ");
        N = Double.Parse(Console.ReadLine());
        Console.WriteLine();    // Adiciona uma linha em branco

        Console.WriteLine("\t\tInforme o 2º número (X): ");
        X = Double.Parse(Console.ReadLine());
        Console.WriteLine();    // Adiciona uma linha em branco

        Console.WriteLine("\t\tInforme o 3º número (Y): ");
        Y = Double.Parse(Console.ReadLine());
        Console.WriteLine();    // Adiciona uma linha em branco

        // Condições de existência
        /*
        Valores de X e Y
        *
        *
        *
        *
        */
        // Verifica a condição de Existência
        if (N <= 5)    // Estrura de Seleção Composta IF / ELSE
            // Utilize chaves para mais de uma instrução (blocos de instruções)
            {
                X = 0;
                Console.WriteLine("\t\tO novo valor de X é: {0}", X);    // O caractere de
escape "\t" adiciona tabulações
                Console.WriteLine();    // Adiciona uma linha em branco
                Y = 0;
                Console.WriteLine("\t\tO novo valor de Y é: {0}", Y);    // O caractere de
escape "\t" adiciona tabulações
                Console.WriteLine();    // Adiciona uma linha em branco
            }
        else
        {
            if (N <= 10)
            {
                X = X * 3;
                Console.WriteLine("\t\tO novo valor de X é: {0}", X);    // O
caractere de escape "\t" adiciona tabulações

```

```

        Console.WriteLine();    // Adiciona uma linha em branco
        Y = Y + 3;
        Console.WriteLine("\t\tO novo valor de Y é: {0}", Y);    // O
caractere de escape "\t" adiciona tabulações
        Console.WriteLine();    // Adiciona uma linha em branco
    }
    else
    {
        if (N <= 15)
        {
            X = X * 4;
            Console.WriteLine("\t\tO novo valor de X é: {0}", X);    //
O caractere de escape "\t" adiciona tabulações
            Console.WriteLine();    // Adiciona uma linha em branco
            Y = Y + 1;
            Console.WriteLine("\t\tO novo valor de Y é: {0}", Y);    //
O caractere de escape "\t" adiciona tabulações
            Console.WriteLine();    // Adiciona uma linha em branco
        }
        else
        {
            X = X * 3;
            Console.WriteLine("\t\tO novo valor de X é: {0}", X);    //
O caractere de escape "\t" adiciona tabulações
            Console.WriteLine();    // Adiciona uma linha em branco
            Y = Y + 2;
            Console.WriteLine("\t\tO novo valor de Y é: {0}", Y);    //
O caractere de escape "\t" adiciona tabulações
            Console.WriteLine();    // Adiciona uma linha em branco
        }
    }
}

    } // fim do método Main

} // Fim da Classe Números
}

```

04.09 – Verifica se aluno foi aprovado

Verificar se o aluno foi aprovado ou não, e se não for aprovado verificar sua nota de exame, emitindo o conceito para cada nota...

```

using System;

namespace Alunos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Notas
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]

```

```

static void Main(string[] args)
{
    // Declaração de Variáveis
    double N1, N2, N3,    // Notas escolares de um aluno
           NME,           // Nota de Média dos exercícios resolvidos
           MA;            // Média de Aproveitamento

    // Condições de Aprovação e Reprovação do Aluno
    // -----> "//" é comentário de uma única linha
    /* -----> "//" é comentário de uma múltiplas linhas */

    /*
        Média de Aproveitamento

    Conceito
        A
        B
        C
        D
        E

    */

    // Solicita e lê as notas de um aluno
    Console.WriteLine("\t\tInforme a 1ª nota do aluno: ");    // O caractere de escape
    "\t" adiciona tabulações
    N1 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme a 2ª nota do aluno: ");    // O caractere de escape
    "\t" adiciona tabulações
    N2 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme a 3ª nota do aluno: ");    // O caractere de escape
    "\t" adiciona tabulações
    N3 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme a Nota Média dos Exercícios: ");
    NME = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    // Calcula a Média de Aproveitamento do Aluno

    MA = (N1 + N2 + N3 + NME) / 4;

    // Verifica se o aluno foi aprovado ou reprovado, seu conceito.
    if (MA < 4)
        // Utilize chaves para mais de uma instrução (blocos de instruções)
        {
            Console.WriteLine("\t\tAluno Reprovado, Conceito = E");
            Console.WriteLine("\t\tA sua Média de Aproveitamento é: {0}", MA);
            Console.WriteLine();    // Adiciona uma linha em branco
        }
    else
    {
        if ((MA >= 4) && (MA < 6))
    
```



```

        {
            Console.WriteLine("\t\tAluno Reprovado, Conceito = D");
            Console.WriteLine("\t\tA sua Média de Aproveitamento é: {0}",
MA);

            Console.WriteLine();    // Adiciona uma linha em branco
        }
        else
        {
            if ((MA >= 6) && (MA < 7.5))
            {
                Console.WriteLine("\t\tAluno Aprovado, Conceito = C");
                Console.WriteLine("\t\tA sua Média de Aproveitamento é:
{0}", MA);

                Console.WriteLine();    // Adiciona uma linha em branco
            }
            else
            {
                if ((MA >= 7.5) && (MA < 9.0))
                {
                    Console.WriteLine("\t\tAluno Aprovado, Conceito
= B");
                    Console.WriteLine("\t\tA sua Média de
Aproveitamento é: {0}", MA);
                    Console.WriteLine();    // Adiciona uma linha em
branco
                }
            }
            else
            {
                Console.WriteLine("\t\tAluno Aprovado, Conceito
= A");
                Console.WriteLine("\t\tA sua Média de
Aproveitamento é: {0}", MA);
                Console.WriteLine();    // Adiciona uma linha em
branco
            }
        }
    }
}

} // Fim do método Main

} // Fim da Classe Notas
}

```


Exercícios 05 – Estrutura de Seleção IF e IF / ELSE

05.01 Soma números e compara resultado

Este programa tem como finalidade mostrar como declarar variáveis e utilizar estruturas de seleção simples. Efetua a soma de dois números e compara a soma, e se este for maior que 10 apresenta uma mensagem na console

```
using System;

namespace Adição
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double      N1, N2, // Números a serem fornecidos pelo usuário
            SOMA; // Armazenará a soma dos números N1 e N2

            // Solicita e lê os números fornecidos pelo usuário
            Console.WriteLine("\tInforme o 1º número: ");
            N1 = Double.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            Console.WriteLine("\tInforme o 2º número: "); // Os caracteres de escape "\t"
            adicionam duas tabulações
            N2 = Double.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            // Efetua a adição de N1 com N2
            SOMA = N1 + N2;

            // Adiciona uma linha em branco
            Console.WriteLine();
            // Efetua a comparação da soma e se o resultado for maior que 10 apresenta
            uma mensagem na console
            if ( SOMA > 10) // Estrutura de Seleção Simples
                Console.WriteLine("\tA soma dos valores {0} + {1} = {2}", N1, N2,
                SOMA);

            Console.WriteLine(); // Adiciona uma linha em branco

        } // Fim do método Main
    }
}
```

```

    } // Fim da Classe Números
}

```

05.02 Reajusta salário

Demonstra como utilizar estruturas de seleção if / else aninhadas, operadores relacionais, blocos de comando. Este programa tem como finalidade calcular o reajuste salarial de um funcionário.

```
using System;
```

```
namespace Funcionário
```

```
{
```

```
    /// <summary>
```

```
    /// Summary description for Class1.
```

```
    /// </summary>
```

```
    class Salário
```

```
    {
```

```
        /// <summary>
```

```
        /// The main entry point for the application.
```

```
        /// </summary>
```

```
        [STAThread]
```

```
        static void Main(string[] args)
```

```
        {
```

```
            // Declaração de Variáveis
```

```
            double salario_atual,          // O salário atual do funcionário
```

```
                percent_aumento,        // O percentual de aumento do salário
```

```
                novo_salario;           // O novo salário do funcionário
```

```
            int          cod_funcionario;    // Número que identifica o funcionário
```

```
            // Solicita e lê o código do Funcionário
```

```
            Console.Write("Informe o código do Funcionário: ");
```

```
            cod_funcionario = Int32.Parse(Console.ReadLine());
```

```
            Console.WriteLine();    // Adiciona uma linha em branco
```

```
            // Solicita e lê o salário do Funcionário
```

```
            Console.Write("Informe o valor do salário do Funcionário: ");
```

```
            salario_atual = Double.Parse(Console.ReadLine());
```

```
            Console.WriteLine();    // Adiciona uma linha em branco
```

```
            /*Condições para Reajuste do Salário do Funcionário
```

```
            *
```

```
            Salário Atual
```

```
            Índice de
```

```
Aumento
```

```
R$ 0,00 - R$ 300,00
```

```
10%
```

```
R$ 300,01 - R$ 600,00
```

```
11%
```

```
R$ 600,01 - R$ 900,00
```

```
12%
```

	R\$ 900,01 - R\$1500,00
6%	
	R\$1500,01 - R\$2000,00
3%	
	ACIMA DE R\$ 2000,00
sem aumento	

```

*/

// Utilize chaves para mais de uma instrução (blocos de instruções)
if (salario_atual <= 300)
{
    percent_aumento = (salario_atual * 0.10);
    novo_salario = (salario_atual + percent_aumento);
    Console.WriteLine("O funcionário {0}, teve um aumento de R$ {1}, e
    agora seu salário é: R${2}", cod_funcionario, percent_aumento, novo_salario);
    Console.WriteLine();
}
else
{
    if (salario_atual <= 600)
    {
        percent_aumento = (salario_atual * 0.11);
        novo_salario = (salario_atual + percent_aumento);
        Console.WriteLine("O funcionário {0}, teve um aumento de R$
        {1}, e agora seu salário é: R${2}", cod_funcionario, percent_aumento, novo_salario);
        Console.WriteLine();
    }
    else
    {
        if (salario_atual <= 900)
        {
            percent_aumento = (salario_atual * 0.12);
            novo_salario = (salario_atual + percent_aumento);
            Console.WriteLine("O funcionário {0}, teve um aumento
            de R$ {1}, e agora seu salário é: R${2}", cod_funcionario, percent_aumento, novo_salario);
            Console.WriteLine();
        }
        else
        {
            if (salario_atual <= 1500)
            {
                percent_aumento = (salario_atual * 0.06);
                novo_salario = (salario_atual +
                percent_aumento);
                Console.WriteLine("O funcionário {0}, teve um
                aumento de R$ {1}, e agora seu salário é: R${2}", cod_funcionario, percent_aumento, novo_salario);
                Console.WriteLine();
            }
            else
            {
                if (salario_atual <= 2000)
                {
                    percent_aumento = (salario_atual *
                    0.03);
                    novo_salario = (salario_atual +
                    percent_aumento);

```



```

        // Verifica se o número é positivo ou negativo
        // Utilize chaves para mais de uma instrução (blocos de instruções)
        if (N > 0)
        {
            X = N;
            Console.WriteLine("\tO valor agora de {0} agora é: {1}", N, X);
            Console.WriteLine();    // Adiciona uma linha em branco
        }
        else
        {
            X = -N;
            Console.WriteLine("\tO valor agora de {0} agora é: {1}", N, X);
            Console.WriteLine();    // Adiciona uma linha em branco
        }

    } // Fim do método Main

} // Fim da Classe Módulo
}

```

05.04 – Acha intervalo de número

Este programa utiliza conceitos de estruturas de seleção composta encadeadas, operadores relacionais, operadores lógicos e blocos de comando. O objetivo deste programa é achar o intervalo ao qual um número pertence e verificar se o mesmo pertence ao intervalo [A,B] ou somente ao intervalo [C,D] ou se pertence a ambos ou se não pertence a nenhum dos dois.

```

using System;

namespace Intervalos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Extremidades
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            double N,    // Um número qualquer que será fornecido pelo usuário

                        A,    // Extremidade de um intervalo [A,B]
                        B,    // Extremidade de um intervalo [A,B]
                        C,    // Extremidade de um intervalo [C,D]
                        D;    // Extremidade de um intervalo [C,D]

            /* Condições de Existência

```

```

A < B respectivamente
A < C respectivamente
C < D respectivamente

*/

// Solicita e lê o número do usuário
Console.WriteLine("\t\tPor favor informe o número: ");
N = Double.Parse(Console.ReadLine());
Console.WriteLine();    // Adiciona uma linha em branco

// Solicita e lê os valores dos intervalos

Console.WriteLine("\t\tPor favor informe o valor da extremidade A: ");
A = Double.Parse(Console.ReadLine());
Console.WriteLine();    // Adiciona uma linha em branco

Console.WriteLine("\t\tPor favor informe o valor da extremidade B: ");
B = Double.Parse(Console.ReadLine());
Console.WriteLine();    // Adiciona uma linha em branco

Console.WriteLine("\t\tPor favor informe o valor da extremidade C: ");
C = Double.Parse(Console.ReadLine());
Console.WriteLine();    // Adiciona uma linha em branco

Console.WriteLine("\t\tPor favor informe o valor da extremidade D: ");
D = Double.Parse(Console.ReadLine());
Console.WriteLine();    // Adiciona uma linha em branco

// Verifica as condições de existência
// Estrutura de Seleção Composta Encadeadas
if ((A < B) && (C < D) && (A < C))    // Utilize chaves para mais de uma
instrução (blocos de instruções)
{
    if ((N < A) || (N > D))    // Operador Lógico Ou...
    {
        Console.WriteLine("\t\tO número {0} não pertence ao intervalo:
[1],2] *** [3],4] ", N, A, B, C, D);
        Console.WriteLine();    // Adiciona uma linha em branco
    }
    else
    {
        if ((N >= A) && (N <= B) && (N >= C) && (N <= D))    //
Operador Lógico E..
        {
            Console.WriteLine("\t\tO número {0} pertence aos
intervalos: [1],2] *** [3],4] ", N, A, B, C, D);
            Console.WriteLine();    // Adiciona uma linha em branco
        }
        else
        {
            if ((N >= A) && (N <= B))
            {
                Console.WriteLine("\t\tO número {0} pertence ao
intervalo: [1],2]", N, A, B);

```



```

        Console.WriteLine(); // Adiciona uma linha em
branco
    }
    else
    {
        Console.WriteLine("\t\tO número {0} pertence ao
intervalo: [{1},{2}]", N, C, D);
        Console.WriteLine(); // Adiciona uma linha em
branco
    }
}
}
}
else
{
    Console.WriteLine("\t\tNúmeros inválidos!");
    Console.WriteLine(); // Adiciona uma linha em branco
}

} // Fim do método Main

} // Fim da Classe Extremidades
}

```

05.05 – Soma funções

Este programa utiliza estrutura de seleção composta (if / else) aninhadas, método Math.Pow para calcular a exponencial de um número qualquer, operadores relacionais. O objetivo é calcular a soma de duas funções como $F(N)$ e $G(N)$ atribuindo o resultado a Y.

```

using System;

namespace Funções
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Cálculos
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            double N, Y, H, F, G;

            // Solicita e lê um número qualquer do usuário
            Console.Write("\t\tPor favor informe um número: ");
            N = Double.Parse(Console.ReadLine());

```

```

        Console.WriteLine();    // Adiciona uma linha em branco

        // Efetua o cálculo da função  $H(N) = (N^2) - 16$ 
        H = (Math.Pow(N,2) - 16);

        // Estrutura de Seleção Composta
        // Efetua o cálculo da função F comparando o valor da função H
        if (H >= 0)
            F = H;
        else
            F = 1;

        // Estrutura de Seleção
        // Efetua o cálculo da função G comparando o valor da função F
        if (F == 0)
            G = (Math.Pow(N,2)+16);
        else
            G = 0;

        // Efetua a soma das funções  $F(N) + G(N)$ 
        Y = F + G;

        // Exibe o resultado das funções
        Console.WriteLine("\t\tA função H tem como valor: {0}", H);
        Console.WriteLine();    // Adiciona uma linha em branco
        Console.WriteLine("\t\tA função F tem como valor: {0}", F);
        Console.WriteLine();    // Adiciona uma linha em branco
        Console.WriteLine("\t\tA função G tem como valor: {0}", G);
        Console.WriteLine();    // Adiciona uma linha em branco
        Console.WriteLine("\t\tA soma das funções  $F({0}) + G({1})$  é: {2}", F, G, Y);
        Console.WriteLine();    // Adiciona uma linha em branco

    } // Fim do Método Main

} // Fim da classe Cálculos
}

```

05.06 – Recebe números e verifica divisores

Este programa tem por finalidade apresentar conceitos de estruturas de seleção, utilização do operador módulo e operadores de igualdade. Recebe quatro números do usuário e verifica quais são divisíveis por 2 e por 3.

```

using System;

namespace Divisão
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>

```

```

/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main(string[] args)
{
    // Declaração de Variáveis

    double N1, N2, N3, N4;          // Números informados pelo usuário

    // Solicita e lê os números
    Console.WriteLine("\t\tInforme o 1º número: ");
    N1 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme o 2º número: ");
    N2 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme o 3º número: ");
    N3 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme o 4º número: ");
    N4 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    // Adiciona uma linha em branco
    Console.WriteLine();

    // Estruturas de Seleção Simples
    if ((N1 % 2 == 0) && (N1 % 3 == 0))          // Utiliza o operador módulo para
pegar o resto da divisão "%"
        Console.WriteLine("O número: {0} é divisível por 2 e por 3", N1);

    if ((N2 % 2 == 0) && (N2 % 3 == 0))
        Console.WriteLine("O número: {0} é divisível por 2 e por 3", N2);

    if ((N3 % 2 == 0) && (N3 % 3 == 0))
        Console.WriteLine("O número: {0} é divisível por 2 e por 3", N3);

    if ((N4 % 2 == 0) && (N4 % 3 == 0))
        Console.WriteLine("O número: {0} é divisível por 2 e por 3", N4);

    // Exibe os números digitados
    Console.WriteLine("Os números informados foram: {0}, {1}, {2}, {3}", N1, N2, N3,
N4);

    Console.WriteLine();    // Adiciona uma linha em branco

} // Fim do método Main

} // Fim da Classe Números
}

```

05.07 – Verifica lados de um triângulo

Este programa tem por finalidade ensinar a utilização de estruturas de seleção aninhadas e encadeadas. Operadores relacionais, operadores lógicos, Método Pow, Declaração de Variáveis. O objetivo do programa é determinar se os lados fornecidos formam um triângulo, e ordenar os mesmos em ordem decrescente.

```
using System;
```

```
namespace TTriângulos
```

```
{
```

```
    /// <summary>
```

```
    /// Summary description for Class1.
```

```
    /// </summary>
```

```
    class Tipos
```

```
    {
```

```
        /// <summary>
```

```
        /// The main entry point for the application.
```

```
        /// </summary>
```

```
        [STAThread]
```

```
        static void Main(string[] args)
```

```
        {
```

```
            // Declaração de Variáveis
```

```
            double A, B, C, // Os lados de um triângulo qualquer
```

```
                MAIOR, // O lado maior de um triângulo qualquer
```

```
                MEDIO, // O lado médio de um triângulo qualquer
```

```
                MENOR; // O lado menor de um triângulo qualquer
```

```
            // Inicialização de Variáveis
```

```
            MAIOR = 0;
```

```
            MEDIO = 0;
```

```
            MENOR = 0;
```

```
            // Solicita e Lê as medidas do lado do triângulo
```

```
            Console.Write("Por favor informe a 1ª medida do lado do triângulo: ");
```

```
            A = Double.Parse(Console.ReadLine());
```

```
            Console.Write("Por favor informe a 2ª medida do lado do triângulo: ");
```

```
            B = Double.Parse(Console.ReadLine());
```

```
            Console.Write("Por favor informe a 3ª medida do lado do triângulo: ");
```

```
            C = Double.Parse(Console.ReadLine());
```

```
            // Exibe uma linha em branco
```

```
            Console.WriteLine();
```

```
            // Verifica qual é o maior dos lados, o médio e o menor
```

```
            if ((A >= B) && (A >= C))
```

```
            {
```

```
                MAIOR = A;
```

```
                if (B >= C)
```

```
                {
```

```

        MEDIO = B;
        MENOR = C;
    }
    else
    {
        MEDIO = C;
        MENOR = B;
    }
    Console.WriteLine("O lado maior tem medida de: {0}",MAIOR);
    Console.WriteLine("O lado médio tem medida de: {0}",MEDIO);
    Console.WriteLine("O lado menor tem medida de: {0}",MENOR);
}
else
{
    if ((B >= A) && (B >= C))
    {
        MAIOR = B;

        if (A >= C)
        {
            MEDIO = A;
            MENOR = C;
        }
        else
        {
            MEDIO = C;
            MENOR = A;
        }
        Console.WriteLine("O lado maior tem medida de: {0}",MAIOR);
        Console.WriteLine("O lado médio tem medida de: {0}",MEDIO);
        Console.WriteLine("O lado menor tem medida de: {0}",MENOR);
    }
    else
    {
        if ((C >= A) && (C >= B))
        {
            MAIOR = C;
            if (A >= B)
            {
                MEDIO = A;
                MENOR = B;
            }
            else
            {
                MEDIO = B;
                MENOR = A;
            }
            Console.WriteLine("O lado maior tem medida de:
{0}",MAIOR);

            Console.WriteLine("O lado médio tem medida de:
{0}",MEDIO);

            Console.WriteLine("O lado menor tem medida de:
{0}",MENOR);
        }
    }
}

```

```

        // Exibe uma linha em branco
        Console.WriteLine();

        // Verifica se as medidas dos três lados formam um triângulo
        if (((MAIOR) > ((MEDIO) + (MENOR))))
            Console.WriteLine("Os lados não formam um triângulo!");
        else
        {
            // Verifica se o triângulo é Retângulo
            if ((Math.Pow(MAIOR,2)) == ((Math.Pow(MEDIO,2)) +
(Math.Pow(MENOR,2))))
                Console.WriteLine("Triângulo Retângulo!");
            else
            {
                // Verifica se o triângulo é Obtusângulo
                if ((Math.Pow(MAIOR,2)) > ((Math.Pow(MEDIO,2)) +
(Math.Pow(MENOR,2))))
                    Console.WriteLine("Triângulo Obtusângulo!");
                else
                {
                    if((Math.Pow(MAIOR,2)) < ((Math.Pow(MEDIO,2)) +
(Math.Pow(MENOR,2))))
                        Console.WriteLine("Triângulo Acutângulo!");
                }
            }
        }

        // Verifica qual o tipo de triângulo
        if((MAIOR == MEDIO) && (MEDIO == MENOR))
            Console.WriteLine("Triângulo Equilátero!");
        else
        {
            if ((MAIOR == MEDIO) || ((MAIOR == MENOR) || (MEDIO == MENOR)))
                Console.WriteLine("Triângulo Isósceles!");
            else
            {
                Console.WriteLine("Triângulo Escaleno!");
            }
        }

    } // Fim do método Main

} // Fim da Classe Tipos
}

```

Exercícios 06 – Estrutura de Seleção IF e IF / ELSE

06.01 – Verifica nome e sexo de pessoa

Demonstra de forma simples como declarar variáveis do tipo string, e o uso de estruturas de seleção, operadores lógicos. O programa irá efetuar a leitura do sexo e do nome de uma pessoa...

```
using System;

namespace Pessoa
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Dados
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            string nome, //Nome de uma pessoa
                sexo;    //Sexo de uma pessoa

            //Solicita e lê os Dados da Pessoa
            Console.Write("Informe o seu nome: ");
            nome = Console.ReadLine();
            Console.WriteLine();    // Adiciona uma linha em branco

            Console.Write("Informe o seu sexo: ");
            sexo = Console.ReadLine();
            Console.WriteLine();    // Adiciona uma linha em branco

            // Verifica o sexo da pessoa
            // Estrutura de Seleção Composta IF / ELSE

            if ((sexo == "masculino") || (sexo == "MASCULINO") || (sexo == "Masculino"))
                Console.WriteLine("Ilmo Sr. {0}", nome);
            else
                Console.WriteLine("Ilma Sra. {0}", nome);

        } // Fim do método Main
    } // Fim da Classe Dados
}
```

06.02 – Compara soma de números

Utiliza estrutura de seleção composto if / else aninhada. Este programa compara a soma de dois números e se o valor for maior que 10 incrementa a soma em 5. Se o valor da soma for menor que 10, decrementa a soma em 7.

using System;

namespace Soma

{

/// <summary>

/// Summary description for Class1.

/// </summary>

class Adição

{

/// <summary>

/// The main entry point for the application.

/// </summary>

[STAThread]

static void Main(string[] args)

{

 // Declaração de Variáveis

 double N1, N2, // Números que serão fornecidos pelo usuário
 SOMA; // Soma os números N1 e N2

 // Solicita e lê os números fornecidos pelo usuário

 Console.Write("Informe o 1º número: ");

 N1 = Double.Parse(Console.ReadLine());

 Console.WriteLine(); // Adiciona uma linha em branco

 Console.Write("Informe o 2º número: ");

 N2 = Double.Parse(Console.ReadLine());

 Console.WriteLine(); // Adiciona uma linha em branco

 // Efetua a soma de N1 e N2

 SOMA = N1 + N2;

 // Compara o resultado da soma e apresenta uma mensagem na tela conforme a

condição

 // Utilize chaves para mais de uma instrução (blocos de instruções)

 if (SOMA >= 10)

 {

 SOMA = SOMA + 5; //Poderia ser escrito ---> SOMA += 5;

 Console.WriteLine("O novo valor da soma agora é: {0}", SOMA);

 Console.WriteLine(); // Adiciona uma linha em branco

 }

 else

 {

 SOMA = SOMA - 7; //Poderia ser escrito ---> SOMA -= 5;

 Console.WriteLine("O novo valor da soma agora é: {0}", SOMA);

 Console.WriteLine(); // Adiciona uma linha em branco

 }


```

        } // Fim do método Main

    } // Fim da classe Adição
}

```

06.03 – Extraí raízes quadradas

Utiliza estruturas de seleção composta encadeadas, método Math.Pow para calcular exponencial e blocos de comando. Este programa tem por finalidade extrair as raízes de uma equação do 2º Grau.

```

using System;

namespace Segundograu
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Equações
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            double A, B, C, // Coeficientes da equação do 2º grau (a*x**2 + b*x + c)
                DELTA, // O Valor de (b**2 - 4*a*c)
                R1, R2; // Raízes da equação do 2º grau

            // Solicita e lê os coeficientes da Equação do 2º grau
            Console.WriteLine("\tInforme o valor de A: ");
            A = Double.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            Console.WriteLine("\tInforme o valor de B: ");
            B = Double.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            Console.WriteLine("\tInforme o valor de C: ");
            C = Double.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            // Realiza o cálculo das raízes da Equação do 2º Grau

            // Verifica a condição de existência para ser uma equação do 2º grau
            /*
            * (A > 0) ou (A < 0)
            */

```

```

// Estruturas de Seleção Composta Encadeadas
// Utilize chaves para mais de uma instrução (blocos de instruções)
if ((A > 0) || (A < 0))    // Operadores relacionais e lógicos
{
    DELTA = (Math.Pow(B,2) - 4 * A * C);
    if (DELTA >= 0)
    {
        R1 = ((-B + (Math.Pow(DELTA,0.5)))/(2 * A));
        R2 = ((-B - (Math.Pow(DELTA,0.5)))/(2 * A));
        Console.WriteLine("\tAs raízes da equação são: {0},{1}", R1, R2);
        Console.WriteLine();    // Adiciona uma linha em branco
    }
    else
    {
        Console.WriteLine("\tA equação apresenta raízes
imaginárias!!!");

        Console.WriteLine();    // Adiciona uma linha em branco
    }
}
else
    Console.WriteLine("\tOs números fornecidos {0}, {1} e {2} não formam
uma equação do 2º grau!", A, B, C);

    Console.WriteLine();    // Adiciona uma linha em branco

} // Fim do método Main

} // Fim da classe Equações
}

```

06.04 – Compara valores

Este programa tem por finalidade utilizar estruturas de seleção composta encadeadas e operadores relacionais. Compara três valores fornecidos pelo usuário, e demonstra qual deles é o maior, médio e o menor

```

using System;

namespace Variáveis
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {

```

```

// Declaração de Variáveis
double A, B, C, // Números a ser fornecido pelo usuário
        MAIOR, MEDIO, MENOR; // Armazenará respectivamente o lado
maior, lado médio e menor lado

// Solicita e lê os números
Console.WriteLine("\t\tInforme o valor de A: ");
A = Double.Parse(Console.ReadLine());
Console.WriteLine(); // Adiciona uma linha em branco

Console.WriteLine("\t\tInforme o o valor de B: ");
B = Double.Parse(Console.ReadLine());
Console.WriteLine(); // Adiciona uma linha em branco

Console.WriteLine("\t\tInforme o o valor de C: ");
C = Double.Parse(Console.ReadLine());
Console.WriteLine(); // Adiciona uma linha em branco

// Verifica qual é o maior lado, o médio e o menor
// Utilize chaves para mais de uma instrução (blocos de instruções)
if ((A > B) && (A > C))
{
    MAIOR = A;
    Console.WriteLine("\t\tLado Maior é A: {0}", MAIOR);
    Console.WriteLine(); // Adiciona uma linha em branco
    if (B > C)
    {
        MEDIO = B;
        Console.WriteLine("\t\tLado Médio é B: {0}", MEDIO);
        Console.WriteLine(); // Adiciona uma linha em branco
        MENOR = C;
        Console.WriteLine("\t\tLado Menor é C: {0}", MENOR);
        Console.WriteLine(); // Adiciona uma linha em branco
    }
    else
    {
        MEDIO = C;
        Console.WriteLine("\t\tLado Médio é C: {0}", MEDIO);
        Console.WriteLine(); // Adiciona uma linha em branco
        MENOR = B;
        Console.WriteLine("\t\tLado Menor é B: {0}", MENOR);
        Console.WriteLine(); // Adiciona uma linha em branco
    }
}
else
{
    if ((B > A) && (B > C))
    {
        MAIOR = B;
        Console.WriteLine("\t\tLado Maior é B: {0}", MAIOR);
        Console.WriteLine(); // Adiciona uma linha em branco
        if (A > C)
        {
            MEDIO = A;
            Console.WriteLine("\t\tLado Médio é A: {0}", MEDIO);
            Console.WriteLine(); // Adiciona uma linha em branco

```

```

        MENOR = C;
        Console.WriteLine("\t\tLado Menor é C: {0}", MENOR);
        Console.WriteLine();    // Adiciona uma linha em branco
    }
    else
    {
        MEDIO = C;
        Console.WriteLine("\t\tLado Médio é C: {0}", MEDIO);
        Console.WriteLine();    // Adiciona uma linha em branco
        MENOR = A;
        Console.WriteLine("\t\tLado Menor é A: {0}", MENOR);
        Console.WriteLine();    // Adiciona uma linha em branco
    }
}
else
{
    if ((C > A) && (C > B))
    {
        MAIOR = C;
        Console.WriteLine("\t\tLado Maior é C: {0}", MAIOR);
        Console.WriteLine();    // Adiciona uma linha em branco
        if (A > B)
        {
            MEDIO = A;
            Console.WriteLine("\t\tLado Médio é A: {0}",
                A);
            Console.WriteLine();    // Adiciona uma linha em
                branco
            MENOR = B;
            Console.WriteLine("\t\tLado Menor é B: {0}",
                B);
            Console.WriteLine();    // Adiciona uma linha em
                branco
        }
        else
        {
            MEDIO = B;
            Console.WriteLine("\t\tLado Médio é B: {0}",
                B);
            Console.WriteLine();    // Adiciona uma linha em
                branco
            MENOR = A;
            Console.WriteLine("\t\tLado Menor é A: {0}",
                A);
            Console.WriteLine();    // Adiciona uma linha em
                branco
        }
    }
}
}
} // Fim do método Main
} // Fim da classe Números
}

```

06.05 – Calcula Imposto de Renda

Este programa tem por finalidade demonstrar a utilização de estruturas de seleção composta encadeadas, operadores relacionais e declaração de variáveis. O objetivo é calcular o imposto de renda de um contribuinte.

```
using System;

namespace Imposto
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Rendas
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            double salario_minimo, // Salário Mínimo do Contribuinte
                renda_mensal, // Renda Mensal do Contribuinte
                desconto, // Desconto a ser efetuado do Imposto
                renda_liquida, // É a Renda Mensal menos o Desconto
                imposto_renda; // Imposto de Renda do Contribuinte

            para cada dependente do Contribuinte
                int ndep; // Número de Dependentes do Contribuinte

            // Solicita e lê o valor do salário Mínimo do Contribuinte
            Console.WriteLine("\t\tInforme o valor do Salário Mínimo: ");
            salario_minimo = Double.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            // Solicita e lê o número de dependentes do Contribuinte
            Console.WriteLine("\t\tInforme a Quantidade de Dependentes: ");
            ndep = Int32.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            // Solicita e lê o valor da Renda Mensal do Contribuinte
            Console.WriteLine("\t\tInforme o valor da Renda Mensal do Contribuinte: ");
            renda_mensal = Double.Parse(Console.ReadLine());
            Console.WriteLine(); // Adiciona uma linha em branco

            // Efetua o cálculo do Desconto do Salário Mínimo por Dependente
            // O desconto por dependente será de 5% do salário mínimo
            desconto = (ndep * 0.05 * salario_minimo);
            Console.WriteLine("\t\tDesconto igual a: {0}", desconto);
            Console.WriteLine(); // Adiciona uma linha em branco

            // Efetua o cálculo da Renda Líquida
```

```

renda_liquida = (renda_mensal - desconto);
Console.WriteLine("\t\tRenda líquida igual a: {0}", renda_liquida);
Console.WriteLine(); // Adiciona uma linha em branco

/* Alíquotas
* Até 2 salários mínimos          Alíquota
* 2 a 3 salários mínimos          5%
* 3 a 5 salários mínimos          10%
* 5 a 7 salários mínimos          15%
* acima de 7 salários mínimos    20%
*/

// Estruturas de Seleção
// Compara a renda líquida para verificar qual o valor da alíquota e
// Calcular o Imposto de Renda

if ((renda_liquida) <= (salario_minimo*2)) // Utilize chaves para mais de
uma instrução (blocos de instruções)
{
    imposto_renda = 0;
    Console.WriteLine("\t\tO valor do imposto de renda do contribuinte é:
{0}", imposto_renda);
    Console.WriteLine(); // Adiciona uma linha em branco
}
else
{
    if ((renda_liquida) <= (salario_minimo*3))
    {
        imposto_renda = (renda_liquida*0.05);
        Console.WriteLine("\t\tO valor do imposto de renda do
contribuinte é: {0}", imposto_renda);
        Console.WriteLine(); // Adiciona uma linha em branco
    }
    else
    {
        if ((renda_liquida) <= (salario_minimo*5))
        {
            imposto_renda = (renda_liquida*0.10);
            Console.WriteLine("\t\tO valor do imposto de renda do
contribuinte é: {0}", imposto_renda);
            Console.WriteLine(); // Adiciona uma linha em branco
        }
        else
        {
            if ((renda_liquida) <= (salario_minimo*7))
            {
                imposto_renda = (renda_liquida*0.15);
                Console.WriteLine("\t\tO valor do imposto de
renda do contribuinte é: {0}", imposto_renda);
                Console.WriteLine(); // Adiciona uma linha em
branco
            }
            else
            {
                imposto_renda = (renda_liquida*0.20);

```

```

renda do contribuinte é: {0}", imposto_renda);
Console.WriteLine("\t\tO valor do imposto de
Console.WriteLine();    // Adiciona uma linha em
branco
    }
    }
}

} // Fim do método Main

} // Fim da Classe Rendas
}

```

06.06 – Reajusta salário

Utiliza estruturas de seleção encadeadas. Este programa tem por finalidade realizar o cálculo do reajuste salarial de um funcionário.

```

using System;

namespace Reajuste
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Funcionário
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            double          salario_inicial, // Salário inicial do funcionário
                           salario_final;   // Salário final do funcionário

            // Solicita e lê o salário do funcionário
            Console.Write("\t\tInforme o valor do salário do funcionário: ");
            salario_inicial = Double.Parse(Console.ReadLine());

            // Exibe uma linha em branco
            Console.WriteLine();

            /* Condições de Existência
            *          Salário Inicial (Atual)
            *          Reajuste
            *          salário menor ou igual a R$ 500,00
            *          15%
            *          salário maior ou igual a R$ 1000,00
            *          10%
            */

```

```

5%          *          salário maior que R$ 1000,00

          */

          // Verifica a condição de existência e calcula o novo salário
          // Utilize chaves para mais de uma instrução (blocos de instruções)

          if (salario_inicial <= 500)
          {
              salario_final = (salario_inicial + (salario_inicial * 0.15));
              Console.WriteLine("\t\tO novo salário do funcionário é igual a: {0}",
salario_final);

              Console.WriteLine();    // Adiciona uma linha em branco
          }
          else
          {
              if (salario_inicial <= 1000)
              {
                  salario_final = (salario_inicial + (salario_inicial * 0.10));
                  Console.WriteLine("\t\tO novo salário do funcionário é igual a:
{0}", salario_final);

                  Console.WriteLine();    // Adiciona uma linha em branco
              }
              else
              {
                  salario_final = (salario_inicial + (salario_inicial * 0.05));
                  Console.WriteLine("\t\tO novo salário do funcionário é igual a:
{0}", salario_final);

                  Console.WriteLine();    // Adiciona uma linha em branco
              }
          }
          } // Fim do método Main

          } // Fim da Classe Funcionário
}

```

06.07 – Verifica número

Este programa testa a estrutura de seleção composta if / else e o operador não lógico -> ! . Verifica um número fornecido pelo usuário e se o seu valor for maior que 5 executa as instruções.

```

using System;

namespace Testalógica
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>

```



```

[STAThread]
static void Main(string[] args)
{
    // Declaração de Variáveis
    double N1, N2, N3,    // Números que serão fornecidos pelo usuário
    C;                    // Armazenará o resultado dependendo do valor
do terceiro número

    // Solicita e lê os números fornecidos pelo usuário
    Console.WriteLine("\t\tInforme o 1º número: ");
    N1 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme o 2º número: ");
    N2 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    Console.WriteLine("\t\tInforme o 3º número: ");
    N3 = Double.Parse(Console.ReadLine());
    Console.WriteLine();    // Adiciona uma linha em branco

    // Efetua a comparação de N3

    // Operador não lógico * ! *

    if (!(N3 > 5))    // Utilize chaves para mais de uma instrução (blocos de
instruções)
    {
        C = (N1 + N2) * N3;
        // Exibe o resultado
        Console.WriteLine("\t\tO resultado é: {0}", C);
        Console.WriteLine();    // Adiciona uma linha em branco
    }
    else
    {
        C = (N1 - N2) * N3;
        // Exibe o resultado
        Console.WriteLine("\t\tO resultado é: {0}", C);
        Console.WriteLine();    // Adiciona uma linha em branco
    }

    } // Fim do método Main

} // Fim da classe Números
}

```

Exercícios 07 – Estrutura de Seleção While e Outras

07.01 – Calcula tabuada de número

Estrutura de repetição while. Calcula a tabuada de um número qualquer.

```
using System;

namespace Tabuada
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    NUM, CONT, TAB;

            // Inicialização de Variáveis
            CONT = 1;
            TAB = 0;

            // Solicita e lê o número fornecido pelo usuário
            Console.Write("Informe o número desejado: ");
            NUM = Int32.Parse(Console.ReadLine());
            Console.WriteLine();

            // Estrutura de repetição while executando a tabuada
            while ( CONT <=10)
            {
                TAB = NUM * CONT;
                Console.WriteLine("A Tabuada do número {0} X {1} é: {2}\n", NUM,
CONT, TAB);

                CONT = CONT + 1;
            }
        } // Fim do Método Main

    } // Fim da Classe Números
}
```

07.02 – Lê número e pede novos valores

Estrutura de repetição while. Este programa faz a leitura de um valor e continuará a pedir novos valores com base na decisão do usuário, o valor para continuar deverá ser "sim".

```
using System;

namespace Usuário
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Responde
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            double resultado, // Resultado da expressão algébrica
                n; // Número qualquer fornecido pelo
            usuário

            string resposta; // Sim ou Não

            resposta = "sim";
            while (resposta == "sim")
            {
                // Solicita e lê um número do usuário
                Console.Write("Informe um número qualquer: ");
                n = Double.Parse(Console.ReadLine());
                Console.WriteLine();

                // Efetua a expressão algébrica
                resultado = n * 3;

                // Exibe o resultado da expressão algébrica
                Console.WriteLine("O resultado do número {0} multiplicado por 3 é:
            {1}\n", n, resultado);

                // Pergunta ao usuário se ele deseja executar
                Console.Write("Digite " + "\"sim\"" + " para continuar ou pressione " +
            "\"qualquer tecla\"" + " para sair.: ");
                resposta = Console.ReadLine();
                Console.WriteLine();
            } // Fim do laço de repetição

        } // Fim do Método Main
    } // Fim da Classe
}
```

07.03 – Quadrado entre um intervalo

Utiliza estrutura de repetição while. Este programa tem por finalidade apresentar os quadrados compreendidos entre 1 a 20.

```
using System;

namespace Quadrados
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int cont,num;

            // Inicialização de Variáveis
            cont = 1;
            num = 0;

            while (cont <= 20)    // Utilize chaves para mais de uma instrução (blocos de
instruções)
            {
                num = cont * cont;
                Console.WriteLine("O quadrado do número {0} é: {1}", cont, num);
                cont = cont + 1;
            }
            } // Fim do método Main

        } // Fim da Classe Números
    }
```

07.04 – Determina idade e percentuais em pesquisa

Estrutura de repetição while e seleção composta. Este programa tem como finalidade determinar a maior idade fornecida em uma pesquisa numa certa região e calcular a porcentagem de mulheres que estão com idade entre 18 e 35 ((idade >= 18 && (idade <= 35)) e que tenham cabelos louros e olhos verdes com um número indeterminado de habitantes...

```
using System;
```

```

namespace Pesquisa
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Entrevistados
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int idade, // Idade do Entrevistado
            m_idade, // Maior idade fornecida de um entrevistado na
pesquisa
            c_pessoas, // Número de pessoas entrevistadas que
participaram da pesquisa
            c_mulheres, // Números de números entrevistas na pesquisa
cistf; // Número de pessoas do sexo feminino (que
estão nas condições de existência)

            string sexo, // sexo do entrevistado
            c_olhos, // Cor dos olhos do entrevistado
            c_cabelos; // Cor dos cabelos do entrevistado

            double pistf; // Porcentagem de pessoas do sexo feminino

            // Inicialização de Variáveis
            c_pessoas = 0;
            c_mulheres = 0;
            cistf = 0;
            m_idade = 0;

            /* Amostragem de uma pesquisa de uma população de uma certa região, a qual
coletou os
            * seguintes dados referentes a cada habitante para serem analisados:
            * sexo (masculino ou feminino)
            * cor dos cabelos (louros, pretos e castanhos)
            * cor dos olhos (azuis, verdes e castanhos)
            */

            /* Calcular:
            * A maior idade dos habitantes
            * A percentagem de indivíduos do sexo feminino cuja idade está entre 18 e 35
anos
            * inclusive e que tenham olhos verdes e cabelos louros
            * O final do conjunto de habitantes é reconhecido pelo valor -1 entrado com
idade...

            */
            // Solicita e lê a Idade de um Entrevistado
            Console.WriteLine("Informe a Idade do Entrevistado, -1 para Sair: ");
            idade = Int32.Parse(Console.ReadLine());
            Console.WriteLine();

```

fornecida

```
// Atribuindo a idade lida como sendo a maior idade, pois é a primeira idade
```

```
while (idade != -1)
```

```
{
```

```
    if (idade >= m_idade)
```

```
    {
```

```
        m_idade = idade;
```

```
    }
```

```
// Solicita e lê o sexo do entrevistado
```

```
Console.WriteLine("\tInforme o sexo do entrevistado: ");
```

```
sexo = Console.ReadLine();
```

```
// Verifica o total de Mulheres que participaram da pesquisa
```

```
if ((sexo == "Feminino") || (sexo == "FEMININO") || (sexo == "feminino"))
```

```
    c_mulheres = c_mulheres + 1;
```

```
// Solicita e lê a Cor dos Olhos do Entrevistado
```

```
Console.WriteLine("\tInforme a cor dos olhos do entrevistado: ");
```

```
c_olhos = Console.ReadLine();
```

```
// Solicita e lê a Cor dos Cabelos Entrevistado
```

```
Console.WriteLine("\tInforme a cor dos cabelos do entrevistado: ");
```

```
c_cabelos = Console.ReadLine();
```

```
Console.WriteLine();
```

```
// Calcula as mulheres nesta condição
```

```
/*
```

```
 * Porcentagem de indivíduos do sexo feminino cuja idade está entre 18
```

e 35 anos inclusive

```
 * e que tenham olhos verdes e cabelos louros
```

```
*/
```

```
if (((sexo == "FEMININO") || (sexo == "feminino") || (sexo == "Feminino"))
&& ((idade >= 18)&&(idade <= 35)) && ((c_olhos == "Verdes") || (c_olhos == "verdes") || (c_olhos ==
"VERDES")) && ((c_cabelos == "Louros") || (c_cabelos == "louros") || (c_cabelos == "LOUROS")))
```

```
{
```

```
    ciscf = ciscf + 1;
```

```
}
```

```
// Solicita e lê a Idade de um Entrevistado
```

```
Console.WriteLine("\tInforme a Idade do Entrevistado, -1 para Sair: ");
```

```
idade = Int32.Parse(Console.ReadLine());
```

```
Console.WriteLine();
```

```
// Calcula a quantidade de pessoas que participaram da pesquisa
```

```
c_pessoas = c_pessoas + 1;
```

```
} // Fim do laço while
```

// Verifica a porcentagem de mulheres que estão nesta condição entre o total de
mulheres entrevistadas

```
if (ciscf > 0)
```

```

        {
            pif = ((cifs * 100) / c_mulheres);
            Console.WriteLine();
            Console.WriteLine("\tMulheres que estão na condição especificadas são:
{0}%\n",pif);
        }
        // Exibe a maior idade encontrada
        if (m_idade == 0)
        {
            Console.WriteLine("\t\tNão foi entrevistado nenhuma pessoa!\n");
        }
        else
        {
            Console.WriteLine("O total de entrevistas foi de: {0}
pessoas",c_pessoas);
            Console.WriteLine("O total de pessoas do sexo feminino na pesquisa foi
de: {0} mulheres",c_mulheres);
            Console.WriteLine("A maior idade fornecida na pesquisa é de: {0}
anos",m_idade);
        }

    } // Fim do método Main

} // Fim da Classe Entrevistados
}

```

07.05 – Lê valores e verifica-os

Estrutura de repetição while. Lê 5 valores para um número qualquer, um de cada vez, e conta quantos destes valores são negativos.

```

using System;

namespace Negativos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    cont, neg;
            double num;

            // Inicialização de variáveis
            cont = 1;

```

```

        neg = 0;

        while( cont <=5)           // Utilize chaves para mais de uma instrução (blocos de
instruções)
        {
            Console.Write("\t\tInforme um número: ");
            num = Double.Parse(Console.ReadLine());
            Console.WriteLine();    // Imprime uma linha em branco
            if (num < 0)
                neg = neg + 1;

            cont = cont + 1;
        }

        Console.WriteLine("\t\tA quantidade números negativos é: {0}\n", neg);

    } // Fim do Método Main

} // Fim da Classe Números
}

```

07.06 – Gera números e verifica divisores

Estrutura de repetição while e operador módulo. Gera os números de 10 a 100 que são divididos por 11 onde o resto da divisão é igual a 5.

```

using System;

namespace Divididos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Onze
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    cont;

            // Inicialização de Variáveis
            cont = 10;

            // Utilize chaves para mais de uma instrução (blocos de instruções)
            while (cont <= 100)
            {
                if (cont % 11 == 5)
                    Console.WriteLine("O número {0} é divisível por 11", cont);
            }
        }
    }
}

```



```

        cont = cont + 1;
    }

    } // Fim do Método Main

    } // Fim da Classe Onze
}

```

07.07 – Termos de PA

Estrutura de repetição while. Este programa tem por finalidade calcular os termos de uma progressão aritmética de N elementos.

```

using System;

namespace Progressão
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Aritmética
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    A1,           // Primeiro termo da Progressão Aritmética
                  AN,           // Enésimo termo da Progressão Aritmética
                  N,             // Número de elementos de Progressão Aritmética
                  R,             // Razão da Progressão Aritmética
                  CONT; // Contador

            // Inicialização de Variáveis
            A1 = 0;
            CONT = 1;
            AN = 0;

            // Solicita e lê o números de elementos da PA
            Console.WriteLine("\tInforme o número de elementos da Progressão Aritmética: ");
            N = Int32.Parse(Console.ReadLine());

            // Solicita e lê a razão da PA
            Console.WriteLine("\tInforme a razão desta Progressão Aritmética: ");
            R = Int32.Parse(Console.ReadLine());

            while (N >= CONT )
            {
                AN = (A1 + ((N - 1)* R));
                Console.WriteLine("\tO {0} elemento da Progressão Aritmética é: {1}\n",
N, AN);
            }
        }
    }
}

```

```

        N = (N - CONT);
    }

    } // Fim do método Main

    } // Fim da Classe Aritmética
}

```

07.08 – Calcula média Geral de alunos

Estrutura de repetição while. Este programa tem por finalidade calcular a Média Geral de uma Turma de alunos.

```

using System;

namespace Turma
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Alunos
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    N_ALUNOS,    // Número de alunos
                CONT;          // Contador de alunos

            double N1, N2, N3, N4, // Notas de um aluno
                MEDIA, // Média de um aluno
                MEDIAT; // Média da turma de alunos

            // Inicialização de Variáveis
            CONT = 1;
            MEDIA = 0;
            MEDIAT = 0;

            // Solicita o número de alunos
            Console.WriteLine("Informe o número de alunos: ");
            N_ALUNOS = Int32.Parse(Console.ReadLine());
            Console.WriteLine();

            while (CONT <= N_ALUNOS)
            {
                // Solicita e lê as notas de um aluno
                Console.WriteLine("Informe a 1º nota do aluno: ");
                N1 = Double.Parse(Console.ReadLine());

                Console.WriteLine("Informe a 2º nota do aluno: ");
            }
        }
    }
}

```

```

        N2 = Double.Parse(Console.ReadLine());

        Console.WriteLine("\t\tInforme a 3º nota do aluno: ");
        N3 = Double.Parse(Console.ReadLine());

        Console.WriteLine("\t\tInforme a 4º nota do aluno: ");
        N4 = Double.Parse(Console.ReadLine());
        Console.WriteLine();

        // Calcula a Média do Aluno
        MEDIA = ((N1 + N2 + N3 + N4) / 4);
        Console.WriteLine("\t\tA Média do {0} aluno é: {1}\n", CONT, MEDIA);

        // Calcula a Média da Turma
        MEDIAT = ((MEDIAT + MEDIA) / CONT);

        CONT = CONT + 1;

    } // Fim do laço de Repetição

    // Exibe a média da turma
    Console.WriteLine("\t\tA Média geral da turma é: {0}\n", MEDIAT);
} // Fim do Método Main

} // Fim da Classe Alunos
}

```

Exercícios 08 – Estrutura de Repetição While e outras

08.01 – Apresenta valores em uma faixa de números

Este programa tem por finalidade explicar o funcionamento da estrutura de repetição While. Apresenta todos os valores numéricos ímpares situados na faixa de 0 a 20.

```
using System;

namespace Valores
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Impares
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    cont = -1;

            while (cont < 19)    // Repete o laço 10 vezes
            {
                cont = cont + 2;
                Console.WriteLine("\t\tOs valores ímpares menores que 20 são: {0}",
cont);

                Console.WriteLine();    // Imprime uma linha em branco
            }
            Console.WriteLine();    // Imprime uma linha em branco

        } // Fim do método Main

    } // Fim da classe Impares
}
```

08.02 – Calcula média de idades

Estrutura de repetição while e conversão para double (cast). Este programa tem por finalidade calcular a média de idades de um grupo de pessoas.

```
using System;

namespace Idade
```

```

{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Grupo
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    s_idade,      // soma das idades do grupo de pessoas
                  idade,       // idade de uma pessoa
                  c_pessoas;    // contador de pessoas

            double m_idade;     // média das idades

            // Inicialização de Variáveis
            m_idade = 0;
            s_idade = 0;
            c_pessoas = 1;

            // Solicita e lê a idade do Indivíduo
            Console.WriteLine("\t\tInforme a idade da pessoa: ");
            idade = Int32.Parse(Console.ReadLine());

            while (idade > 0 )
            {
                s_idade = s_idade + idade;
                m_idade = (double) s_idade / c_pessoas;
                c_pessoas = c_pessoas + 1;

                // Solicita e lê a idade do Indivíduo
                Console.WriteLine("\t\tInforme a idade da pessoa: ");
                idade = Int32.Parse(Console.ReadLine());

            } // Fim do laço de repetição

            // Apresenta a média de idades do grupo de pessoas
            Console.WriteLine("\n\t\tA média das idades fornecidas é: {0}\n", m_idade);

        } // Fim do método Main

    } // Fim da classe Grupo
}

```

08.03 – Calcula população de um país

Estruturas de repetição while. Calcula o tempo necessário para que um País A se iguale ou ultrapasse o número de habitantes de um País B.

```

using System;

namespace População
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Habitantes
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int cont_anos; // Número de anos para a população se igualar
            double popA, // Número de habitantes do País A
                popB; // Número de habitantes do País B

            // Inicialização de Variáveis
            popA = 90000000; // 90.000.000 habitantes
            popB = 120000000; // 120.000.000 habitantes
            cont_anos = 0;

            while(popA <= popB)
            {
                // População A com 3% de
                // População B com 1.5
                // de crescimento anual
                popA = (popA + (popA * 0.03));
                popB = (popB + (popB * 0.015));
                cont_anos = cont_anos + 1;
            }
            Console.WriteLine("O País A se igualou em número de habitantes ao País B em
{0} anos\n", cont_anos);

        } // Fim do método Main

    } // Fim da Classe Habitantes
}

```

08.04 – Média de uma turma de alunos

Estrutura de repetição while. Este programa calcula a média de uma turma de 20 alunos.

```

using System;

namespace Notas
{
    /// <summary>
    /// Summary description for Class1.

```

```

/// </summary>
class Alunos
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main(string[] args)
    {
        // Declaração de Variáveis
        int    cont;    // Contador de Alunos

        double n1, n2, n3, n4, // Notas escolares de um aluno
               media, // média de um aluno
               mediat; // Média de uma turma de 20 alunos

        // Inicialização de Variáveis

        cont = 0;
        media = 0;
        mediat = 0;
        while (cont < 20)    // Repete o laço 20 vezes
        {
            // Solicita e lê a nota de um aluno
            Console.WriteLine();
            Console.WriteLine("\t\tInforme a 1ª nota de um aluno: ");
            n1 = Double.Parse(Console.ReadLine());

            Console.WriteLine("\t\tInforme a 2ª nota de um aluno: ");
            n2 = Double.Parse(Console.ReadLine());

            Console.WriteLine("\t\tInforme a 3ª nota de um aluno: ");
            n3 = Double.Parse(Console.ReadLine());

            Console.WriteLine("\t\tInforme a 4ª nota de um aluno: ");
            n4 = Double.Parse(Console.ReadLine());
            Console.WriteLine();

            // Calcula a Média de um aluno
            media = (n1 + n2 + n3 + n4)/4;

            cont = cont + 1;

            // Exibe a Média do Aluno
            Console.WriteLine("\t\tA média do {0} aluno é: {1}", cont, media);

            // Calcula a Média da turma de alunos
            mediat = ((mediat + media)/cont);

        } // Fim do laço de repetição while

        // Exibe a Média da Turma de Alunos
        Console.WriteLine("\n\t\tA média da turma de {0} aluno(s) é: {1}\n", cont, mediat);

    } // Fim do método Main
}

```

```
    } // Fim da Classe Alunos  
}
```

08.05 – Soma dos primeiros números inteiros

Este programa tem por finalidade apresentar o funcionamento da estrutura de repetição While. Apresenta o total da soma obtido dos cem primeiros números inteiros.

```
using System;  
  
namespace Soma  
{  
    /// <summary>  
    /// Summary description for Class1.  
    /// </summary>  
    class Números  
    {  
        /// <summary>  
        /// The main entry point for the application.  
        /// </summary>  
        [STAThread]  
        static void Main(string[] args)  
        {  
            // Declaração de Variáveis  
            int    cont,          // Conta os números  
                soma;           // total dos cem números inteiros  
  
            // Inicialização de Variáveis  
            cont = 0;  
            soma = 0;  
            while ( cont < 100)  
            {  
                cont = cont + 1;  
                soma = soma + cont;  
            }  
  
            // Exibe o resultado  
            Console.WriteLine("\tA soma dos cem primeiros inteiros é: {0}", soma);  
        }  
    } // Fim do Método Main  
}  
} // Fim da Classe Números  
}
```

08.06 – Resultado de pesquisa

Estrutura de repetição while encadeada com if / else e aninhadas. Calcula os resultados de uma pesquisa de aceitação de um produto com base na resposta do entrevistado (sim ou não), sabendo-se que foram entrevistados 500 pessoas.

```
using System;
```

```
namespace Firma
```

```
{
```

```
    /// <summary>
```

```
    /// Summary description for Class1.
```

```
    /// </summary>
```

```
    class Pesquisa
```

```
    {
```

```
        /// <summary>
```

```
        /// The main entry point for the application.
```

```
        /// </summary>
```

```
        [STAThread]
```

```
        static void Main(string[] args)
```

```
        {
```

```
            // Declaração de Variáveis
```

da pesquisa	int	c_pessoas,	// Número de Pessoas que participaram
		nprs,	// Número de pessoas que responderam
sim		nprn,	// Número de pessoas que responderam
não		c_psf,	// Número de pessoas do sexo feminino
		c_psm,	// Número de pessoas do sexo masculino
responderam sim		c_psfrs,	// Número de pessoas do sexo feminino que
que responderam não		c_psmrn;	// Número de pessoas do sexo masculino
	double	ppsfrs,	// Porcentagem de pessoas do sexo feminino que
responderam sim		ppsmrn;	// Porcentagem de pessoas do
sexo masculino que responderam não			

```
            string  sexo,          // Sexo Masculino ou Feminino
```

```
                resposta;        // sim ou não
```

```
            // Inicialização de Variáveis
```

```
            c_pessoas = 0;
```

```
            nprs = 0;
```

```
            nprn = 0;
```

```
            c_psf = 0;
```

```
            c_psm = 0;
```

```
            c_psfrs = 0;
```

```
            c_psmrn = 0;
```

```
            ppsfrs = 0;
```

```
            ppsmrn = 0;
```

```
            while( c_pessoas < 5 )
```

```
            {
```

```
                // Solicita e lê o sexo do entrevistado
```

```

        Console.WriteLine("\t\tInforme o sexo do entrevistado: ");
        sexo = Console.ReadLine();

        // Solicita e lê a resposta do entrevistado
        Console.WriteLine("\t\tInforme a resposta do entrevistado: ");
        resposta = Console.ReadLine();
        Console.WriteLine();

        // Calcula o número de pessoas que responderam sim ou não
        if ((resposta == "SIM") || (resposta == "Sim") || (resposta == "sim") ||
(resposta == "S") || (resposta == "s"))
            nprs = nprs + 1;
        else
            nprn = nprn + 1;

        // Calcula o número de pessoas que pertencem ao sexo feminino ou
masculino
        if ((sexo == "FEMININO") || (sexo == "Feminino") || (sexo == "feminino"))
            c_psf = c_psf + 1;
        else
            c_psm = c_psm + 1;

        // Calcula o número de pessoas do sexo feminino que responderam sim
        if (((sexo == "FEMININO") || (sexo == "Feminino") || (sexo == "feminino")) &&
((resposta == "SIM") || (resposta == "Sim") || (resposta == "sim") || (resposta == "S") || (resposta == "s")))
            c_psfrs = c_psfrs + 1;

        // Calcula o número de pessoas do sexo masculino que responderam não
        if (((sexo == "MASCULINO") || (sexo == "Masculino") || (sexo ==
"masculino")) && ((resposta == "NÃO") || (resposta == "Não") || (resposta == "não") || (resposta == "N") ||
(resposta == "n")))
            c_psmrn = c_psmrn + 1;

        // Incrementa o contador de pessoas
        c_pessoas = c_pessoas + 1;

    } // Fim do laço de repetição while

    // Verifica se houve pessoas do sexo feminino na pesquisa
    if (c_psf > 0)
    {
        // Calcula a porcentagem de pessoas do sexo feminino que responderam
sim
        ppsfrs = ((c_psfrs * 100) / c_psf);
        Console.WriteLine("\t\tMulheres que responderam sim são: {0}%\n",
ppsfrs);
    }
    else
    {
        Console.WriteLine("\t\tNenhuma mulher participou da pesquisa!\n");
    }

    // Verifica se houve pessoas do sexo masculino na pesquisa
    if (c_psm > 0)
    {

```

```

// Calcula a porcentagem de pessoas do sexo masculino que
responderam não
ppsmrn = ((c_psmrn * 100)/c_psm);
Console.WriteLine("\t\tHomens que responderam não são: {0}%\n",
ppsmrn);
    }
    else
    {
        Console.WriteLine("\t\tNenhum homem participou da pesquisa!");
    }

// Exibe os resultados dos cálculos
Console.WriteLine("\t\tO total de pessoas que responderam sim é: {0}\n", nprs);
Console.WriteLine("\t\tO total de pessoas que responderam não é: {0}\n", nprn);

    } // Fim do método Main

} // Fim da Classe Pesquisa
}

```

08.07 – Potencias de um intervalo

Demonstra como utilizar estruturas de repetição e o método Pow. Apresenta as potências de 3 variando de 0 a 15.

```

using System;

namespace Potências
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            double      NUM, EXP, POT;

            // Inicialização de Variáveis
            EXP = 0;
            NUM = 3;

            // Fase de Processamento
            while( EXP <= 15)
            {
                POT = Math.Pow(NUM, EXP);
            }
        }
    }
}

```

```

        Console.WriteLine("\tA Potência de {0} elevado a {1} é: {2}\n", NUM, EXP,
POT);

        EXP = EXP + 1;

    }

} // Fim do Método Main

} // Fim da Classe Números
}

```

08.08 – Lê números em um intervalo

Utilização da estrutura de seleção simples **If** com estrutura de repetição While. Este programa tem por finalidade ler um número qualquer menor ou igual a 50 e apresentar o valor obtido da multiplicação sucessiva do número por 3 ($N * 3$) enquanto o produto for menor que 250.

```

using System;

namespace Multiplicação
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Sucessiva
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis

            int          N, PRODUTO;

            // Inicialização de Variáveis
            PRODUTO = 0;

            Console.Write("\t\tInforme um número: ");
            N = Int32.Parse(Console.ReadLine());

            if (N <= 50)    // Utilize chaves para mais de uma instrução (blocos de
instruções)
            {
                PRODUTO = N * 3;

                while(PRODUTO < 250)
                {
                    Console.WriteLine("\t\tO produto é: {0}", PRODUTO);
                    PRODUTO = PRODUTO * 3;
                }
            }
        }
    }
}

```

```
        }  
    } // Fim do Método Main  
  
} // Fim da classe Sucessiva  
}
```

Exercícios 09 – Estrutura de Repetição IF, ELSE, While e outras

09.01 – Utilizando operadores lógicos

Este programa tem por finalidade verificar o sexo de uma pessoa. Demonstra como utilizar operadores lógicos, relacionais e declarar strings.

```
using System;

namespace Sexo
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Pessoa
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            string sexo;

            // Solicita e lê o sexo de uma pessoa
            Console.WriteLine("\n\tInforme o seu sexo: ");
            sexo = Console.ReadLine();

            if ((sexo == "MASCULINO") || (sexo == "Masculino") || (sexo == "masculino") ||
                (sexo == "FEMININO") || (sexo == "Feminino") || (sexo == "feminino"))
            {
                Console.WriteLine("\n\tO seu sexo é válido!");
            }
            else
            {
                Console.WriteLine("\n\tO seu sexo é inválido!");
            }

            } // Fim do Método Main

        } // Fim da Classe Pessoa
    }
```

09.02 – Utilizando while, Match.Pow e cast in

Calcula o quadrado dos números entre 1 e 5. Demonstra como utilizar a estrutura de repetição while, método Math.Pow ou utilizar um cast int.

```

using System;

namespace Quadrados
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int cont, num; // Se utilizar o Método Math.Pow declare a variável como
                           // sendo do tipo double ou crie um cast do tipo
            (int)

            // Inicialização de Variáveis
            cont = 1;
            num = 0;

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");

            // Estrutura de repetição while
            while (cont <= 5)
            {
                num = cont * cont; // num = (int)Math.Pow(cont,2);

                // Exibe o resultado
                Console.WriteLine("\n\t\t\tO quadrado do número " + cont + " é: " + num);

                cont++; // Incrementa o contador em 1
            }

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");

            Console.WriteLine("\n\t\t\t\t\thttp://www.msdnbrasil.com.br");

            // Exibe uma linha em branco na tela
            Console.WriteLine();

            } // Fim do Método Main

        } // Fim da classe Números
    }

```

09.03 – Estruturas de seleção simples e repetição while

Este programa lê um número menor ou igual a 50 e apresenta o valor obtido da multiplicação do número por 3 sucessivamente enquanto o produto for menor que 250. Demonstra a utilização de estrutura de seleção simples, e estrutura de repetição while.

```
using System;

namespace Produto
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Numeros
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int n, produto;

            // Inicialização de Variáveis
            produto = 0;

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");

            // Solicita e lê um número do usuário
            Console.Write("\n\t\tInforme um número qualquer: ");
            n = Int32.Parse(Console.ReadLine());

            if (n <= 50)    // Estrutura de seleção simples
            {
                produto = n * 3;

                // Estrutura de repetição while
                while ( produto < 250)
                {
                    // Exibe o resultado na tela
                    Console.WriteLine("\n\t\tO produto é: {0}", produto);
                    produto *= 3;    // Poderia ser escrito (produto = produto * 3);
                }
            }

            // Exibe uma linha na tela
        }
    }
}
```



```

Console.WriteLine("\t\t_____");

        Console.WriteLine("\n\t\t\t\t\thttp://www.msdnbrasil.com.br");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do Método Main

} // Fim da classe Números
}

```

09.04 – Estrutura de repetição while e variáveis

Este programa calcula a média de uma turma de 10 alunos. Utiliza métodos de console, estrutura de repetição while e inicialização de variáveis.

```

using System;

namespace Classes
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Alunos
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int contador;

            double media, nota, total;

            // Inicialização de Variáveis
            total = 0;
            contador = 1;

            while (contador <= 10)
            {
                // Solicita e lê a nota do usuário
                Console.Write("\n\t\t\t\t\tEntre com a {0} nota do aluno: ", contador);
                nota = Double.Parse(Console.ReadLine());

                // Adiciona nota ao total
                total = total + nota;

                // Adiciona 1 ao contador
                contador = contador + 1;
            }
        }
    }
}

```

```

    }

    // Fase de Conclusão
    media = total / 10;

    // Exibe o resultado
    Console.WriteLine("\n\t\tClasse com média igual a: " + media);

} // Fim do Método Main

} // Fim da Classe Alunos
}

```

09.05 – Contadores e incremento e while

Este programa tem por finalidade apresentar os números ímpares situados na faixa de 0 a 10. Demonstra a utilização da estrutura de repetição while, contadores e incremento.

```

using System;

namespace Ímpares
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Numeros
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int n, cont;

            // Inicialização de Variáveis
            n = 1; // Primeiro número ímpar
            cont = 1;

            Console.WriteLine("\n\t\t Exibe os números ímpares situados entre 0 e 10!");

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");

            while (n <= 10)
            {
                // Estrutura de repetição while

```



```

N1 = Double.Parse(Console.ReadLine());
Console.WriteLine();    // Adiciona uma linha em branco

Console.WriteLine("\t\tInforme a 2º nota do aluno: ");
N2 = Double.Parse(Console.ReadLine());
Console.WriteLine();    // Adiciona uma linha em branco

Console.WriteLine("\t\tInforme a 3º nota do aluno: ");
N3 = Double.Parse(Console.ReadLine());
Console.WriteLine();    // Adiciona uma linha em branco

Console.WriteLine("\t\tInforme a 4º nota do aluno: ");
N4 = Double.Parse(Console.ReadLine());
Console.WriteLine();    // Adiciona uma linha em branco

// Calcula a Média do Aluno

MEDIA = (N1 + N2 + N3 + N4)/4;

// Verifica se o aluno foi aprovado ou não

if (MEDIA >=7) // Estrutura de Seleção Composta
    Console.WriteLine("\t\tO aluno foi aprovado e sua média é: {0}", MEDIA);
else
{
    // Utilize chaves em bloco de instruções

    Console.WriteLine("\t\tInforme a nota do exame: ");
    NE = Double.Parse(Console.ReadLine());

    // Calcula a nova média do aluno
    NM = (NE + MEDIA)/2;

    if ( NM >= 7)
    {
        Console.WriteLine("\n\t\tAluno \"Aprovado\" no exame com média
final igual a: " + NM + "\n\n");
    }
    else
    {
        Console.WriteLine("\n\t\tAluno \"Reprovado\" no exame com
média final igual a: " + NM + "\n\n");
    }
}

} // Fim do método Main

} // Fim da classe Alunos
}

```

09.07 – Recebe números com e extrai cada dígito

Este programa recebe um número com 4 dígitos entre 1000 e 9999 e extrai cada dígito. Utiliza os operadores módulos, métodos de console.

```

using System;

namespace Digitos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Numericos
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int num,          // Número a ser fornecido pelo usuário
                d1, d2, d3, d4; // Dígitos a serem extraídos do número fornecido pelo
usuário

            // Solicita e lê um número de 4 dígitos
            Console.WriteLine("\n\tInforme o Número: ");
            num = Int32.Parse(Console.ReadLine());

            // Extrai os 4 dígitos do número fornecido
            d1 = (num / 1000) % 10;
            d2 = (num / 100) % 10;
            d3 = (num / 10) % 10;
            d4 = (num % 10);

            // Exibe o número
            Console.WriteLine("\n\tO número fornecido é: " + d1 + " " + d2 + " " + d3 + "
" + d4 + "\n\n");

        } // Fim do Método Main

    } // Fim da Classe Numéricos
}

```

09.08 – Aninhamento de estruturas de seleção

Este programa tem por finalidade mostrar a utilização do aninhamento de estruturas de seleção dupla if / else.

```

using System;

namespace Aninhamento
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Estruturas
    {

```

```

/// <summary>
/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main(string[] args)
{
    int nota;

    // Solicita e lê a nota final de um aluno
    Console.Write("Informe a nota final do aluno: ");
    nota = Int32.Parse(Console.ReadLine());

    if (nota >= 90)
    {
        Console.WriteLine("Categoria do aluno é: A");
        Console.WriteLine("Aluno Aprovado!");
    }
    else
    {
        if (nota >= 80)
        {
            Console.WriteLine("Categoria do aluno é: B");
            Console.WriteLine("Aluno Aprovado!");
        }
        else
        {
            if (nota >= 70)
            {
                Console.WriteLine("Categoria do aluno é: C");
                Console.WriteLine("Aluno Aprovado!");
            }
            else
            {
                if (nota >= 60)
                {
                    Console.WriteLine("Categoria do aluno é: D");
                    Console.WriteLine("Aluno Reprovado!");
                }
                else
                {
                    Console.WriteLine("Categoria do aluno é: E");
                    Console.WriteLine("Aluno Reprovado!");
                }
            }
        }
    }
}

} // Fim do Método Main

} // Fim da Classe Estruturas
}

```

09.09 – Soma dez primeiros números

Tem por finalidade efetuar a soma dos dez primeiros números inteiros. Demonstra como utilizar a estrutura de repetição while, incremento e contadores.

```
using System;

namespace Somatorio
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Numeros
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int cont, soma;

            cont = 1;
            soma = 0;

            // Exibe uma linha na tela
            Console.WriteLine("\t\t_____");

            while (cont <= 10)    // Estrutura de repetição while
            {
                soma += cont;
                cont++;
            }

            // Exibe o resultado da soma dos dez primeiros números
            Console.WriteLine("\n\t\tA soma dos 10 primeiros números inteiros é: " + soma);

            // Exibe uma linha na tela
            Console.WriteLine("\t\t_____");

            Console.WriteLine("\n\t\t\t\thttp://www.msdnbrasil.com.br");

            // Exibe uma linha em branco na tela
            Console.WriteLine();

        } // Fim do Método Main
    } // Fim da Classe Numeros
}
```


Exercícios 10 – Estrutura de Repetição While e outras

10.01 – Calcula tabuada de um número

Estrutura de repetição while. Calcula a tabuada de um número qualquer.

```
using System;

namespace Tabuada
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int     NUM, CONT, TAB;


            // Inicialização de Variáveis
            CONT = 1;
            TAB = 0;


            Console.WriteLine("\n\t\t\tExibe a tabuada de um número qualquer!");


            // Exibe uma linha na tela

Console.WriteLine("\t\t_____");


            // Solicita e lê o número fornecido pelo usuário
            Console.Write("\t\tInforme o número desejado: ");
            NUM = Int32.Parse(Console.ReadLine());
            Console.WriteLine();
            Console.WriteLine();


            // Estrutura de repetição while executando a tabuada
            while (CONT <=10)
            {
                TAB = NUM * CONT;
                Console.WriteLine("\t\t\t\t{0} X {1} = {2}", NUM, CONT, TAB);
                CONT++;
            }


            // Exibe uma linha na tela

Console.WriteLine("\t\t_____");
```

```

        Console.WriteLine("\n\t\t\t\t\thttp://www.msdnbrasil.com.br");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do Método Main

} // Fim da Classe Números
}

```

10.02 – Lê valores e pede novos valores

Estrutura de repetição while e manipulação com *strings*. Este programa faz a leitura de um valor e continuará a pedir novos valores com base na decisão do usuário, o valor para continuar deverá ser "sim".

```

using System;

namespace Usuário
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Responde
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            double resultado, // Resultado da expressão algébrica
            usuário          // Número qualquer fornecido pelo

            string resposta; // Sim ou Não

            resposta = "sim";
            while (resposta == "sim")
            {
                // Solicita e lê um número do usuário
                Console.Write("Informe um número qualquer: ");
                n = Double.Parse(Console.ReadLine());
                Console.WriteLine();

                // Efetua a expressão algébrica
                resultado = n * 3;

                // Exibe o resultado da expressão algébrica
                Console.WriteLine("O resultado do número {0} multiplicado por 3 é:
{1}\n", n, resultado);
            }
        }
    }
}

```



```

c_pessoas, // Número de pessoas entrevistadas que
participaram da pesquisa

c_mulheres, // Números de números entrevistas na pesquisa
c_olhos, // Número de pessoas do sexo feminino (que
estão nas condições de existência)

string sexo, // sexo do entrevistado
c_olhos, // Cor dos olhos do entrevistado
c_cabelos; // Cor dos cabelos do entrevistado

double p_18_35; // Porcentagem de pessoas do sexo feminino

// Inicialização de Variáveis
c_pessoas = 0;
c_mulheres = 0;
c_olhos = 0;
m_idade = 0;

/* Amostragem de uma pesquisa de uma população de uma certa região, a qual
coletou os

* seguintes dados referentes a cada habitante para serem analisados:
* sexo (masculino ou feminino)
* cor dos cabelos (louros, pretos e castanhos)
* cor dos olhos (azuis, verdes e castanhos)
* /

/* Calcular:
* A maior idade dos habitantes
* A porcentagem de indivíduos do sexo feminino cuja idade está entre 18 e 35
anos

* inclusive e que tenham olhos verdes e cabelos louros
* O final do conjunto de habitantes é reconhecido pelo valor -1 entrado com
idade...

*/
// Solicita e lê a Idade de um Entrevistado
Console.WriteLine("\tInforme a Idade do Entrevistado, -1 para Sair: ");
idade = Int32.Parse(Console.ReadLine());
Console.WriteLine();

// Atribuindo a idade lida como sendo a maior idade, pois é a primeira idade
fornecida

while (idade != -1)
{
    if (idade >= m_idade)
    {
        m_idade = idade;
    }

    // Solicita e lê o sexo do entrevistado
    Console.WriteLine("\tInforme o sexo do entrevistado: ");
    sexo = Console.ReadLine();

    // Verifica o total de Mulheres que participaram da pesquisa
    if ((sexo == "Feminino") || (sexo == "FEMININO") || (sexo == "feminino"))
        c_mulheres = c_mulheres + 1;
}

```

```

// Solicita e lê a Cor dos Olhos do Entrevistado
Console.WriteLine("\tInforme a cor dos olhos do entrevistado: ");
c_olhos = Console.ReadLine();

// Solicita e lê a Cor dos Cabelos Entrevistado
Console.WriteLine("\tInforme a cor dos cabelos do entrevistado: ");
c_cabelos = Console.ReadLine();
Console.WriteLine();

// Calcula as mulheres nesta condição
/*
* Porcentagem de indivíduos do sexo feminino cuja idade está entre 18
e 35 anos inclusive
* e que tenham olhos verdes e cabelos louros
*/

if (((sexo == "FEMININO") || (sexo == "feminino") || (sexo == "Feminino"))
&& ((idade >= 18) && (idade <= 35)) && ((c_olhos == "Verdes") || (c_olhos == "verdes") || (c_olhos ==
"VERDES")) && ((c_cabelos == "Louros") || (c_cabelos == "louros") || (c_cabelos == "LOUROS")))
{
    ciscf = ciscf + 1;
}

// Solicita e lê a Idade de um Entrevistado
Console.WriteLine("\tInforme a Idade do Entrevistado, -1 para Sair: ");
idade = Int32.Parse(Console.ReadLine());
Console.WriteLine();

// Calcula a quantidade de pessoas que participaram da pesquisa
c_pessoas = c_pessoas + 1;

} // Fim do laço while

// Verifica a porcentagem de mulheres que estão nesta condição entre o total de
mulheres entrevistadas

if (ciscf > 0)
{
    pscf = ((ciscf * 100) / c_mulheres);
    Console.WriteLine();
    Console.WriteLine("\tMulheres que estão na condição especificadas são:
{0}%\n", pscf);
}
// Exibe a maior idade encontrada
if (m_idade == 0)
{
    Console.WriteLine("\t\tNão foi entrevistado nenhuma pessoa\n");
}
else
{
    Console.WriteLine("O total de entrevistas foi de: {0}
pessoas", c_pessoas);
    Console.WriteLine("O total de pessoas do sexo feminino na pesquisa foi
de: {0} mulheres", c_mulheres);
}

```

```

        Console.WriteLine("A maior idade fornecida na pesquisa é de: {0}
anos",m_idade);
    }

    // Exibe uma linha na tela

    Console.WriteLine("\t\t_____");
    Console.WriteLine("\n\t\t\t\t\thttp://www.msdnbrasil.com.br");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

} // Fim do método Main

} // Fim da Classe Entrevistados
}

```

10.04 – Lê valores e conta os números negativos

Estrutura de repetição while. Lê 5 valores para um número qualquer, um de cada vez, e conta quantos destes valores são negativos.

```

using System;

namespace Negativos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    cont, neg;
            double num;

            // Inicialização de variáveis
            cont = 1;      // contador de números informados pelo usuário
            neg = 0;       // contador de números negativos

            Console.WriteLine("\n\t\tExibe o total de números negativos informados pelo
usuário!");

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");

            // Estrutura de repetição while

```

```

instruções) while( cont <=5)           // Utilize chaves para mais de uma instrução (blocos de
{
    // Solicita e lê um número informado pelo usuário
    Console.WriteLine("\t\tInforme um número: ");
    num = Double.Parse(Console.ReadLine());

    Console.WriteLine();    // Imprime uma linha em branco

    if (num < 0)             // Estrutura de seleção simples
        neg++;

    cont += 1;
}

// Exibe o resultado na tela
Console.WriteLine("\t\tA quantidade números negativos é: {0}\n", neg);

// Exibe uma linha na tela
Console.WriteLine("\t\t_____");
Console.WriteLine("\n\t\t\t\t\thttp://www.msdnbrasil.com.br");

// Exibe uma linha em branco na tela
Console.WriteLine();

} // Fim do Método Main

} // Fim da Classe Números
}

```

10.05 – Calcula média de idade

Estrutura de repetição while e conversão para double (cast). Este programa tem por finalidade calcular a média de idades de um grupo de pessoas.

```

using System;

namespace Idade
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Grupo
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)

```

```

{
    // Declaração de Variáveis
    int      s_idade,      // soma das idades do grupo de pessoas
            idade,        // idade de uma pessoa
            c_pessoas;    // contador de pessoas

    double m_idade;      // média das idades

    // Inicialização de Variáveis
    m_idade = 0;
    s_idade = 0;
    c_pessoas = 1;

    // Solicita e lê a idade do Indivíduo
    Console.WriteLine("\t\tInforme a idade da pessoa, \'-1 para Sair\': ");
    idade = Int32.Parse(Console.ReadLine());

    while (idade > 0 )
    {
        s_idade = s_idade + idade;
        m_idade = (double) s_idade / c_pessoas;
        c_pessoas = c_pessoas + 1;

        // Solicita e lê a idade do Indivíduo
        Console.WriteLine("\t\tInforme a idade da pessoa, \'-1 para Sair\': ");
        idade = Int32.Parse(Console.ReadLine());

    } // Fim do laço de repetição

    // Apresenta a média de idades do grupo de pessoas
    Console.WriteLine("\n\t\tA média das idades fornecidas é: {0}\n", m_idade);

    // Exibe uma linha na tela
    Console.WriteLine("\t\t_____");
    Console.WriteLine("\n\t\t\t\t\thttp://www.msdnbrasil.com.br");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

    } // Fim do método Main

} // Fim da classe Grupo
}

```

10.06 – Calcula tempo

Estruturas de repetição while. Calcula o tempo necessário para que um País A se iguale ou ultrapasse o número de habitantes de um País B.

```
using System;
```

```
namespace População
```



```

{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Habitantes
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int cont_anos; // Número de anos para a população se igualar
            double popA, // Número de habitantes do País A
                popB; // Número de habitantes do País B

            // Inicialização de Variáveis
            popA = 90000000; // 90.000.000 habitantes
            popB = 120000000; // 120.000.000 habitantes
            cont_anos = 0;

            Console.WriteLine("\n\tCalcula o tempo (anos) para dois países igualarem suas
populações!");

            // Exibe uma linha na tela

            Console.WriteLine("\t_____");

            while(popA <= popB) // Estrutura de repetição while
            {
                popA = (popA + (popA * 0.03)); // População A com 3% de
crescimento anual
                popB = (popB + (popB * 0.015)); // População B com 1.5
% de crescimento anual
                cont_anos = cont_anos + 1;
            }

            // Exibe o resultado
            Console.WriteLine("\n\n\n\tO País A se igualou em número de habitantes ao
País B em {0} anos\n", cont_anos);

            // Exibe uma linha na tela

            Console.WriteLine("\n\n\n\t_____
—");

            Console.WriteLine("\n\t\t\t\t\thttp://www.msdnbrasil.com.br");

            // Exibe uma linha em branco na tela
            Console.WriteLine();

        } // Fim do método Main
    } // Fim da Classe Habitantes
}

```

```
}
```

10.07 – Gera números divisíveis

Estrutura de repetição while e operador módulo e estrutura de seleção simples encadeada. Gera os números de 10 a 100 que são divididos por 11 onde o resto da divisão é igual a 5.

```
using System;

namespace Divididos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    cont;

            // Inicialização de Variáveis
            cont = 10;

            Console.WriteLine("\n\t\tExibe os números que são divisíveis por 11 com resto
igual a 5!");

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");

            // Utilize chaves para mais de uma instrução (blocos de instruções)
            while (cont <= 100)
            {
                if (cont % 11 == 5)
                    Console.WriteLine("\n\t\tO número {0} é divisível por 11", cont);

                cont += 1;
            }

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");
            Console.WriteLine("\n\t\t\t\thttp://www.msdnbrasil.com.br");

            // Exibe uma linha em branco na tela
            Console.WriteLine();
        }
    }
}
```

```

        } // Fim do Método Main

    } // Fim da Classe Números
}

```

10.08 – Média geral de alunos

Estrutura de repetição while. Este programa tem por finalidade calcular a Média Geral de uma Turma de alunos.

```

using System;

namespace Turma
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Alunos
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Declaração de Variáveis
            int    N_ALUNOS,    // Número de alunos
                  CONT;        // Contador de alunos

            double N1, N2, N3, N4, // Notas de um aluno
                  MEDIA,           // Média de um aluno
                  MEDIAT,          // Média da turma
            de alunos
                  MEDIAG;          // Média Geral

            // Inicialização de Variáveis
            CONT = 1;
            MEDIAG = 0;
            MEDIA = 0;
            MEDIAT = 0;

            // Solicita o número de alunos
            Console.WriteLine("Informe o número de alunos: ");
            N_ALUNOS = Int32.Parse(Console.ReadLine());
            Console.WriteLine();

            while (CONT <= N_ALUNOS)
            {
                // Solicita e lê as notas de um aluno
                Console.WriteLine("Informe a 1º nota do aluno: ");
                N1 = Double.Parse(Console.ReadLine());

                Console.WriteLine("Informe a 2º nota do aluno: ");

```

```

        N2 = Double.Parse(Console.ReadLine());

        Console.WriteLine("\t\tInforme a 3º nota do aluno: ");
        N3 = Double.Parse(Console.ReadLine());

        Console.WriteLine("\t\tInforme a 4º nota do aluno: ");
        N4 = Double.Parse(Console.ReadLine());
        Console.WriteLine();

        // Calcula a Média do Aluno
        MEDIA = ((N1 + N2 + N3 + N4) / 4);
        Console.WriteLine("\t\tA Média do {0}º aluno é: {1}\n", CONT, MEDIA);

        // Calcula a Média da Turma
        MEDIAT = (MEDIAT + MEDIA);

        CONT = CONT + 1;

    } // Fim do laço de Repetição

    MEDIAG = MEDIAT/N_ALUNOS;

    // Exibe a média da turma
    Console.WriteLine("\t\tA Média geral da turma é: {0}\n", MEDIAG);

    // Exibe uma linha na tela
    Console.WriteLine("\t\t_____");
    Console.WriteLine("\n\t\t\t\t\thttp://www.msdnbrasil.com.br");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

    } // Fim do Método Main

} // Fim da Classe Alunos
}

```

10.09 – Calcula média de turma de alunos

Estrutura de repetição while. Este programa calcula a média de uma turma de 10 alunos.

```

using System;

namespace Notas
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Alunos
    {
        /// <summary>

```

[illegible]

```
// Calcula a Média Total da Turma
mediag = (mediat/cont);

// Exibe a Média da Turma de Alunos
Console.WriteLine("\n\t\tA média da turma de {0} aluno(s) é: {1}\n", cont, mediag);

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");
Console.WriteLine("\n\t\t\t\t\thttp://www.msdnbrasil.com.br");

// Exibe uma linha em branco na tela
Console.WriteLine();

} // Fim do método Main

} // Fim da Classe Alunos
}
```

```
Console.WriteLine("\t\t{0}\t\t {1}", n, soma);
```



```

static void Main(string[] args)
{
    // Exibe uma linha na tela

Console.WriteLine("\t\t_____");
    Console.WriteLine("\t\tCalcula o número de alunos reprovados e aprovados!");

    int aprovados = 0,          // número de aprovados
        reprovados = 0, // número de reprovados
        alunos = 1,      // contador de alunos
        resultado;        // resultado do exame

    // processa 10 alunos; laço controlado por contador
    while (alunos <=10)
    {
        Console.Write("\n\t\tEntre com o resultado(1 = aprovado, 2 = reprovado):

");

        resultado = Int32.Parse(Console.ReadLine());

        if (resultado == 1)
        {
            aprovados = aprovados + 1;
        }
        else
        {
            reprovados = reprovados + 1;
        }
        alunos = alunos + 1;

    } // Fim da estrutura de repetição while

    // fase de Conclusão
    Console.WriteLine();
    Console.WriteLine("\t\t\t\tAprovados: " + aprovados + "\n");
    Console.WriteLine("\t\t\t\tReprovados: " + reprovados + "\n");

    if (aprovados > 8)
    {
        Console.WriteLine("\t\t\t\tAumentar o preço do curso\n");
    }

    // Exibe uma linha na tela

Console.WriteLine("\t\t_____");
    Console.WriteLine("\n\t\t\t\thttp://www.gupnet.com.br");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

    } // fim do método Main

} // Fim da classe Análise
}

```

11.04 – Resultado de uma pesquisa de aceitação

Estrutura de repetição while encadeada com if / else e aninhadas. Calcula os resultados de uma pesquisa de aceitação de um produto com base na resposta do entrevistado (sim ou não), sabendo-se que foram entrevistados 500 pessoas. Calcular:

- * O número de pessoas que responderam sim;
- * O número de pessoas que responderam não;
- * A percentagem de pessoas do sexo feminino que respondem sim;
- * A percentagem de pessoas do sexo masculino que respondem não.

```
using System;
```

```
namespace Firma
```

```

{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Pesquisa
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            Console.WriteLine("\n\t\t\tPesquisa de aceitação de um produto!!");

            Console.WriteLine("\n\t\t\t_____ \n");

            // Declaração de Variáveis
            int c_pessoas, // Número de Pessoas que participaram
da pesquisa
            nprs, // Número de pessoas que responderam
sim
            nprn, // Número de pessoas que responderam
não
            c_psf, // Número de pessoas do sexo feminino
            c_psm, // Número de pessoas do sexo masculino
            c_psfrs, // Número de pessoas do sexo feminino que
responderam sim
            c_psmrn; // Número de pessoas do sexo masculino
que responderam não

            double ppsfrs, // Porcentagem de pessoas do sexo feminino que
responderam sim
            ppsmrn; // Porcentagem de pessoas do
sexo masculino que responderam não

```

```

string  sexo,           // Sexo Masculino ou Feminino
        resposta;      // sim ou não

// Inicialização de Variáveis
c_pessoas = 0;
nprs = 0;
nprn = 0;
c_psf = 0;
c_psm = 0;
c_psfrs = 0;
c_psmrn = 0;
ppsfrs = 0;
ppsmrn = 0;

while( c_pessoas < 5 ) // Estrutura de repetição while
{
    // Solicita e lê o sexo do entrevistado
    Console.WriteLine("\t\tInforme o sexo do entrevistado: ");
    sexo = Console.ReadLine();

    // Solicita e lê a resposta do entrevistado
    Console.WriteLine("\t\tInforme a resposta do entrevistado: ");
    resposta = Console.ReadLine();
    Console.WriteLine();

    // Calcula o número de pessoas que responderam sim ou não
    if ((resposta == "SIM")|| (resposta == "Sim")|| (resposta == "sim")||
(resposta == "S")|| (resposta == "s"))
        nprs++;
    else
        nprn++;

    // Calcula o número de pessoas que pertencem ao sexo feminino ou
masculino
    if ((sexo == "FEMININO")|| (sexo == "Feminino")|| (sexo == "feminino"))
        c_psf++;
    else
        c_psm++;

    // Calcula o número de pessoas do sexo feminino que responderam sim
    if (((sexo == "FEMININO")|| (sexo == "Feminino")|| (sexo == "feminino"))&&
((resposta == "SIM")|| (resposta == "Sim")|| (resposta == "sim")|| (resposta == "S")|| (resposta == "s")))
        c_psfrs++;

    // Calcula o número de pessoas do sexo masculino que responderam não
    if (((sexo == "MASCULINO")|| (sexo == "Masculino")|| (sexo ==
"masculino"))&& ((resposta == "NÃO")|| (resposta == "Não")|| (resposta == "não")|| (resposta == "N")||
(resposta == "n")))
        c_psmrn++;

    // Incrementa o contador de pessoas
    c_pessoas++;

} // Fim do laço de repetição while

// Verifica se houve pessoas do sexo feminino na pesquisa

```

```

        if (c_psf > 0)
        {
            // Calcula a porcentagem de pessoas do sexo feminino que responderam
            sim
            ppsfrs = ((c_psf * 100)/c_psf);
            Console.WriteLine("\t\tMulheres que responderam sim são: {0}%\n",
            ppsfrs);
        }
        else
        {
            Console.WriteLine("\t\tNenhuma mulher participou da pesquisa!\n");
        }

        // Verifica se houve pessoas do sexo masculino na pesquisa
        if (c_psm > 0)
        {
            // Calcula a porcentagem de pessoas do sexo masculino que
            responderam não
            ppsmrn = ((c_psm * 100)/c_psm);
            Console.WriteLine("\t\tHomens que responderam não são: {0}%\n",
            ppsmrn);
        }
        else
        {
            Console.WriteLine("\t\tNenhum homem participou da pesquisa!");
        }

        // Exibe os resultados dos cálculos
        Console.WriteLine("\t\tO total de pessoas que responderam sim é: {0}\n", nprs);
        Console.WriteLine("\t\tO total de pessoas que responderam não é: {0}\n", nprn);

        // Exibe uma linha na tela
        Console.WriteLine("\t\t_____");
        Console.WriteLine("\n\t\t\t\t\thttp://www.gupnet.com.br");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do método Main

} // Fim da Classe Pesquisa
}

```

11.05 – Calcula média de uma turma de alunos

Este programa calcula a média de uma turma de alunos. Utiliza métodos de console, estrutura de repetição while e inicialização de variáveis.

```

using System;

namespace Classes
{

```

```

/// <summary>
/// Summary description for Class1.
/// </summary>
class Alunos
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main(string[] args)
    {
        // Exibe uma linha na tela

Console.WriteLine("\t\t");
        Console.WriteLine("\t\tCalcula a média de uma turma de alunos!!!");

        // Declaração de Variáveis
        int contador;

        double media, nota, total;

        // Inicialização de Variáveis
        total = 0;
        contador = 0;

        // Fase de processamento
        Console.Write("\n\t\tEntre com a nota do aluno, -1 para Sair: ");
        nota = Double.Parse(Console.ReadLine());

        // Estrutura de repetição while
        while (nota != -1)
        {
            // Adiciona nota ao total
            total = total + nota;

            // Adiciona 1 ao contador
            contador = contador + 1;

            // Fase de processamento
            Console.Write("\n\t\tEntre com a nota do aluno, -1 para Sair: ");
            nota = Double.Parse(Console.ReadLine());

        } // Fim da estrutura de repetição while

        // Fase de Conclusão
        if( contador !=0)
        {
            media = total / contador;

            // Exibe a média das notas do teste
            Console.WriteLine("\n\t\tClasse com média igual a: " + media);
        }
        else
        {
            // Exibe o resultado

```

```

        Console.WriteLine("\n\t\tNenhuma nota foi informada!!!");
    }

    // Exibe uma linha na tela

Console.WriteLine("\t\t_____");
    Console.WriteLine("\n\t\t\t\thttp://www.gupnet.com.br");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

} // Fim do método Main

} // Fim da Classe Alunos
}

```

11.06 – Lê número e soma entre uma série

Estrutura de repetição for. Lê um número inteiro (N) fornecido pelo usuário e soma os números entre 1 e N.

```

using System;

namespace Indeterminado
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Quantidade
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela

Console.WriteLine("\t\t_____");
            Console.WriteLine("\t\tLê um número inteiro (N) fornecido pelo usuário!");

            // Declaração de Variáveis
            int    N, SOMA;

            // Inicialização de variáveis
            SOMA = 0;

            // Solicita e lê um número do usuário
            Console.Write("\t\tInforme um número: ");
            N = Int32.Parse(Console.ReadLine());

            // Pula uma linha
            Console.WriteLine();

```

```

        // Exibe o cabeçalho da tabela
        Console.WriteLine("\t\tN\t\tSOMA");

        // Estrutura de repetição for
        for (int CONT = 1; CONT <= N; CONT++)
        {
            SOMA = SOMA + CONT;
            Console.WriteLine("\t\t{0}\t\t {1}", CONT, SOMA);
        } // fim da estrutura de repetição for

        // Exibe uma linha em branco na tela
        Console.WriteLine();

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");
        Console.WriteLine("\n\t\t\t\t\thttp://www.gupnet.com.br");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do método Main

} // Fim da classe Quantidade
}

```

11.07 – Calcula uma função

Estrutura de repetição For. Calcula a seguinte função:

$$* F(X,Y) = (((X**2) + 3X + (Y**2))/((XY) - 5Y - 3X + 15))$$

* Para cada valor de x = 1 até 100 e y = de 0 até 5

```

using System;

namespace Função
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Quadratica
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            Console.WriteLine("\t\tCalcula uma função!");
        }
    }
}

```



```

Console.WriteLine("\t\t_____");

        // Declaração de Variáveis
        double F;      //Função

        // Inicialização de variáveis
        F = 0;
        Console.WriteLine("\t\tX\t\tY\t\tF\n");
        for ( int X = 1; X <= 5; X++)      // Para cada valor de x calcula Y
        {
            // N = Math.Pow(X,2);

            for ( int Y = 0; Y <= 5; Y++)
            {
                F = (Math.Pow(X,2)) + 3*X + (Math.Pow(Y,2));
                Console.WriteLine("\t\t{0}\t\t{1}\t\t{2}",X,Y,F);
            }
            Console.WriteLine();

        } // Fim da estrutura de repetição For

        // Exibe uma linha na tela

Console.WriteLine("\t\t_____");
        Console.WriteLine("\n\t\t\t\t\thttp://www.gupnet.com.br");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do método Main

} // Fim da classe Quadrática
}

```

11.08 – Calcula termos de PA

Estrutura de repetição while. Este programa tem por finalidade calcular os termos de uma progressão aritmética de N elementos...

```

using System;

namespace Progressão
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Aritmética
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]

```

```
static void Main(string[] args)
{
    // Exibe uma linha na tela

    Console.WriteLine("_____");
    Console.WriteLine("\t\tCalcula os termos de uma progressão aritmética!");

    // Declaração de Variáveis
    int    A1,           // Primeiro termo da Progressão Aritmética
           AN,           // Enésimo termo da Progressão Aritmética
           N,            // Número de elementos de Progressão Aritmética
           R,            // Razão da Progressão Aritmética
           CONT; // Contador

    // Inicialização de Variáveis
    CONT = 1;
    AN = 0;

    // Solicita e lê o números de elementos da PA
    Console.Write("\tInforme o número de elementos da Progressão Aritmética: ");
    N = Int32.Parse(Console.ReadLine());

    // Solicita e lê o 1º elemento da PA
    Console.Write("\tInforme o 1º elemento da Progressão Aritmética: ");
    A1 = Int32.Parse(Console.ReadLine());

    // Solicita e lê a razão da PA
    Console.Write("\tInforme a razão desta Progressão Aritmética: ");
    R = Int32.Parse(Console.ReadLine());

    Console.WriteLine();

    while (N >= CONT )
    {
        AN = (A1 + ((N - 1) * R));
        Console.WriteLine("\tO {0}º elemento da Progressão Aritmética é: {1}\n",
N, AN);

        N = (N - CONT);
    }

    // Exibe uma linha na tela

    Console.WriteLine("_____");
    Console.WriteLine("\n\t\t\t\t\thttp://www.gupnet.com.br");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

} // Fim do método Main

} // Fim da Classe Aritmética
}
```

11.09 – Calcula temperatura em graus diversos e com variações

Estrutura de repetição For. Este programa tem por finalidade converter a temperatura em graus Fahrenheit para Centígrados variando de 1 em 1 de 32 a 64.

```
using System;

namespace Conversão
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Temperatura
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            Console.WriteLine("\n\t\tConversão de Temperatura de graus Fahrenheit para
Celsius!");

            Console.WriteLine("\t\t_____ \n");

            // Declaração de Variáveis

            double C,      // Temperatura em graus Celsius
                F;          // Temperatura em graus Fahrenheit

            // Inicialização de Variáveis

            F = 32;
            C = 0;

            Console.WriteLine("\t\tFahrenheit\t\tCelsius");

            // Estrutura de repetição for
            for (int cont = 50; cont <= 150; cont++)
            {
                C = (5*(F - 32))/9;
                Console.WriteLine("\t\t {0}" + "\t\t\t {1}", F, C);
                F = F + 1;
            } // Fim da estrutura de repetição for

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");
            Console.WriteLine("\n\t\t\t\t\thttp://www.gupnet.com.br");

            // Exibe uma linha em branco na tela
            Console.WriteLine();
        }
    }
}
```

```
        } // Fim do método Main  
    } // Fim da Classe Temperatura  
}
```

12.01 – Calcula a soma de termos de uma série

$$s = 1000/1 - 997/2 + 994/3 - 991/4 + \dots$$

```
using System;

namespace termos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Serie
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            Console.WriteLine("\n\t\t\tCalcula a soma 50 termos de uma série!!!");

            Console.WriteLine("\t\t\t_____");

            // Declaração de Variáveis
            double parc, // parcela da expressão algébrica
                s, // somatório da expressão algébrica
                num, // numerador
                den, // denominador
                m; // troca o sinal

            // Inicialização de Variáveis
            parc = 0;
            s = 0;
            num = 1000;
            den = 1;
            m = 1;

            // Estrutura de repetição for
            for( int cont = 1; cont <=50; cont++)
            {
                parc = ((num/den)*m);
                s = s + parc;
                m = m * (-1);
                num = num - 3;
                den = den + 1;
            } // Fim da estrutura de repetição for
        }
    }
}
```



```
static void Main(string[] args)
{
    // Exibe uma linha na tela

Console.WriteLine("\t\t_____");
    Console.WriteLine("\t\tCalcula a resposta (sim ou não) em uma pesquisa de
satisfação!\n");

    // Declaração de Variáveis

    string  sexo,          // sexo do funcionário
            resposta;      // resposta do funcionário (sim ou não)

    int     cpsf,          // Número de pessoas do sexo feminino
            cpsm,          // Número de pessoas do sexo masculino
            nprs,          // Número de pessoas que responderam sim na
pesquisa
            nprn,          // Número de pessoas que responderam não na
pesquisa
            npsfrs,        // Número de pessoas do sexo feminino que
responderam sim
            npsmrn;        // Número de pessoas do sexo masculino
que responderam não

    double  ppsfrs,        // Porcentagem de pessoas do sexo feminino que
responderam sim
            ppsmrn;        // Porcentagem de pessoas do sexo
masculino que responderam não

    // Inicialização de variáveis
    cpsm = 0;
    cpsf = 0;
    nprs = 0;
    nprn = 0;
    npsfrs = 0;
    npsmrn = 0;
    ppsfrs = 0;
    ppsmrn = 0;

    // Estrutura de repetição for

for ( int cont = 1; cont <= 5; cont++)
{
    // Solicita e lê os dados do entrevistado(a)
    Console.Write("Informe o sexo, F: (Feminino) ou M: (Masculino): ");
    sexo = Console.ReadLine();

    Console.Write("Informe a resposta, S: (Sim) ou N: (Não): ");
    resposta = Console.ReadLine();

    // Verifica o número de entrevistados que disseram sim ou não
if ((resposta == "s")|| (resposta == "S")|| (resposta == "Sim")|| (resposta ==
"SIM"))|| (resposta == "sim"))
{
            nprs += 1;      // Poderia ser escrito assim nprs++ ou nprs =
nprs + 1;
```



```

    }
    else
    {
        nprn += 1;    // Poderia ser escrito assim nprn++ ou nprn =
nprn + 1;
    }

    // Verifica o número de entrevistados que são do sexo feminino ou
masculino
    if ((sexo == "F")||(sexo == "f")||(sexo == "feminino")||(sexo ==
"Feminino")||(sexo == "FEMININO"))
    {
        cpsf = cpsf + 1;
    }
    else
    {
        cpsm = cpsm + 1;
    }

    // Verifica o número de entrevistados que são do sexo feminino e
responderam sim
    if ((sexo == "F")||(sexo == "f")||(sexo == "feminino")||(sexo ==
"Feminino")||(sexo == "FEMININO")&&(resposta == "S")||(resposta == "Sim")||(resposta == "SIM")||
(resposta == "sim"))
    {
        npsfrs = npsfrs + 1;
    }

    // Verifica o número de entrevistados que são do sexo masculino e
responderam não
    if ((sexo == "M")||(sexo == "m")||(sexo == "masculino")||(sexo ==
"Masculino")||(sexo == "MASCULINO")&&(resposta == "n")||(resposta == "N")||(resposta == "Não")||
(resposta == "NÃO")||(resposta == "não"))
    {
        npsmrn = npsmrn + 1;
    }
} // Fim da estrutura de repetição For

// Verifica se houve pessoas do sexo feminino entrevistadas
// E calcula a porcentagem de pessoas entrevistadas que disseram sim
if (cpsf > 0)
{
    ppsfrs = ((npsfrs * 100)/cpsf);
    Console.WriteLine();
    Console.WriteLine("A porcentagem de pessoas do sexo feminino que
participaram da pesquisa e \ndisseram sim foi de: {0}%", ppsfrs);
}
else
{
    Console.WriteLine();
    Console.WriteLine("Nenhuma mulher gostou do produto lançado no
mercado...");
}
if (cpsm > 0)
{
    ppsmrn = ((npsmrn * 100)/cpsm);
    Console.WriteLine();

```



```

static void Main(string[] args)
{
    // Exibe uma linha na tela
    Console.WriteLine("\t\tEfetua a soma dos números de 2 a 100!!!");

    Console.WriteLine("\n\t\t_____");

    int soma = 0;

    // Estrutura de repetição for
    for (int numero = 2; numero <=100; numero += 2)
    {
        soma += numero;
    }

    // Exibe a soma numa caixa de mensagem
    MessageBox.Show(" A soma é : " + soma,
        "Soma todos os interios de 2 até 100",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);

    } // Fim do Método Main

} // Fim da Classe Somatório
}

```

12.05 – Média de uma turma

Este programa calcula a média de uma turma de 10 alunos. Utiliza métodos de console, estrutura de repetição while e inicialização de variáveis.

```

using System;

namespace Classes
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Alunos
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");
            Console.WriteLine("\t\tCalcula a média de uma turma de 10 alunos!!!");

            // Declaração de Variáveis
            int contador;

```

```

double media, nota, total;

// Inicialização de Variáveis
total = 0;
contador = 1;

// Estrutura de repetição while
while (contador <= 10)
{
    // Solicita e lê a nota do usuário
    Console.Write("\n\t\tEntre com a {0}ª nota do aluno: ", contador);
    nota = Double.Parse(Console.ReadLine());

    // Adiciona nota ao total
    total = total + nota;

    // Adiciona 1 ao contador
    contador = contador + 1;
}

// Fase de Conclusão
media = total / 10;

// Exibe o resultado
Console.WriteLine("\n\t\tClasse com média igual a: " + media);

// Exibe uma linha na tela
Console.WriteLine("\t\t_____");
Console.WriteLine("\n\t\t\t\thttp://www.gupnet.com.br");

// Exibe uma linha em branco na tela
Console.WriteLine();

} // Fim do método Main

} // Fim da classe Alunos
}

```

12.06 – Juros compostos com MessageBox

Estrutura de repetição for e utilização de MessageBox. Calculando juros compostos.

```

using System;
using System.Windows.Forms; // Adicione uma referência no Solution Explorer para utilizar uma
                              MessageBox.

namespace Conta
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>

```



```

namespace Funcionário
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Pessoa
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela

Console.WriteLine("\t\t_____");
            Console.WriteLine("\t\t Calcula o reajuste salarial de 10 funcionários!\n");


            // Declaração de Variáveis

            double salario, // salário do funcionário
                reajuste,    // reajuste do salário do funcionário
                salarion;     // novo salário do funcionário


            // Inicialização de variáveis
            reajuste = 0;
            salarion = 0;


            // Estrutura de repetição for
            for(int cont = 1; cont <= 10; cont++)
            {
                Console.Write("Informe o salário do funcionário R$: ");
                salario = Double.Parse(Console.ReadLine());


                // Pula uma linha
                Console.WriteLine();


                /* Verifica a situação do salário com base nas seguintes condições:
                * Os funcionários com salário inferior a 10.000,00 devem ter reajuste de
55%
                * Os funcionários com salário entre 10.000,00 (inclusive) e 25.000,00
(inclusive)
                * devem ter reajuste de 30%
                * Os funcionários com salário superior a 25.000,00 devem ter reajuste
de 20%
                */
                if (salario < 10000)
                {
                    reajuste = (salario * 0.55);
                    salarion = salario + reajuste;
                    Console.WriteLine("O reajuste salarial R${0} do funcionário e seu
novo salário são: R${1}\n", reajuste, salarion);
                }
                else
                {

```

```

        if (salario <= 25000)
        {
            reajuste = (salario * 0.30);
            salarion = salario + reajuste;
            Console.WriteLine("O reajuste salarial R${0} do
funcionário e seu novo salário são: R${1}\n", reajuste, salarion);
        }
        else
        {
            reajuste = (salario * 0.20);
            salarion = salario + reajuste;
            Console.WriteLine("O reajuste salarial R${0} do
funcionário e seu novo salário são: R${1}\n", reajuste, salarion);
        }
    }
} // Fim da estrutura de repetição For

Console.WriteLine("\t\t_____");
Console.WriteLine("\n\t\t\t\t\thttp://www.gupnet.com.br");

// Exibe uma linha em branco na tela
Console.WriteLine();

} // Fim do método Main

} // Fim da classe Pessoa
}

```

12.08 – Repetição controlada por contador

Exemplos utilizando a estrutura de repetição for. Repetição controlada por contador

```

using System;

namespace Fores
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Exemplosfor
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            Console.WriteLine("\n\t\t Exemplos utilizando a estrutura de repetição for!");

            Console.WriteLine("\t\t_____ \n");
        }
    }
}

```



```
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Bonus
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela

Console.WriteLine("\t\t\t\t\t");
                Console.WriteLine("\t\t\t\t\tCalcula o bônus salarial!!!\n");

            // Declaração de variáveis
            double salario, // salário do funcionário
                    reajuste, // reajuste salarial do funcionário (bônus)
                    montante; // total dos bônus aplicados aos 30 funcionários

            int cont_anos; // tempo que o funcionário atua na empresa em anos

            string sexo; // sexo do funcionário(a)

            // Inicialização de Variáveis
            reajuste = 0;
            montante = 0;

            // Estrutura de repetição for
            for (int cont = 1; cont <= 10; cont++)
            {
                // Solicita e lê o sexo do funcionário(a)
                // Lendo um valor do tipo string
                Console.Write("Informe o sexo do funcionário(a): ");
                sexo = Console.ReadLine();
                // Pula uma linha
                Console.WriteLine();

                // Solicita e lê o valor do salário do funcionário(a)
                // Lendo um valor do tipo double
                Console.Write("Informe o salário do funcionário(a):R$ ");
                salario = Double.Parse(Console.ReadLine());
                // Pula uma linha
                Console.WriteLine();

                // Solicita e lê a quantidade de tempo de casa do funcionário(a)
                // Lendo um valor do tipo int
                Console.Write("Informe o quantidade de anos trabalhados na empresa:");

                cont_anos = Int32.Parse(Console.ReadLine());
                // Pula uma linha
                Console.WriteLine();
            }
        }
    }
}
```

```

/* Verifica a condição em que se enquadra o salário do funcionário(a)
* para o tempo de serviço executado na empresa, conforme a seguinte
tabela:
anos terão
um bônus

* Os funcionários do sexo masculino com tempo de casa superior a 15
* direito a um bônus de 20% do seu salário;
* As funcionárias com tempo de casa superior a 10 anos terão direito a
* de 25% do seu salário;
* Os demais funcionários terão direito a um bônus de R$ 5.000,00
*/

if (((sexo == "masculino")||(sexo == "Masculino")||(sexo ==
"MASCULINO")||(sexo == "masc")||(sexo == "MASC")||(sexo == "Masc"))&& (cont_anos > 15))
{
    reajuste = (salario * 0.20);
    Console.WriteLine("O reajuste salarial do funcionário foi de:
R${0}\n", reajuste);
}
else
{
    if (((sexo == "feminino")||(sexo == "Feminino")||(sexo ==
"FEMININO")||(sexo == "fem")||(sexo == "FEM")||(sexo == "Fem"))&& (cont_anos > 10))
    {
        reajuste = (salario * 0.25);
        Console.WriteLine("O reajuste salarial da funcionária foi
de: R${0}\n", reajuste);
    }
    else
    {
        reajuste = 5000.00;
        Console.WriteLine("O reajuste salarial da funcionário(a)
foi de: R${0}\n", reajuste);
    }
}

// Calcula o montante (total de bônus) gasto com os funcionários
montante = montante + reajuste;

} // Fim do laço de repetição for

// Exibe o resultado
Console.WriteLine("O montante gasto com os funcionários(as) foi de: R${0}",
montante);

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");
Console.WriteLine("\n\t\t\t\t\thttp://www.gupnet.com.br");

// Exibe uma linha em branco na tela
Console.WriteLine();

} // Fim do método Main

} // Fim da classe Bônus

```

```
}
```

12.10 – Lê número e faz multiplicações

Efetua a leitura de um número qualquer e multiplica o mesmo por 3, isso em 5 vezes. Estrutura de repetição for.

```
using System;

namespace Valor
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Qualquer
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            Console.WriteLine("\n\tEfetua a multiplicação de um número por 3 e faz isso em
5 vezes!");

            Console.WriteLine("\t
_____");

            // Declaração de variáveis

            int    numero, resposta;

            for(int cont = 1; cont <=5; cont++)    // Executa o teste de repetição 5 vezes
            {
                Console.WriteLine("\t\tInforme o " + cont + " º valor: ");
                numero = Int32.Parse(Console.ReadLine());

                Console.WriteLine();

                // Efetua a multiplicação do número informado por * 3
                resposta = numero * 3;

                Console.WriteLine("\t\tO valor agora é: {0}\n", resposta);
            }

            // Exibe uma linha na tela

            Console.WriteLine("\t\t
_____");
            Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");
```

```
        Console.WriteLine();  
    }  
}  

```

Exercícios 13 – Estrutura de Repetição For, While e outras

13.01 – Faz pesquisa de preços por região

Estrutura de seleção composta (switch). Verifica a região e o preço do produto e informa ao usuário o local

```
using System;
```

namespace Região

{

/// <summary>

```
/// Summary description for Class1.
```

/// </summary>

```
class Produto
```

{

```
/// <summary>
```

```
/// The main entry point for the application.
```

/// </summary>

[STAThread]

```
static void Main(string[] args)
```

{

```
// Exibe uma linha na tela
```

```
Console.WriteLine("\n\t\t\t\t\tVerifica a região e o preço do produto\n\t\t\t\t\t e informa
```

```
ao usuário o local!!");
```

```
Console.WriteLine("\t\t_____ \n");
```

```
// Declaração de Variáveis
```

```
double preco; // Preço do Produto
```

```
int origem; // Código da origem do produto
```

```
string resposta; // Resposta do usuário para verificar novos produtos
```

```
// Inicialização de variáveis
```

```
resposta = "SIM";
```

```
while (resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==
```

```
"s" || resposta == "S")
```

{

// Solicita e lê o Preço do produto

```
Console.WriteLine("Informe o preço do produto:");
```

```
preco = Double.Parse(Console.ReadLine());
```

```
// Pula uma linha
```

```
Console.WriteLine();
```

```
// Solicita e lê o código de origem do produto
```

```
Console.WriteLine("\t\t1 - Sul\t" + "\t\t5 ou 6 - Nordeste\n")
```

```
+ "\t\t2 - Norte\t" + "\t7, 8 ou 9 - Sudeste\n"
```

```

        + "\t\t3 - Leste\t" + "\t10 - Centro-Oeste\n"
        + "\t\t4 - Oeste\t" + "\t11 - Centro-Leste\n\n");

Console.WriteLine("\t\tInforme o código de origem do produto: ");
origem = Int32.Parse(Console.ReadLine());

// Pula uma linha
Console.WriteLine();

switch(origem)
{
    case 1:
        Console.WriteLine("\tO preço do produto é de R$" +
preco + " e sua região é a Sul\n");
        break;

    case 2:
        Console.WriteLine("\tO preço do produto é de R$" +
preco + " e sua região é a Norte\n");
        break;

    case 3:
        Console.WriteLine("\tO preço do produto é de R$" +
preco + " e sua região é a Leste\n");
        break;

    case 4:
        Console.WriteLine("\tO preço do produto é de R$" +
preco + " e sua região é a Oeste\n");
        break;

    case 5: // Intervalo de 5 até 6 a mesma opção
    case 6: // Intervalo de 5 até 6 a mesma opção
        Console.WriteLine("\tO preço do produto é de R$" +
preco + " e sua região é a Nordeste\n");
        break;

    case 7: // Intervalo de 7 até 9 a mesma opção
    case 8: // Intervalo de 7 até 9 a mesma opção
    case 9: // Intervalo de 7 até 9 a mesma opção
        Console.WriteLine("\tO preço do produto é de R$" +
preco + " e sua região é a Sudeste\n");
        break;

    case 10:
        Console.WriteLine("\tO preço do produto é de R$" +
preco + " e sua região é a Centro-Oeste\n");
        break;

    case 11:
        Console.WriteLine("\tO preço do produto é de R$" +
preco + " e sua região é a Centro-Leste\n");
        break;
}

```

```

                                default: // Verifica todos os outros códigos que não estão na
opção inicial
                                Console.WriteLine("\tVocê não informou um código de
origem de produto correto!\n");
                                break;

                                } // Fim de switch

                                // Verifica se o usuário deseja continuar pesquisando.
                                Console.Write("\tVocê deseja continuar?" + " Digite \"SIM\" para
prosseguir: \a\a");

                                resposta = Console.ReadLine();

                                // Pula uma linha
                                Console.WriteLine();

                                } // Fim do while

                                // Exibe uma linha na tela

Console.WriteLine("\t\t_____");
                                Console.WriteLine("\t\t\t\t\thttp://www.gupnet.com.br");

                                // Exibe uma linha em branco na tela
                                Console.WriteLine();

                                } // Fim do método Main

                                } // Fim da Classe
}

```

13.02 – Reajuste salarial de funcionário

Utiliza estrutura de seleção múltipla switch aninhado em um while. Calcula o reajuste salarial de um funcionário.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para o Método MessageBox.Show

namespace Reajuste
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Salarial
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {

```

[illegible]


```
char opc;

// Inicialização de Variáveis
resposta = "sim";

// Exibe uma linha na tela
MessageBox.Show("Informe 4 números!", "Calcula o quadrado de quatro
números",
    MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

Console.WriteLine("\t\t\t\t\t");

while (resposta == "sim" || resposta == "Sim" || resposta == "SIM" || resposta ==
"s" || resposta == "S")
{
    // Solicita e lê quatro números fornecidos pelo usuário
    Console.Write("\t\t\tInforme o 1º número: ");
    num1 = Double.Parse( Console.ReadLine() );

    Console.Write("\t\t\tInforme o 2º número: ");
    num2 = Double.Parse( Console.ReadLine() );

    Console.Write("\t\t\tInforme o 3º número: ");
    num3 = Double.Parse( Console.ReadLine() );

    Console.Write("\t\t\tInforme o 4º número: ");
    num4 = Double.Parse( Console.ReadLine() );

    Console.Write("\t\t\tInforme uma letra para você ver o quadrado" +
"\t\t\nde um dos quatro números informados: " +
        "\t\t\t\t\tA - Quadrado do 1º número" +
        "\t\t\t\t\tB - Quadrado do 2º número" +
        "\t\t\t\t\tC - Quadrado do 3º número" +
        "\t\t\t\t\tD - Quadrado do 4º número" +
        "\t\t\t\t\tInforme sua opção: ");
    opc = Char.Parse( Console.ReadLine() );

    // Exibe uma linha na tela

    Console.WriteLine("\t\t\t\t\t");

    switch( opc )
    {
        case 'a':
        case 'A':
            nux1 = Math.Pow(num1,2);
            Console.WriteLine("\t\t\tO quadrado de " + num1 + " é: "
+ nux1);

            // Verifica se o quadrado obtido é maior que 10
            if (nux1 >= 10)
                Console.WriteLine (" \t\t\tO número {0} é maior
que 10", nux1);
            else
```

```

        Console.WriteLine ("n\tO número {0} é menor
que 10", nux1);

Console.WriteLine("\t\t_____ \n");

        break;

        case 'b':
        case 'B':
            nux2 = Math.Pow(num2,2);
            Console.WriteLine("n\tO quadrado de " + num2 + " é: "
+ nux2);

            // Verifica se o quadrado obtido é maior que 100
            if (nux2 >= 100)
                Console.WriteLine ("n\tO número {0} é maior
que 100", nux2);
            else
                Console.WriteLine ("n\tO número {0} é menor
que 100", nux2);

Console.WriteLine("\t\t_____ \n");

        break;

        case 'c':
        case 'C':
            nux3 = Math.Pow(num3,2);
            Console.WriteLine("n\tO quadrado de " + num3 + " é: "
+ nux3);

            // Verifica se o quadrado obtido é maior que 1000
            if (nux3 >= 1000)
                Console.WriteLine ("n\tO número {0} é maior
que 1000", nux3);
            else
                Console.WriteLine ("n\tO número {0} é menor
que 1000", nux3);

Console.WriteLine("\t\t_____ \n");

        break;

        case 'd':
        case 'D':
            nux4 = Math.Pow(num4,2);
            Console.WriteLine("n\tO quadrado de " + num4 + " é: "
+ nux4);

            // Verifica se o quadrado obtido é maior que 10000
            if (nux4 >= 10000)
                Console.WriteLine ("n\tO número {0} é maior
que 10000", nux4);
            else
                Console.WriteLine ("n\tO número {0} é menor
que 10000", nux4);

Console.WriteLine("\t\t_____ \n");

        break;

```

```

                                default:
                                    Console.WriteLine("\t\t Opção Inválida, informe um das
letras acima!!!");
                                // Exibe uma linha na tela

Console.WriteLine("\t\t_____ \n");

                                break;
                            } // Fim do switch

                                // Verifica se o usuário deseja continuar
                                Console.Write("\n\t\tDeseja continuar?" + "\n\t\tDigite \"Sim\" para
prosseguir: ");

                                resposta = Console.ReadLine();
                                // Exibe uma linha na tela

Console.WriteLine("\t\t_____ \n");

                                } // Fim do while

                                // Exibe uma linha na tela
                                MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
                                    MessageBoxButtons.OK, MessageBoxIcon.Information);

                                Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

Console.WriteLine("\t\t_____");

                                // Exibe uma linha em branco na tela
                                Console.WriteLine();

                                } // Fim do método Main

                                } // Fim da Classe Números
}

```

13.04 – Peso ideal de uma pessoa com base em dados pessoais

Estrutura de seleção múltipla switch. Calcula o peso ideal de uma pessoa tendo como base:

- * sua altura e seu sexo.
- * utiliza as seguintes fórmulas:
- * para homens : $(72.7 * \text{altura}) - 58$
- * para mulheres : $(62.1 * \text{altura}) - 44.7$

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Peso
{

```

[illegible]

```

                                MessageBoxButtons.OK,
MessageBoxIcon.Information);
                                break;

                                default: // Sexo diferente
                                    MessageBox.Show("Sexo desconhecido!!! " + sexo,
"Verificando seu sexo",
                                MessageBoxButtons.OK,
MessageBoxIcon.Error);
                                break;

                                }

                                // Exibe uma linha na tela

Console.WriteLine("\t\t_____");

                                // Verifica se o usuário deseja continuar pesquisando
Console.Write("\n\t\tDeseja continuar pesquisando?" + "\n\t\t\tDigite
'Sim\' para continuar: ");
                                resposta = Console.ReadLine();

                                // Exibe uma linha na tela

Console.WriteLine("\t\t_____");

                                // Exibe uma linha na tela

Console.WriteLine("\t\t_____");
                                Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

                                // Exibe uma linha em branco na tela
Console.WriteLine();

                                } // Fim do while

                                } // Fim do Método Main

                                } // Fim da Classe Ideal
}

```

13.05 – Recebe informações de produtos: Preço, região, etc

Estrutura de seleção composta (switch). Este programa tem por finalidade receber as seguintes informações do usuário:

- * Preço de um produto
- * Código da região onde o produto foi adquirido

case 2:

```

        Console.WriteLine("\tO preço do produto é de R$" + preco + " e
sua região é a Norte\n");
        break;

    case 3:
        Console.WriteLine("\tO preço do produto é de R$" + preco + " e
sua região é a Leste\n");
        break;

    case 4:
        Console.WriteLine("\tO preço do produto é de R$" + preco + " e
sua região é a Oeste\n");
        break;

    case 5:
    case 6:
        Console.WriteLine("\tO preço do produto é de R$" + preco + " e
sua região é a Nordeste\n");
        break;

    case 7:
    case 8:
    case 9:
        Console.WriteLine("\tO preço do produto é de R$" + preco + " e
sua região é a Sudeste\n");
        break;

    case 10:
    case 11:
    case 12:
    case 13:
    case 14:
    case 15:
    case 16:
    case 17:
    case 18:
    case 19:
    case 20:
    case 21:
    case 22:
    case 23:
    case 24:
        Console.WriteLine("\tO preço do produto é de R$" + preco + " e
sua região é a Centro-Oeste\n");
        break;

    case 25:
    case 26:
    case 27:
    case 28:
    case 29:
    case 30:
    case 31:
    case 32:
    case 33:
    case 34:

```



```

        case 35:
        case 36:
        case 37:
        case 38:
        case 39:
        case 40:
        case 41:
        case 42:
        case 43:
        case 44:
        case 45:
        case 46:
        case 47:
        case 48:
        case 49:
        case 50:
            Console.WriteLine("\tO preço do produto é de R$" + preco + " e
sua região é a Nordeste\n");
            break;

        default: // Emite uma mensagem caso o código de origem seja incorreto.
            Console.WriteLine("\tVocê não informou um código de origem de
produto correto!");
            break;

    } // fim de switch

    // Exibe uma linha na tela

    Console.WriteLine("\t\t_____");
    Console.WriteLine("\t\t\t\t\thttp://www.gupnet.com.br");

    } // Fim do método Main

} // Fim da classe Tipos
}

```

13.06 – Verifica notas informadas

Estrutura de seleção múltipla switch e estrutura de repetição for. Este programa verifica entre 10 notas informadas e quantas são do tipo A, B, C, D e F.

```

using System;
using System.Windows.Forms;

namespace Notas
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Escolares
    {

```

```

/// <summary>
/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main(string[] args)
{
    // Exibe uma linha na tela
    Console.WriteLine("\n\tVerifica entre 10 notas e quantas são do tipo A, B, C, D e
F!!");

    Console.WriteLine("\t_____ \n");

    char nota;    // uma nota informada
    int aCont = 0, // contador de notas A
        bCont = 0, // contador de notas B
        cCont = 0, // contador de notas C
        dCont = 0, // contador de notas D
        fCont = 0; // contador de notas F

    for (int i = 1; i <= 10; i++)
    {
        // Solicita e lê a nota de um aluno
        Console.Write("\n\tInforme a nota do Aluno: ");
        nota = Char.Parse( Console.ReadLine());

        switch ( nota ) // estrutura de seleção múltipla switch
        {
            case 'A':    // a nota é A maiúsculo
            case 'a':    // a nota é a minúsculo
                ++aCont;
                break;

            case 'B':    // a nota é B maiúsculo
            case 'b':    // a nota é b minúsculo
                ++bCont;
                break;

            case 'C':    // a nota é C maiúsculo
            case 'c':    // a nota é c minúsculo
                ++cCont;
                break;

            case 'D':    // a nota é D maiúsculo
            case 'd':    // a nota é d minúsculo
                ++dCont;
                break;

            case 'F':    // a nota é F maiúsculo
            case 'f':    // a nota é f minúsculo
                ++fCont;
                break;

            default: // Verifica todos os outros caracteres informados
                Console.WriteLine("\n\t\t\tNota informada foi incorreta" +
                    "\n\t\t\tA nota não será adicionada aos totais!!!");
        }
    }
}

```

```

        break;

    } // fim da estrutura de seleção múltipla switch

} // fim da estrutura de repetição for

// Exibe os resultados obtidos
MessageBox.Show("As notas informadas foram: " +
    "\n\nA: " + aCont + "\tB: " + bCont + "\tC: " + cCont + "\tD: " + dCont +
"\tF: " + fCont,
    "Verifica as notas informadas num teste com 10 alunos",
    MessageBoxButtons.OK,
    MessageBoxIcon.Exclamation);

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");
Console.WriteLine("\t\t\t\t\thttp://www.gupnet.com.br");

// Exibe uma linha em branco na tela
Console.WriteLine();

} // fim do método Main

} // Fim da Classe Escolares
}

```

13.07 – Valor máximo de gastos de clientes

Estrutura de seleção múltipla switch. Este programa calcula o valor que um cliente poderá gastar em uma loja de eletrodomésticos.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Crédito
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Salarial
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            Console.WriteLine("\n\t\t Calcula o valor que um cliente poderá gastar!!");

            Console.WriteLine("\t\t_____ \n");
        }
    }
}

```

```

// Declaração de Variáveis

double salario,          // Salário do cliente
        mediasal,        // Média salarial do cliente
        credito, // Crédito do cliente
        somasal;          // soma os salários informados do cliente

int      contsal; // Contador de salários do cliente

char classe;

string resposta;

// Inicialização de Variáveis
contsal = 1;
mediasal = 0;
somasal = 0;
credito = 0;

resposta = "sim";

// Verifica se o funcionário deseja continuar verificando a situação do cliente
while ( resposta == "sim" || resposta == "Sim" || resposta == "SIM" || resposta ==
"s" || resposta == "S")
{
    // Estrutura para solicita os salários do cliente
    while ( contsal <= 3)
    {
        // Solicita os 3 últimos salários do cliente
        Console.WriteLine("\n\tInforme o {0}º salário do cliente: R$ ",
contsal);

        salario = Double.Parse( Console.ReadLine() );
        somasal += salario;
        contsal++;
    } // Fim do while

    // Calcula a media salarial do cliente
    mediasal = somasal/3;

    // Exibe a média salarial do cliente
    MessageBox.Show("Média salarial: R$ " + mediasal, "Média dos 3
últimos salários do cliente",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

    /*
    * Média                                     Classe
    Crédito
    * R$ 0,00 ---> R$ 400,00          A          Não tem
    crédito
    * R$ 401,00 -> R$ 600,00          B          10%
    * R$ 601,00 -> R$ 800,00          C          12%
    * R$ 801,00 -> R$ 1000,00         D          15%
    * Acima de --> R$ 1001,00         E          18%
    */
}

```

```

// Solicita e lê a classe salarial do cliente
Console.WriteLine("\n\t\tInforme a classe que o cliente se enquadra, digite: "
+
"\n\t\t\tA - Média salarial até R$ 400,00" +
"\n\t\t\tB - Média salarial de R$ 401,00 ---> R$ 600,00" +
"\n\t\t\tC - Média salarial de R$ 601,00 ---> R$ 800,00" +
"\n\t\t\tD - Média salarial de R$ 801,00 --> R$ 1000,00" +
"\n\t\t\tE - Média salarial acima de R$ 1001,00" +
"\n\t\t\tInforme a opção: ");

classe = Char.Parse( Console.ReadLine() );

// Estrutura de seleção múltipla switch
switch (classe)
{
    case 'a':          // Para clientes que não tem crédito
    case 'A':
        credito = mediasal * 0;
        Console.WriteLine("\n\t\t\tCliente não tem crédito de: R$
{0:C}", credito + "\n");

        break;

    case 'b':
    case 'B':
        credito = mediasal * 0.10;
        Console.WriteLine("\n\t\t\tCliente tem crédito de: R$
{0:C}", credito + "\n");

        break;

    case 'c':
    case 'C':
        credito = mediasal * 0.12;
        Console.WriteLine("\n\t\t\tCliente tem crédito de: R$
{0:C}", credito + "\n");

        break;

    case 'd':
    case 'D':
        credito = mediasal * 0.15;
        Console.WriteLine("\n\t\t\tCliente tem crédito de: R$
{0:C}", credito + "\n");

        break;

    case 'e':
    case 'E':
        credito = mediasal * 0.18;
        Console.WriteLine("\n\t\t\tCliente tem crédito de: R$
{0:C}", credito + "\n");

        break;

    default:
        Console.WriteLine("\n\t\t\tClasse informada inválida!" +
"\n");

        break;
}

```

```

        } // Fim do switch

        // Zera todos os valores para nova pesquisa
        contsal = 1;
        mediasal = 0;
        somasal = 0;

        // Imprime uma linha em branco

        Console.WriteLine("\t\t_____ \n");

        // Verifica se o usuário deseja continuar pesquisando.
        Console.Write("\t\tVocê deseja continuar?" + " Digite \"SIM\" para
prosseguir: \a\a");

        resposta = Console.ReadLine();

    } // fim do while

    // Exibe uma linha na tela

    Console.WriteLine("\t\t_____");
    Console.WriteLine("\t\t\t\t\thttp://www.gupnet.com.br");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

    } // Fim do método Main

    } // Fim da Classe Salarial
}

```

13.08 – Verifica dados com base em classificação

Estrutura de seleção composta switch aninhada em while. Verifica o grau do aço com base nos seguintes dados:

* Um certo aço é classificado de acordo com o resultado de três testes, nos quais são informados:

- * número de amostra,
- * conteúdo de carbono (em %),
- * a dureza Rokwell,
- * e a resistência à tração (em psi).

```

using System;
using System.Windows.Forms;

namespace Metal
{
    /// <summary>
    /// Summary description for Class1.

```

```
// </summary>
class Aço
{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main(string[] args)
    {
        // Exibe uma linha na tela
        Console.WriteLine("\n\t\tVerifica a classificação de amostras de aço!!!");

        Console.WriteLine("\t\t_____ \n");

        // Declaração de Variáveis
        int namostra,          // número da amostra
            dureza,             // grau de dureza (Rokwell)
            resistencia,       // grau de resistência à tração (em psi)
            testes,           // número do teste
            grau;              // Grau final obtido com o Aço

        double quantidade;     // conteúdo de carbono (em %)

        string resposta;

        // Inicialização de Variáveis
        resposta = "sim";

        // Verifica se o usuário deseja continuar pesquisando outras amostragens
        while ( resposta == "sim" || resposta == "Sim" || resposta == "SIM" || resposta ==
"s" || resposta == "S")
        {
            // Solicita e lê o número da amostra
            Console.Write("\n\tInforme o número da amostragem: ");
            namostra = Int32.Parse( Console.ReadLine() );

            // Solicita e lê a porcentagem de carbono
            Console.Write("\n\tInforme a % de carbono da amostra: ");
            quantidade = Double.Parse( Console.ReadLine() );

            // Solicita e lê a dureza do carbono (rokwell)
            Console.Write("\n\tInforme o grau de dureza do carbono: ");
            dureza = Int32.Parse( Console.ReadLine() );

            // Solicita e lê a resistência do carbono a tração
            Console.Write("\n\tInforme o grau de resistência do carbono: ");
            resistencia = Int32.Parse( Console.ReadLine() );

            // Efetuará os testes
            Console.Write("\n\tVerifique o grau do Aço obtido: " +
                "\n\tDigite: " + "\n\t\t1 - Grau 10" +
                "\n\t\t2 - Grau 9" + "\n\t\t3 - Grau 8" +
                "\n\t\t4 - Grau 7" + "\n\t\tInforme sua opção: ");
            testes = Int32.Parse( Console.ReadLine() );

            // Estrutura de seleção múltipla switch
```

```

switch (testes)
{
    case 1: // Ao aço é atribuído o grau 10, se passa pelos três testes
    {
        // Teste 1: Conteúdo de carbono abaixo de 7%
        if ( quantidade < 7)
            Console.WriteLine("\t\t\t\tAprovado no teste
1!");
        else
            Console.WriteLine("\t\t\t\tReprovado no teste
1!");

        // Teste 2: Dureza Rokwell maior que 50
        if (dureza > 50)
            Console.WriteLine("\t\t\t\tAprovado no teste
2!");
        else
            Console.WriteLine("\t\t\t\tReprovado no teste
2!");

        // Teste 3: Resistência à tração maior do que 80.000 psi.
        if ( resistencia > 80000)
            Console.WriteLine("\t\t\t\tAprovado no teste
3!");
        else
            Console.WriteLine("\t\t\t\tReprovado no teste
3!");

        grau = 10;

        MessageBox.Show("Aço com grau: " + grau, "Amostra
aprovada em todos os testes",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);
    }

    break;

    case 2: // Ao aço é atribuído o grau 9, se passa apenas nos
testes 1 e 2
    {
        // Teste 1: Conteúdo de carbono abaixo de 7%.
        if ( quantidade < 7)
            Console.WriteLine("\t\t\t\tAprovado no teste
1!");
        else
            Console.WriteLine("\t\t\t\tReprovado no teste
1!");

        // Teste 2: Dureza Rokwell maior que 50.
        if (dureza > 50)
            Console.WriteLine("\t\t\t\tAprovado no teste
2!");
        else
            Console.WriteLine("\t\t\t\tReprovado no teste
2!");
    }
}

```



```

// Teste 3: Resistência à tração maior do que 80.000 psi.
if ( resistencia > 80000)
    Console.WriteLine("\n\t\t\t\tAprovado no teste
3!");
else
    Console.WriteLine("\n\t\t\t\tReprovado no teste
3!");

    grau = 9;

    MessageBox.Show("Aço com grau: " + grau, "Amostra
    aprovada apenas nos testes 1 e 2",
    MessageBoxButtons.OK,
    MessageBoxIcon.Information);
}
break;

case 3: // Ao aço é atribuído o grau 8, se passa apenas no teste
1
{
    // Teste 1: Conteúdo de carbono abaixo de 7%.
    if ( quantidade < 7)
        Console.WriteLine("\n\t\t\t\tAprovado no teste
1!");
    else
        Console.WriteLine("\n\t\t\t\tReprovado no teste
1!");

    // Teste 2: Dureza Rokwell maior que 50.
    if (dureza > 50)
        Console.WriteLine("\n\t\t\t\tAprovado no teste
2!");
    else
        Console.WriteLine("\n\t\t\t\tReprovado no teste
2!");

    // Teste 3: Resistência à tração maior do que 80.000 psi.
    if ( resistencia > 80000)
        Console.WriteLine("\n\t\t\t\tAprovado no teste
3!");
    else
        Console.WriteLine("\n\t\t\t\tReprovado no teste
3!");

    grau = 8;

    MessageBox.Show("Aço com grau: " + grau, "Amostra
    aprovada apenas nos teste 1",
    MessageBoxButtons.OK,
    MessageBoxIcon.Information);
}
break;

case 4: // // Ao aço é atribuído o grau 7, se não passa em
nenhum dos testes
{
    // Teste 1: Conteúdo de carbono abaixo de 7%.

```

```

1!");
    if ( quantidade < 7)
        Console.WriteLine("\t\t\t\tAprovado no teste

1!");
    else
        Console.WriteLine("\t\t\t\tReprovado no teste

    // Teste 2: Dureza Rokwell maior que 50.
    if (dureza > 50)
        Console.WriteLine("\t\t\t\tAprovado no teste

2!");
    else
        Console.WriteLine("\t\t\t\tReprovado no teste

2!");

    // Teste 3: Resistência à tração maior do que 80.000 psi.
    if ( resistencia > 80000)
        Console.WriteLine("\t\t\t\tAprovado no teste

3!");
    else
        Console.WriteLine("\t\t\t\tReprovado no teste

3!");

    grau = 7;

    MessageBox.Show("Aço com grau: " + grau, "Amostra

reprovada em todos os testes",
    MessageBoxButtons.OK,
    MessageBoxIcon.Information);
}
break;

default: // Teste não encontrado.
    Console.WriteLine("\t\t\t\tInformações incorretas!!!");
    break;

} // Fim do switch

// Verifica se o usuário deseja continuar
Console.Write("\t\tDeseja continuar?" + "\t\t\t\tDigite \"Sim\" para

continuar: ");

resposta = Console.ReadLine();

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");
    Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

// Exibe uma linha em branco na tela
Console.WriteLine();

} // Fim do laço while

} // Fim do método Main

} // Fim da Classe Aço

```

[illegible]

```

// Solicita e lê o operador matemática para efetuar o cálculo
Console.WriteLine("\n\t\tQual operação matemática você deseja efetuar? " +
    "\n\t\t\t\t1 - Adição" + "\n\t\t\t\t2 - Subtração" + "\n\t\t\t\t3 -
Divisão" +
    "\n\t\t\t\t4 - Multiplicação" + "\n\t\t\t\t5 - Exponenciação" +
    "\n\t\t\t\tInforme a opção: ");
OPC = Int32.Parse( Console.ReadLine() );

// Estrutura de seleção switch
switch (OPC)
{
    case 1: // Adição
        RESULTADO = NUM1 + NUM2;
        MessageBox.Show("O resultado é: " + RESULTADO,
"Calculando a soma", MessageBoxButtons.OK, MessageBoxIcon.Information);
        break;

    case 2: // Subtração
        RESULTADO = NUM1 - NUM2;
        MessageBox.Show("O resultado é: " + RESULTADO,
"Calculando a subtração", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        break;

    case 3: // Divisão
        RESULTADO = NUM1 / NUM2;
        MessageBox.Show("O resultado é: " + RESULTADO,
"Calculando a divisão", MessageBoxButtons.OK, MessageBoxIcon.Warning);
        break;

    case 4: // Multiplicação
        RESULTADO = NUM1 * NUM2;
        MessageBox.Show("O resultado é: " + RESULTADO,
"Calculando a multiplicação", MessageBoxButtons.OK, MessageBoxIcon.Stop);
        break;

    case 5: // Exponenciação
        RESULTADO = Math.Pow(NUM1, NUM2);
        MessageBox.Show("O resultado é: " + RESULTADO,
"Calculando a multiplicação", MessageBoxButtons.OK, MessageBoxIcon.None);
        break;

    default: // Considera outras opções
        MessageBox.Show("Opção inválida!!! " + RESULTADO,
"Tente novamente!", MessageBoxButtons.OK, MessageBoxIcon.Error);
        break;
} // Fim do switch

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");

// Efetua novas pesquisas dependendo do resultado informado pelo
usuário

```

```

continuar: ");

        Console.WriteLine("\n\t\tVocê deseja continuar?" + "\tDigite \"SIM\" para

        RESPOSTA = Console.ReadLine();

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");

        Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

        } // Estrutura de repetição while

    } // Fim do método Main

} // Fim da classe Matemática
}

```

13.10 – Comparação entre números

Estrutura de seleção composta (Switch), Simples (if / else), e estrutura de repetição (while). Este programa tem por finalidade apresentar ao usuário qual é o maior número, menor número ou se eles são iguais.

```

using System;

namespace Usuário
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Condição
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            Console.WriteLine("\n\t\tFaz comparação entre 2 números!!!");

            Console.WriteLine("\t\t_____ \n");

            // Declaração de Variáveis

            int            N1,                // Número a ser fornecido pelo usuário
                        N2,                // Número a ser fornecido pelo usuário

```

```

        MAIOR,           // Armazernará o maior número
        MENOR,          // Armazernará o maior número
        OPC;            // Opção de entrada do usuário para ver o maior,
ou menor ou se os números são iguais ou diferentes.

        string  RESP;    /* Resposta do usuário para continuar a verificar a
condição                * dos números*/

// Inicialização de Variáveis

MENOR = 0;
MAIOR = 0;
RESP = "SIM";

// Verifica se o usuário deseja continuar a pesquisa
while (RESP == "SIM" || RESP == "sim" || RESP == "Sim" || RESP == "s" || RESP
== "S" )
{
    // Solicita e lê os números informados pelo usuário
    Console.WriteLine("\t\tInforme o 1º número: ");
    N1 = Int32.Parse(Console.ReadLine());

    Console.WriteLine("\t\tInforme o 2º número: ");
    N2 = Int32.Parse(Console.ReadLine());

    // Pula uma linha
    Console.WriteLine();

    Console.WriteLine("\t\tInforme: 1 para ver o maior número\n" + "\t\tInforme: 2
para ver o menor número\n" + "\t\tInforme: 3 para ver se os números são iguais---> ");
    OPC = Int32.Parse(Console.ReadLine());

    // Pula uma linha
    Console.WriteLine();

    switch (OPC) // Estrutura de seleção switch
    {
        case 1: // Maior número
            if (N1 == N2)
                Console.WriteLine("\n\t\tOs números
informados são iguais!\n\n");

            else
            {
                if (N1 > N2)
                {
                    MAIOR = N1;
                    MENOR = N2;
                    Console.WriteLine("\t\tO Maior número é:
{0}\n", MAIOR);

                }
                else
                {
                    MAIOR = N2;
                    MENOR = N1;

```

```

        Console.WriteLine("\t\tO Maior número é:
{0}\n", MAIOR);
    }
}
break;

case 2: // Menor número

    if (N1 == N2)
        Console.WriteLine("\n\t\tOs números
informados são iguais!\n\n");
    else
    {
        if (N2 > N1)
        {
            MAIOR = N2;
            MENOR = N1;
            Console.WriteLine("\t\tO Menor
número é: {0}\n", MENOR);
        }
        else
        {
            MAIOR = N1;
            MENOR = N2;
            Console.WriteLine("\t\tO Menor
número é: {0}\n", MENOR);
        }
    }
    break;

case 3: // Iguais
    if (N2 == N1)
    {
        Console.WriteLine("\t\tOs números {0} e {1} são
iguais.\n", N1, N2);
    }
    else
    {
        Console.WriteLine("\t\tOs números {0} e {1} são
diferentes.\n", N1, N2);
    }
    break;

default: // Verifica se a opção está entre 1, 2 ou 3
    Console.WriteLine("\t\tVocê não informou uma opção
correta!\n");
    break;

} // fim da estrutura de seleção composta switch

// Exibe uma linha na tela

Console.WriteLine("\t\t_____ \n");

// Solicita ao usuário para verificar se o mesmo deseja continuar a
executar o programa

```

```
        Console.WriteLine("\t\tDeseja continuar? " + "Digite \"SIM\" para continuar: ");
        RESP = Console.ReadLine();

        // Pula uma linha
        Console.WriteLine();

    } // Fim do While

    // Exibe uma linha na tela

    Console.WriteLine("\t\t_____");
    Console.WriteLine("\t\t\t\t\thttp://www.gupnet.com.br");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

    } // Fim do método Main

} // Fim da classe condição
}
```


14.01 – Calcula uma série

* $S = 1/1 + 3/2 + 5/3 + 7/4 + \dots + 99/50$

```
using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Somatório
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Verifique o código do programa para aprender do/while!",
                "Calcula uma série de números....",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\t\t_____ \n");

            // Inicialização e Declaração de Variáveis
            int numerador = 1, denominador = 1;
            double s = 0, parc;

            do
            {
                // Efetuando um cast...
                parc = (double)numerador/denominador;
                s += parc;
                numerador += 2;
                denominador++;

            }while (denominador <= 50);

            //Exibe o resultado
```



```

        Console.WriteLine("\n\t\tData: " + DateTime.Now);

        // Exibe uma linha na tela
        MessageBox.Show("Calcula o reajuste salarial", "Informe os valores para o
cálculo",
                        MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

        Console.WriteLine("\t\t_____ \n");

        // Declaração e Inicialização de Variáveis
        int codfunc, // código do funcionário
            contf = 1; // contador de funcionários

        double salario, // Salário do funcionário
            reajuste = 0, // Reajuste salarial
            salarion = 0, // Novo salário
            reajustet = 0; // Reajuste total com todos os funcionários

        string nome; // Nome do funcionário
        string mensagem;
        string resposta = "sim";

        // Estrutura de repetição do/while
        do
        {
            // Solicita e lê os dados do funcionário
            do
            {
                Console.Write("\n\t\tDigite os dados do {0}º funcionário: ", contf);

                Console.Write("\n\n\t\t\t\tInforme o código: ");
                codfunc = Int32.Parse(Console.ReadLine());

                Console.Write("\n\t\t\t\tInforme o nome: ");
                nome = Console.ReadLine();

                Console.Write("\n\t\t\t\tInforme o salário: R$ ");
                salario = Double.Parse(Console.ReadLine());

                // Exibe uma linha na tela

                Console.WriteLine("\n\t\t_____ \n");

                // Compara o salário dos funcionários para o cálculo
                // do reajuste salarial

                if (salario > 25000)
                {
                    reajuste = (salario * (0.2));
                    salarion = salario + reajuste;

                    mensagem = "Dados do funcionário:\n\n\tCódigo: " +
                                "\n\tNome: " + String.Format("{0}", nome) +
                                "\n\tSalário de: " + String.Format("{0:c}", salario) +

```

```

String.Format("{0:c}", reajuste) +
String.Format("{0:c}", salarion);

funcionário...",
MessageBoxIcon.Information);

}
else
{
    if (salario >= 10000)
    {
        reajuste = (salario * (0.3));
        salarion = salario + reajuste;

        mensagem = "Dados do funcionário:\n\n\tCódigo:

        \n\tNome: " + String.Format("{0}", nome)

        \n\tSalário de: " +

        \n\tReajuste salarial de: " +

        \n\tSalário reajustado de: " +

        MessageBox.Show(mensagem, "Verificando os

        MessageBoxButtons.OK,

    }
    else
    {
        reajuste = (salario * (0.55));
        salarion = salario + reajuste;

        mensagem = "Dados do funcionário:\n\n\tCódigo:

        \n\tNome: " + String.Format("{0}", nome)

        \n\tSalário de: " +

        \n\tReajuste salarial de: " +

        \n\tSalário reajustado de: " +

        MessageBox.Show(mensagem, "Verificando os

        MessageBoxButtons.OK,

    }
}

// Calcula o valor do reajuste global

```

```

        reajustet += reajuste;
        contf++;

    }while(contf <= 5);

    // Exibe o resultado do reajuste global
    Console.WriteLine("\n\t\tO reajuste total é de: {0:c}", reajustet);

    // Exibe uma linha na tela
    Console.WriteLine("\t\t\t\t\t_____");

    // Reinicialização dos valores (funcionários e reajuste total)
    contf = 1;
    reajustet = 0;

    // Verifica se o usuário deseja continuar
    Console.Write("\n\t\tVocê deseja continuar?, digite \"sim\" para
prosseguir: ");

    resposta = Console.ReadLine();

    // Exibe uma linha na tela

    Console.WriteLine("\t\t_____");

    }while(resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==
"S" || resposta == "s");

    // Exibe uma linha na tela
    MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

    Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
    Console.WriteLine("\t\t\t\t\thttp://www.gupnet.com.br");

    Console.WriteLine("\t\t_____");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

    } // Fim do Método Main

    } // Fim da Classe Salarial
}

```

14.03 – Calcula número “perfeito”

Estrutura de repetição do/while aninhadas. Verifica se um número é perfeito ou não.

* Obs: Número perfeito é aquele cuja soma de seus divisores, exceto ele próprio, é igual a ele mesmo.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Número
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Perfeito
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Efetua cada pesquisa para 5 números inteiros...!", "Verifica
se um número é perfeito ou não!",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\n\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            int num, // Número a ser informado pelo usuário
                cont = 1,
                cont1 = 0, // Conta quantas vezes o laço é executado
                cont2 = 1, // Contador de divisores
                soma = 0, // Soma dos restos das divisões
                div = 0; // pega divisor

            double resto = 0;
            string resposta = "sim";

            // Verificará quantas vezes o usuário desejará efetuar a pesquisa
            do
            {
                // Solicita e lê um número informado pelo usuário
                Console.Write("\t\tInforme um número: ");
                num = Int32.Parse( Console.ReadLine() );

                // Executará o laço 5 vezes
                do
                {
                    /* Somará os divisores que tem resto igual a zero enquanto
                    * eles forem menor que o número informado */
                    do
                    {
                        resto = num % cont;
                        if (resto == 0)

```

```

        {
            if (cont < num)
            {
                div = cont;
                // Exibe o seu divisor quando o resto for
                zero
                Console.WriteLine("\n\t\t\t\tO {0}º divisor
do número {1} é: {2}", cont2, num, div);

                cont2++;
                soma += div;
            }
        }
        cont++;
    }while(num >= cont);

    // Exibe a soma dos divisores
    Console.WriteLine("\n\t\t\t\t\tA soma dos divisores é: {0}", soma);
    Console.WriteLine("\t\t\t\t\t_____ \n");

    if (soma == num)
    {
        Console.WriteLine("\n\t\t\t\t\tO número é perfeito!");
        // Exibe a data e hora!

        Console.WriteLine("\t\t\t\t\t_____ \n");
    }
    else
    {
        Console.WriteLine("\n\t\t\t\t\tO número não é perfeito!");
        // Exibe a data e hora!

        Console.WriteLine("\t\t\t\t\t_____ \n");
    }

    // Reinicialização de Variáveis
    cont = 1;
    cont2 = 1;
    soma = 0;
    num++;
    cont1++;

}while(cont1 < 5);

cont1 = 0;
// Solicitará ao usuário para continuar pesquisando números perfeitos e
imperfeitos!

Console.Write("\n\t\t\t\tVocê deseja continuar verificando números?" +
"\n\t\t\t\tDigite \'Sim\' para pesquisar: ");
resposta = Console.ReadLine();

// Exibe uma linha na tela

Console.WriteLine("\n\t\t\t\t_____");

}while(resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==
"S" || resposta == "s");

```

```

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
            MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
        Console.WriteLine("\t\t\t\t http://www.gupnet.com.br");

Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();
    } // Fim do Método Main

} // Fim da Classe Perfeito
}

```

14.04 – Lê notas escolares e efetua cálculos

Estrutura de repetição do/while. Lê 10 notas escolares e calcula a média aritmética dessas notas.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Notas
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Escolares
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            MessageBox.Show("Você deverá informar 10 notas!", "Calcula a média
aritmética!",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

Console.WriteLine("\t\t_____\\n");

```



```

// Declaração e Inicialização de Variáveis
double nota, // Uma nota escolar
        soma = 0, // Soma as notas obtidas
        media = 0; // Média das notas escolares

int contador = 0;           // Contador de notas fornecidas

do
{
    Console.WriteLine("\t\tInforme a {0}ª nota do aluno: ", contador+1);
    nota = Double.Parse( Console.ReadLine() );
    soma += nota;
    contador++;
} while (contador < 10);

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");

// Calcula a média aritmética das notas obtidas
media = soma/10;

// Exibe o resultado
MessageBox.Show("A média das 10 notas é: " + media, "Calculando a média...",
MessageBoxButtons.OK, MessageBoxIcon.Information);

// Exibe uma linha na tela
MessageBox.Show("\thttp://www.gupnet.com.br", "Você já deu uma espiadinha
no site do Gup .Net hoje?",
        MessageBoxButtons.OK, MessageBoxIcon.Question);

Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET\n");
Console.WriteLine("\t\t\thttp://www.gupnet.com.br");

Console.WriteLine("\t\t_____");

// Exibe uma linha em branco na tela
Console.WriteLine();

    }
}
}

```

14.05 – Calcula uma expressão

Estrutura de repetição do/while. Calcula uma expressão da seguinte forma:

$$* S = 1/1 - 2/4 + 3/9 - 4/16 + 5/25 - 6/36 + \dots - 10/100$$

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

```

```

namespace Valor
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Expressão
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Neste programa você só verá o resultado final!", "Analise o
código do programa!",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            int num = 1,    // numerador
                m = 1,      // inversor de sinal
                cont = 0;
            double s = 0,    // soma de todas as parcelas da série
                parc;        // cada parcela (fração separadas)

            // Estrutura de repetição do/while
            do
            {
                parc = (num/(Math.Pow(num,2)))*m;
                m *= -1;
                s += parc;
                num += 1;
                cont++;
            }while (num <= 10); // Fim do laço de repetição
            // O laço de repetição do/while sempre ocorrerá uma vez!

            // Exibe o resultado
            Console.WriteLine("\n\t\tA soma dos {0} termos da série é: {1:n}", cont, s + "\n" );

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");

            // Exibe uma linha na tela
            MessageBox.Show("        http://www.gupnet.com.br",
                "Você já deu uma espiadinha no site do Gup .Net hoje?",
                MessageBoxButtons.OK, MessageBoxIcon.Question);

            Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
            Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

```



```

        string mensagem = "Parcela\t\tValor\t\tSomatório\n\n";

        // Solicita e lê um número
        Console.Write("\n\t\tInforme um número: ");
        x = Double.Parse( Console.ReadLine());

        // Exibe uma linha na tela

Console.WriteLine("\t\t_____");

        // Estrutura de repetição do/while
        do
        {
            parc = ((Math.Pow(x,exp))/contador)*m;
            s += parc;

            mensagem += String.Format("{0}",contador) + "\t\t" +
                String.Format("{0:n}",parc) + "\t\t" +
                String.Format("{0:n}",s) + "\n";
            m *= -1;
            exp--;
            contador++;

        }while( contador <=25);

        // Exibe o resultado
        MessageBox.Show(mensagem,"Calculando a
série",MessageBoxButtons.OK,MessageBoxIcon.Information);

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
            MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
        Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();
    } // Fim do Método Main

    } // Fim da classe Somatório
}

```

14.07 – Volume de uma esfera

Estrutura de repetição do / while. Calcula o volume de uma esfera em função da medida do seu raio. O raio deverá variar de 0 a 20 cm de 0.5 em 0.5.

```

using System;
using System.Windows.Forms;

// Adicione uma referência para MessageBox.Show

namespace Volume
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Esfera
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Calcula o volume de uma esfera!", "Preste atenção nesse
código!",
                            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\n\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            int contador = 1;

            double raio = 0, volume = 0;

            string mensagem = "Raio\t\tVolume\n\n";

            do
            {
                // Calcula o volume da esfera
                volume = 4 * Math.PI * Math.Pow(raio, 3) / 3;
                raio += 0.5;
                contador++;
                mensagem += raio + "\t\t" + String.Format("{0:n}", volume) + "\n";

                // Exibe o resultado do volume da esfera a cada iteração do laço
                MessageBox.Show(mensagem, "Efetuando os cálculos para o volume",
                                MessageBoxButtons.OK, MessageBoxIcon.Information);

            } while (contador <= 20);

            // Exibe uma linha na tela
            MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
                            MessageBoxButtons.OK, MessageBoxIcon.Information);

            Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
            Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

```



```

        MessageBox.Show("Os números são: " + counter, "Veja os
números",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        counter++;

    }while ( counter <= 10);

    // Exibe uma linha na tela

Console.WriteLine("\t_____");
);

        Console.Write("\n\tDeseja continuar?" + "\n\t\tDigite \"Sim\" para
prosseguir: ");

        resposta = Console.ReadLine();

        // Exibe uma linha na tela

Console.WriteLine("\t_____ \
n");

        counter = 1;

    } while ( resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta
== "S" || resposta == "s");

        Console.WriteLine("\n");

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

Console.WriteLine("\t_____");
);

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do Método Main

} // Fim da Classe TesteDoWhile
}

```

14.09 – Calcula média de um aluno

Estrutura de repetição do /while. Calcula a média de um número de alunos informado pelo usuário...

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Notas
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Escola
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Informe as notas de um conjunto de alunos!", "Calcula a
média ponderada...",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\n\t\t_____ \n");

            int alunos, // número de alunos
                codalu, // código do aluno (matrícula)
                contal = 1; // contador de alunos

            double n1, n2, n3, // notas escolares de um aluno
                maior = 0, // maior nota do aluno
                medio = 0, // segunda nota maior nota do aluno
                menor = 0, // menor nota do aluno
                mediap = 0; // média ponderada do aluno

            string resposta = "sim";

            do
            {
                // Solicita e lê o número total de alunos
                Console.Write("\n\t\tInforme o número total de alunos: ");
                alunos = Int32.Parse( Console.ReadLine());

                do
                {
                    // Solicita e lê a matrícula do aluno
                    Console.Write("\n\t\tInforme a matrícula do {0}º aluno: ", contal);
                    codalu = Int32.Parse( Console.ReadLine() );

                    // Solicita e lê as notas escolares de um aluno
                    Console.Write("\n\n\t\t\t\tInforme a 1º nota do aluno: ");

```



```

n1 = Double.Parse( Console.ReadLine());

Console.WriteLine("\n\t\t\t\tInforme a 2º nota do aluno: ");
n2 = Double.Parse( Console.ReadLine());

Console.WriteLine("\n\t\t\t\tInforme a 3º nota do aluno: ");
n3 = Double.Parse( Console.ReadLine());

// Verificará qual a maior nota, a nota média e a menor nota.
if (n1 >= n2 && n1 >= n3)
{
    maior = n1;
    if (n2 >= n3)
    {
        medio = n2;
        menor = n3;
    }
    else
    {
        medio = n3;
        menor = n2;
    }

    // Calcula a Média Ponderada do aluno
    mediap = (((maior*4) + (medio*3) + (menor*3))/(4+3+3));
    MessageBox.Show("A média do aluno é: "+ mediap,
"Calculando a Média Ponderada",
MessageBoxButtons.OK,MessageBoxIcon.Information);
}
else
{
    if (n2 >= n1 && n2 >= n3)
    {
        maior = n2;
        if (n1 >= n3)
        {
            medio = n1;
            menor = n3;
        }
        else
        {
            medio = n3;
            menor = n1;
        }

        // Calcula a Média Ponderada do aluno
        mediap = (((maior*4) + (medio*3) + (menor*3))/(
(4+3+3));
        MessageBox.Show("A média do aluno é: "+
mediap, "Calculando a Média Ponderada",
MessageBoxButtons.OK,MessageBoxIcon.Information);
    }
    else
    {

```

```

        maior = n3;
        if (n1 >= n2)
        {
            medio = n1;
            menor = n2;
        }
        else
        {
            medio = n2;
            menor = n1;
        }

        // Calcula a Média Ponderada do aluno
        mediap = (((maior*4) + (medio*3) + (menor*3))/
(4+3+3));

        MessageBox.Show("A média do aluno é: "+
mediap, "Calculando a Média Ponderada",
MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}
// Verifica se o aluno foi aprovado ou não
if (mediap >= 5)
{
    Console.WriteLine("\n\n\t\t\t\t\tO aluno {0} foi aprovado!
", codalu);

    // Exibe uma linha na tela

    Console.WriteLine("\t\t\t\t\t_____");

    // Exibe uma linha na tela

    Console.WriteLine("\t\t_____");
}
else
{
    Console.WriteLine("\n\n\t\t\t\t\tO aluno {0} foi reprovado!
", codalu);

    // Exibe uma linha na tela

    Console.WriteLine("\t\t\t\t\t_____");

    // Exibe uma linha na tela

    Console.WriteLine("\t\t_____");
}

    contal++;    // Incrementa o contador de alunos

}while(contal <= alunos);

// Reinicializamdo o contador de alunos
contal = 1;

// Verifica se o professor deseja continuar a pesquisar outros alunos
Console.Write("\n\t\tDeseja continuar?, Digite \"Sim\" para prosseguir: ");

```

```

        resposta = Console.ReadLine();

        // Exibe uma linha na tela

Console.WriteLine("\t\t_____");

        }while(resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==
"S" || resposta == "s");

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\n\t\t Grupo de Usuários do Paraná - Plataforma .NET");
        Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do Método Main

} // Fim da Classe Escola
}

```

14.10 – Calcula uma expressão

Estrutura de repetição do/while aninhada em do/while. Calcula uma expressão com base em um número fornecido pelo usuário da seguinte forma:

* $H = 1/1 + 1/2 + 1/3 + 1/4 + 1/5 + 1/6 + \dots + 1/N$

* N ---> deverá ser lido pelo usuário...

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Valor
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Somatório
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!

```

```

        Console.WriteLine("\n\t\tData: " + DateTime.Now);

        // Exibe uma linha na tela
        MessageBox.Show("Neste programa você só verá o resultado final!", "Análise o
código do programa!",
                        MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

        Console.WriteLine("\n\t\t_____ \n");

        // Declaração e Inicialização de Variáveis
        int cont = 1;

        double parc, // Parcela de cada fração
                h = 0; // Soma de todos os membros da série

        string resposta = "sim";

        // Estrutura de repetição do/while
        do
        {
            // Solicita e lê um número do usuário
            Console.Write("\n\t\tInforme um número: ");
            int num = Int32.Parse(Console.ReadLine());

            // Estrutura de repetição do/while
            /* Executará o laço até que o contador seja igual ao número informado
            pelo usuário */
            do
            {
                // Efetuando um cast
                parc = (double)1/cont;
                h += parc;
                cont++;
            }while(cont <= num);
            // A estrutura sempre será executada ao menos uma vez!

            // Exibe o resultado
            Console.WriteLine("\n\t\tA soma dos {0} termos da série é: {1:n}", num,
h);

            // Exibe uma linha na tela

        Console.WriteLine("\n\t\t_____");

            // Prepara os valores para o reinício do looping
            cont = 1;
            h = 0;
            // Verifica se o usuário deseja continuar
            Console.Write("\n\t\tDeseja continuar?" + "\n\t\t\tDigite \"Sim\" para
prosseguir: ");

            resposta = Console.ReadLine();

            // Exibe uma linha na tela

        Console.WriteLine("\n\t\t_____");

```

```
        }while(resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==  
"s" || resposta == "S");  
  
        // Exibe uma linha na tela  
        MessageBox.Show("                http://www.gupnet.com.br",  
            "Você já deu uma espiadinha no site do Gup .Net hoje?",  
            MessageBoxButtons.OK, MessageBoxIcon.Question);  
  
        Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");  
        Console.WriteLine("\t\t\t\t http://www.gupnet.com.br");  
  
        Console.WriteLine("\t\t_____");  
  
        // Exibe uma linha em branco na tela  
        Console.WriteLine();  
  
    } // Fim do Método Main  
  
} // Fim da Classe Expressão  
}
```

Exercícios 15 – Estrutura de Repetição Do /While e outras

15.01 – Soma de termos de uma série

Estrutura de repetição do/while aninhada em do/while. Calcula a seguinte soma dos seguintes termos da série:

$$* S = 1000/1 - 997/2 + 994/3 - 991/4 + \dots$$

```
using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Série
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Informe a quantidade de termos da expressão!", "Calcula o somatório de vários termos!",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\n\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            double s = 0, // soma dos termos
                n, // número de parcelas
                parc; // fração

            int num = 1000, // numerador da fração
                den = 1, // denominador da fração
                m = 1, // inversor de sinal
                cont = 1; // contador de parcelas

            string resposta = "sim",
                mensagem = "Parcela\t\t\tSoma\n\n";

            // Estrutura de repetição do / while
            do
```

```
{
    // Solicita e lê a quantidade de termos da série
    Console.WriteLine("\n\tInforme a quantidade de termos: ");
    n = Int32.Parse(Console.ReadLine());

    // Exibe uma linha na tela
    Console.WriteLine("\t\t_____");

    // Estrutura de repetição do while calculando a soma e as parcelas
    do
    {
        parc = (num/den)* m;
        s += parc;
        m *= -1;
        num -= 3;
        den += 1;
        mensagem += cont + "º = " + String.Format("{0:n}",parc) + "\t\t" +
            String.Format("{0:n}", s) + "\n";
        cont++;
    }while (cont <= n);

    // Exibe o resultado
    MessageBox.Show(mensagem, "Calculando a soma dos termos " + n + "
termos",
                    MessageBoxButtons.OK, MessageBoxIcon.Information);

    // Reinicialização de Variáveis
    cont = 1;
    s = 0;
    parc = 0;
    num = 1000;
    den = 1;
    m = 1;
    mensagem = "Parcela\t\tSoma\n\n";

    // Verifica se o usuário deseja continuar
    Console.WriteLine("\n\tDeseja continuar?" + "\n\t\tDigite \"Sim\" para
prosseguir: ");

    resposta = Console.ReadLine();

    // Exibe uma linha na tela
    Console.WriteLine("\t\t_____");

    }while( resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==
"S" || resposta == "s");

    // Exibe uma linha na tela
    MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
                    MessageBoxButtons.OK, MessageBoxIcon.Information);

    Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
    Console.WriteLine("\t\t\t\t\t http://www.gupnet.com.br");
```

```

Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do Método Main

} // Fim da Classe Números
}

```

15.02 – Pesquisa de Mercado

Estrutura de repetição do/while. Este programa efetua uma pesquisa de mercado verificando se as pessoas gostaram ou não de um produto lançado no mercado. Para isso, deverá ser fornecido o sexo e sua resposta (sim ou não), sendo entrevistados 200 pessoas, deverá ser calculado:

- * o número de pessoas que responderam sim.
- * o número de pessoas que responderam não.
- * a porcentagem de pessoas do sexo feminino que responderam sim.
- * a porcentagem de pessoas do sexo masculino que responderam não.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Pesquisa
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Produto
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Verifique a situação da pesquisa", "Pesquisa de satisfação
de produto...",
                            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\t\t_____ \n");

```



```
//Declaração e Inicialização de Variáveis
int entrevistados, // número de pessoas entrevistadas
    mulheres = 0, // número de mulheres entrevistadas
    homens = 0, // número de homens entrevistados
    nsim = 0, // número de pessoas que disseram sim
    nnao = 0, // número de pessoas que disseram não
    cont = 1, // contador de pessoas
    chn = 0, // contador de homens que disseram não
    cms = 0; // contador de mulheres que disseram sim

double phn = 0, // porcentagem de homens que disseram não
    pms = 0; // porcentagem de mulheres que disseram sim

string sexo, // Sexo do entrevistado
    resposta, // resposta do entrevistado quanto ao produto
    resposta2 = "sim"; // resposta do usuário para continuar a
verificar a pesquisa

// Estrutura de repetição do / while
do
{
    // Solicita e lê a quantidade de pessoas entrevistadas
    Console.WriteLine("\n\tInforme a quantidade de entrevistados: ");
    entrevistados = Int32.Parse( Console.ReadLine());

    // Exibe uma linha na tela
    Console.WriteLine("\t\t_____");

    // Estrutura de repetição do / while
    do
    {
        // Solicita e lê os dados do entrevistado
        Console.WriteLine("\n\tDigite os dados do {0}º entrevistado: ", cont);
        Console.WriteLine("\n\t\t\t\t\tSexo: ");
        sexo = Console.ReadLine();

        Console.WriteLine("\n\t\t\t\t\tResposta: ");
        resposta = Console.ReadLine();

        // Exibe uma linha na tela
        Console.WriteLine("\t\t_____");

        // Verifica o número de pessoas que disseram sim ou não
        if ( resposta == "SIM" || resposta == "Sim" || resposta == "sim" ||
resposta == "S" || resposta == "s")
            nsim++;
        else
            nnao++;

        // Verifica o número de pessoas do sexo masculino ou feminino
        if (sexo == "MASCULINO" || sexo == "Masculino" || sexo ==
"masculino" || sexo == "M" || sexo == "m")
            homens++;

```

```

else
    mulheres++;

    // Verifica o número de mulheres que disseram sim
    if ((sexo == "FEMININO" || sexo == "Feminino" || sexo ==
"feminino" || sexo == "F" || sexo == "f") && (resposta == "SIM" || resposta == "Sim" || resposta == "sim" ||
resposta == "S" || resposta == "s"))
        cms++;

    // Verifica o número de homens que disseram não
    if ((sexo == "MASCULINO" || sexo == "Masculino" || sexo ==
"masculino" || sexo == "M" || sexo == "m") && (resposta == "NÃO" || resposta == "Não" || resposta ==
"não" || resposta == "N" || resposta == "n"))
        chn++;

    // Incrementa o contador de pessoas entrevistadas
    cont++;

}while (cont <= entrevistados); // Fim do / while

if (mulheres > 0)
{
    pms = cms * 100/mulheres;
    Console.WriteLine("\n\t\tA porcentagem de mulheres que
disseram sim é: {0:n} %", pms);

    // Exibe uma linha na tela

    Console.WriteLine("\t\t_____");
}

if (homens > 0)
{
    phn = chn * 100/homens;
    Console.WriteLine("\n\t\tA porcentagem de homens que disseram
não é: {0:n} %", phn);

    // Exibe uma linha na tela

    Console.WriteLine("\t\t_____");
}

// Exibe o resultado
Console.WriteLine("\n\t\tO número de pessoas que disseram sim foi de: "
+ nsim);

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");

Console.WriteLine("\n\t\tO número de pessoas que disseram não foi de: "
+ nnao);

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");

```

```

        // Solicita e lê a resposta do usuário para continuar pesquisando
        Console.WriteLine("\n\t\tDeseja continuar pesquisando?" +
            "\n\t\t\tDigite \"Sim\" para prosseguir: ");
        resposta2 = Console.ReadLine();

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");

        // Reinicializando as variáveis para o novo laço
        cont = 1;
        phn = 0;
        pms = 0;
        mulheres = 0;
        homens = 0;
        nsim = 0;
        nnao = 0;
        chn = 0;
        cms = 0;

        }while(resposta2 == "SIM" || resposta2 == "Sim" || resposta2 == "sim" || resposta2
        == "S" || resposta2 == "s");

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
            MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
        Console.WriteLine("\t\t\t\t\thttp://www.gupnet.com.br");

        Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

        } // Fim do Método Main

    } // Fim da classe Produtos
}

```

15.03 – Bônus salarial

Estrutura de repetição do/while.

Cálculo de bônus salarial de 3 funcionários do seguinte modo:

* os funcionários do sexo masculino com tempo de casa superior a 15 anos terão direito a um bônus de 20% de seu salário.

* as funcionárias com tempo de serviço superior a 10 anos terão direito a um bônus de 25% de seu salário.

* os demais funcionários terão direito a um bônus de R\$ 5.000,00

```
using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Bonus
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Salarial
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Verifique o seu bônus!", "Cálcula o bônus salarial de alguns
funcionários",
                            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\n\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            string sexo;
            string mensagem = "Os dados do funcionário(a) são:\n ";
            string resposta = "sim";

            int c_anos, // tempo de casa do funcionário
                n_func = 3, // número de funcionários
                cont = 1; // contador de funcionários

            double salario, // Salário do funcionário
                salarion = 0, // Novo salário do funcionário
                bonus = 0, // Bônus do funcionário
                montante = 0; // Montante total

            // Estrutura de repetição do/while
            do
            {
                do
                {
                    // Solicita e lê os dados dos funcionários
                    Console.WriteLine("\n\t\tInforme os dados do {0}º funcionário(a): ",
cont);

                    Console.WriteLine("\n\t\t\tSexo: ");
                    sexo = Console.ReadLine();

                    Console.WriteLine("\n\t\t\tTempo de casa: ");
```

```

        c_anos = Int32.Parse(Console.ReadLine());

        Console.WriteLine("\n\t\tSalário: R$ ");
        salario = Double.Parse(Console.ReadLine());

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");

        // Bônus salarial de 20%
        if (( sexo == "MASCULINO" || sexo == "Masculino" || sexo ==
"masculino" || sexo == "M" || sexo == "m") && (c_anos > 15))
        {
            bonus = salario * 0.20;
            salarion = salario + bonus;
            mensagem = "\n\tSexo: " + String.Format("{0}", sexo) +
"\n\tTempo de Serviço: " + String.Format("{0}",
c_anos) +
"\n\tSalário: " + String.Format("{0:c}", salario) +
"\n\tBônus: " + String.Format("{0:c}", bonus) +
"\n\tSalário reajustado: " + String.Format("{0:c}",
salarion);

            MessageBox.Show(mensagem, "Calculando o bônus
salarial...",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);
        }
        else
        { // Bônus salarial de 25%
            if (( sexo == "FEMININO" || sexo == "Feminino" || sexo
== "feminino" || sexo == "F" || sexo == "f") && (c_anos > 10))
            {
                bonus = salario * 0.25;
                salarion = salario + bonus;
                mensagem = "\n\tSexo: " + String.Format("{0}",
sexo) +
"\n\tTempo de Serviço: " +
String.Format("{0}", c_anos) +
"\n\tSalário: " + String.Format("{0:c}",
salario) +
"\n\tBônus: " + String.Format("{0:c}",
bonus) +
"\n\tSalário reajustado: " +
String.Format("{0:c}", salarion);

                MessageBox.Show(mensagem, "Calculando o
bônus salarial...",
                MessageBoxButtons.OK,
                MessageBoxIcon.Information);
            }
            else
            { // Bônus salarial de R$ 5000,00
                bonus = 5000.00;
                salarion = salario + bonus;
                mensagem = "\n\tSexo: " + String.Format("{0}",
sexo) +

```

```
String.Format("{0}", c_anos) +  
    salario) +  
    bonus) +  
    String.Format("{0:c}", salarion);  
  
    bônus salarial...",  
    MessageBoxIcon.Information);  
  
        }  
    }  
  
    // Cálculo do reajuste salarial  
    montante += bonus;  
    cont++;  
  
}while(cont <= n_func);  
  
// Exibe o valor total gasto com o bônus para todos os funcionários  
  
Console.WriteLine("\n\t\t\t\tBônus total de: {0:c}", montante);  
  
Console.WriteLine("\t\t\t\t\t_____ \n");  
  
// Verifica se o usuário deseja continuar  
Console.Write("\n\t\tDeseja continuar?, \"Sim\" para prosseguir: ");  
resposta = Console.ReadLine();  
  
// Exibe uma linha na tela  
  
Console.WriteLine("\t\t_____");  
  
// Reinicialização dos valores  
montante = 0;  
cont = 1;  
  
}while(resposta == "Sim" || resposta == "sim" || resposta == "SIM" || resposta ==  
"S" || resposta == "s");  
  
Console.WriteLine("\t\t_____");  
  
// Exibe uma linha na tela  
MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",  
    MessageBoxButtons.OK, MessageBoxIcon.Information);  
  
Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");  
Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");  
  
Console.WriteLine("\t\t_____");  
  
// Exibe uma linha em branco na tela  
Console.WriteLine();
```

```

        } // Fim do Método Main

    } // Fim da Classe Salarial
}

```

15.04 – Menor número

Estrutura de repetição do/while. Verifica qual o menor número lido.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Menor
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Valor
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Informe um número!", "Verifica qual o menor valor lido",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\n\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            double menor, // Menor valor lido
                numero;    // Número a ser informado pelo usuário

            string resposta = "sim";

            do
            {
                // Solicita e lê um número do usuário
                Console.Write("\n\t\t\tInforme um número, digite \"-1\" para sair: ");
                numero = Double.Parse(Console.ReadLine());

                menor = numero;

                // Estrutura de repetição do/while
                do

```



```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Impulsos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Telefone
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Vamos verificar seu telefone...", "Verifica a quantidade de
impulsos por assinante.",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\n\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            int impulsos, // Contém o número de impulsos por assinante
                quantos, // contém quantos assinantes atingiram o maior número
de impulsos
                maiorpulsos; // conterà o maior número de impulsos encontrados

            quantos = 0;

            // Solicita e lê a quantidade de impulsos por assinante
            Console.Write("\n\t\tInforme a quantidade de impulsos: ");
            impulsos = Int32.Parse(Console.ReadLine());

            maiorpulsos = impulsos; // Supondo que o primeiro já é o maior

            do
            {
                if (impulsos > maiorpulsos)
                {
                    maiorpulsos = impulsos;
                    quantos = 1;
                }
                else if (impulsos == maiorpulsos)
                    quantos++;

                // Solicita e lê a quantidade de impulsos por assinante
                Console.Write("\n\t\tInforme a quantidade de impulsos, -1 para \"Sair\": ");
                impulsos = Int32.Parse(Console.ReadLine());
            }
        }
    }
}

```

```

// Exibe uma linha na tela
Console.WriteLine("\t\t_____");

        }while(impulsos != -1);

        // Exibe o resultado
        MessageBox.Show("Maior número de impulsos no mês: " + maiorpulsos,
"Verificando a quantidade de pulsos mensal",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

        MessageBox.Show("Número de Assinantes: " + quantos, "Verificando o número
de assinantes",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
        Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do Método Main

} // Fim da Classe Telefone
}

```

15.06 – Gera uma série Fibonacci

Estrutura de repetição do/while. Gera e imprime a série de Fibonacci.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Fibonacci
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Números
    {
        /// <summary>
        /// The main entry point for the application.

```

```

/// </summary>
[STAThread]
static void Main(string[] args)
{
    // Exibe a data e hora!
    Console.WriteLine("\n\t\tData: " + DateTime.Now);

    // Exibe uma linha na tela
    MessageBox.Show("Informe a quantidade de termos da série!", "Gera e imprime
a série de Fibonacci.",
        MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

    Console.WriteLine("\t\t_____ \n");

    // Declaração e Inicialização de Variáveis
    int ANT1 = 1, ANT2 = 1, ATUAL, N, CONT = 1;
    string MENSAGEM = "Ant1\t\tAnt2\t\tAtual\n\n";
    string RESPOSTA = "Sim";

    // Estrutura de repetição do/while
    do
    {
        Console.WriteLine("\n\t\tInforme a quantidade de termos da série de
Fibonacci: ");

        N = Int32.Parse(Console.ReadLine());

        do
        {
            ATUAL = ANT1 + ANT2;
            MENSAGEM += String.Format("{0}",ANT1) + "\t\t" +
                String.Format("{0}",ANT2) + "\t\t" +
                String.Format("{0}",ATUAL) + "\n";
            ANT1 = ANT2;
            ANT2 = ATUAL;
            CONT++;

        }while(CONT <= N);

        // Exibe o resultado
        MessageBox.Show(MENSAGEM, "Verificando a série de Fibonacci",
            MessageBoxButtons.OK, MessageBoxIcon.Information);

        // Reinicialização de variáveis
        ANT1 = 1;
        ANT2 = 1;
        CONT = 1;
        MENSAGEM = "Ant1\t\tAnt2\t\tAtual\n\n";

        // Verifica se o usuário deseja continuar a gerar novas séries...
        Console.WriteLine("\n\t\tDeseja continuar?, digite \"Sim\" para prosseguir: ");
        RESPOSTA = Console.ReadLine();

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");
    }
}

```

```

        }while(RESPOSTA == "SIM" || RESPOSTA == "Sim" || RESPOSTA == "sim" ||
RESPOSTA == "S" || RESPOSTA == "s");

        // Exibe uma linha na tela

Console.WriteLine("\t\t_____");

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
        Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do Método Main

} // Fim da classe Números
}

```

15.07 - Fatorial de um número

Estrutura de repetição do/while... Calcula o fatorial de um número qualquer.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Fatorial
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Número
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Informe um número!", "Calcula o fatorial de um número...",

```

```

        MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

Console.WriteLine("\t\t_____ \n");

        // Declaração e Inicialização de Variáveis
        int n, // Número a ser fornecido pelo usuário
            cont = 1, // Contador
            fat = 1; // Fatorial

        string resposta = "sim";

        do
        {

            // Solicita e lê um número de um usuário
            Console.Write("\t\tInforme um número: ");
            n = Int32.Parse(Console.ReadLine());

            // Exibe uma linha na tela

Console.WriteLine("\t\t_____");

            // Estrutura de repetição do/while
            do
            {
                if (n == 0) // Não existe fatorial de número negativo!
                    fat = 1;
                else
                    fat *= cont;
                cont++;
            }while(cont <= n);

            // Exibe o resultado
            Console.WriteLine("\t\t\t\t\tO fatorial de {0} é = {1}", n, fat);

            // Reinicialização de variáveis
            cont = 1;
            fat = 1;

            // Exibe uma linha na tela
            Console.WriteLine("\t\t\t\t\t_____");

            // Verifica se o usuário deseja continuar pesquisando o fatorial de um
número qualquer

            Console.Write("\t\tDeseja continuar?, digite \"Sim\" para prosseguir: ");
            resposta = Console.ReadLine();

            // Exibe uma linha na tela

Console.WriteLine("\t\t_____");

        }while(resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==
"S" || resposta == "s");

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",

```

```

        MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
        Console.WriteLine("\t\t\t\t\t http://www.gupnet.com.br");

        Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

        } // Fim do Método Main
    } // Fim da Classe Número
}

```

15.08 – Equação do 2º grau

Estrutura de repetição do/while aninhada em do/while. Calcula a seguinte equação do segundo grau:

* para $X = 1, 2, 3, 4, \dots 10$

* $F(X) = (X^2 - 3X - 4)$

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Função
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Matemática
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Verifique o código do programa!", "Calcula uma equação...",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            int X = 1;

```



```

namespace Conversão
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Temperatura
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Informe a temperatura!", "Converte temperaturas",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            int contador = 1;

            double celsius = 0, // Temperatura na escala Celsius
                fahrenheit; // Temperatura na escala Fahrenheit

            string mensagem = "Fahrenheit\tCelsius\n";

            string resposta = "sim";

            // Estrutura de repetição do / while
            do
            {
                // Solicita e lê a temperatura na escala Fahrenheit
                Console.Write("\n\t\tInforme a temperatura em graus Fahrenheit: ");
                fahrenheit = Double.Parse(Console.ReadLine());

                // Exibe uma linha na tela

                Console.WriteLine("\t\t_____ \n");

                // Estrutura de repetição do while
                do
                {
                    celsius = 5 * (fahrenheit - 32)/9;
                    fahrenheit++;
                    mensagem += "    " + fahrenheit + " °F" + "\t\t" +
                        String.Format( "{0:n}", celsius) + " °C\n";

                    MessageBox.Show(mensagem, "Convertendo temperaturas",
                        MessageBoxButtons.OK, MessageBoxIcon.Information);

                    contador++;
                } while (resposta == "sim");
            } while (resposta == "sim");
        }
    }
}

```



```

{
    /// <summary>
    /// The main entry point for the application.
    /// </summary>
    [STAThread]
    static void Main(string[] args)
    {
        // Exibe a data e hora!
        Console.WriteLine("\n\t\tData: " + DateTime.Now);

        // Exibe uma linha na tela
        MessageBox.Show("Verifique o código!", "Calcula os termos de uma
expressão...",
                        MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

        Console.WriteLine("\n\t\t_____ \n");

        // Declaração e Inicialização de Variáveis
        int cont = 1,    // Contador de parcelas
            quant,       // quantidade de termos
            m = 1,       // inverte o sinal da parcela
            num = 2,     // numerador da parcela (fração)
            den = 500,   // denominador da parcela (fração)
            aux = 0;     // auxiliar para alternar o valor do numerador

        double s = 0,   // efetuará o somatório da expressão
            parc;       // parcela (fração)

        string mensagem = " Soma = ";

        // Solicita e lê o número de parcelas da expressão
        Console.Write("\n\t\tInforme a quantidade de termos da expressão: ");
        quant = Int32.Parse( Console.ReadLine() );

        // Estrutura de repetição do while
        do
        {
            parc = (double)(num + aux)*m/den;//2
            s = s + parc;
            aux = aux * (-1) + 5;
            m = m * (-1);
            den = den - 50;
            num = num*(-1) + 2;

            // Exibe o valor de cada parcela
            Console.WriteLine("\n\t\tA {0}º parcela é: {1}",cont, parc);
            cont++;

        }while( cont <= quant);

        // Exibe uma linha na tela

        Console.WriteLine("\n\t\t_____");

        // Exibe o resultado da soma

```

```

        mensagem += String.Format("{0}", s) + "\n";

        MessageBox.Show(mensagem, "Calculando o somatório...",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

        // Exibe o valor de cada parcela
        Console.WriteLine("\n\t\tA soma das {0}º parcelas é: {1}", cont-1, s + "\n");

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
        Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

        Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do Método Main

} // Fim da Classe Somatório
}

```

Exercícios 16 - Estrutura de Seleção Múltipla Switch e outras

16.01 – Lê idade e classifica categoria

Estrutura de seleção múltipla switch. Este programa lê a idade de um nadador e classifica o mesmo em uma das categorias:

- * Infantil A = 5 a 7 anos
- * Infantil B = 8 a 11 anos
- * Juvenil A = 12 a 13 anos
- * Juvenil B = 14 a 17 anos
- * Adultos = Maiores de 18 anos

```
using System;  
using System.Windows.Forms;  
// Adicione uma referência para MessageBox.Show
```

```
namespace Nadador
```

[illegible]

```

Console.WriteLine("Informe a idade do nadador: ");
int idade = Int32.Parse( Console.ReadLine() );

// Verifica a categoria que o nadador se enquadra
switch ( idade )
{
    case 1: // Infantil A = 0 a 4 anos
    case 2:
    case 3:
    case 4:
        Console.WriteLine("\t\t\t\t\tInfantil Mirim");
        // Exibe uma linha na tela

Console.WriteLine("\t\t\t\t\t");
        break;

    case 5: // Infantil A = 5 a 7 anos
    case 6:
    case 7:
        Console.WriteLine("\t\t\t\t\tInfantil A");
        // Exibe uma linha na tela

Console.WriteLine("\t\t\t\t\t");
        break;

    case 8: // Infantil B = 8 a 11 anos
    case 9:
    case 10:
    case 11:
        Console.WriteLine("\t\t\t\t\tInfantil B");
        // Exibe uma linha na tela

Console.WriteLine("\t\t\t\t\t");
        break;

    case 12: // Juvenil A = 12 a 13 anos
    case 13:
        Console.WriteLine("\t\t\t\t\tJuvenil A");
        // Exibe uma linha na tela

Console.WriteLine("\t\t\t\t\t");
        break;

    case 14: // Juvenil B = 14 a 17 anos
    case 15:
    case 16:
    case 17:
        Console.WriteLine("\t\t\t\t\tJuvenil B");
        // Exibe uma linha na tela

Console.WriteLine("\t\t\t\t\t");
        break;

} // Fim do switch

// Verifica Adultos = Maiores de 18 anos

```



```
{
    // Exibe uma linha na tela
    MessageBox.Show("Escolha as opções do menu!", "Calcula Médias!",
        MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

Console.WriteLine("\t\t_____ \n");

    // Declaração e Inicialização de Variáveis

double num1, num2, num3,      // Números a serem fornecidos pelo usuário
    mediah = 0,                // Média Harmônica
    mediag = 0,                // Média Geométrica
    mediaar = 0;               // Média Aritmética

char opc;                      // Opção do Menu

string resposta = "sim"; // Resposta para o usuário

while (resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==
"S" || resposta == "s")
{
    //Apresenta o menu ao usuário
    Console.Write("\n\tQual média deseja calcular? " +
        "\n\t\t1 - Média Harmônica" +
        "\n\t\t2 - Média Geométrica" +
        "\n\t\t3 - Média Aritmética" +
        "\n\t\t\tDigite sua opção: ");

    opc = Char.Parse( Console.ReadLine() );
    // Exibe uma linha na tela

Console.WriteLine("\t\t_____ \n");

    // Estrutura de seleção múltipla switch
    switch ( opc )
    {
        case '1': // Calcula a Média Harmônica

            // Solicita e lê os números para efetuar o cálculo
            Console.Write("\t\tInforme o 1º número: ");
            num1 = Double.Parse( Console.ReadLine() );

            Console.Write("\t\tInforme o 2º número: ");
            num2 = Double.Parse( Console.ReadLine() );

            Console.Write("\t\tInforme o 3º número: ");
            num3 = Double.Parse( Console.ReadLine() );

            mediah = ((num1 * num2 * num3)/((num1*num2)+
(num1*num3)+(num2*num3)));

            // Exibe o resultado da Média
            MessageBox.Show("A média Harmônica é: " + mediah,
                "Calculando a Média...",
                    MessageBoxButtons.OK,
                        MessageBoxIcon.Information);
```

```

        break;

    case '2': // Calcula a Média Geométrica

        // Solicita e lê os números para efetuar o cálculo
        Console.WriteLine("\t\tInforme o 1º número: ");
        num1 = Double.Parse( Console.ReadLine() );

        Console.WriteLine("\t\tInforme o 2º número: ");
        num2 = Double.Parse( Console.ReadLine() );

        Console.WriteLine("\t\tInforme o 3º número: ");
        num3 = Double.Parse( Console.ReadLine() );
        mediag = Math.Pow(num1*num2*num3,3);

        // Exibe o resultado da Média
        MessageBox.Show("A média Geométrica é: " + mediag,
            "Calculando a Média...",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);

        break;

    case '3': // Calcula a Média Aritmética

        // Solicita e lê os números para efetuar o cálculo
        Console.WriteLine("\t\tInforme o 1º número: ");
        num1 = Double.Parse( Console.ReadLine() );

        Console.WriteLine("\t\tInforme o 2º número: ");
        num2 = Double.Parse( Console.ReadLine() );

        Console.WriteLine("\t\tInforme o 3º número: ");
        num3 = Double.Parse( Console.ReadLine() );
        mediaar = (num1 + num2 + num3)/3;

        // Exibe o resultado da Média
        MessageBox.Show("A média Aritmética é: " + mediaar,
            "Calculando a Média...",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);

        break;

    default:
        MessageBox.Show("Opção Inválida!!!", "Verificando sua
opção...",
            MessageBoxButtons.OK,
            MessageBoxIcon.Error);

        break;
}

// Verifica se o usuário deseja prosseguir calculando as médias
Console.WriteLine("\n\t\tDeseja continuar calculando as médias?" +
    "\n\t\tDigite \"Sim\" para prosseguir: ");
resposta = Console.ReadLine();

// Exibe uma linha na tela

```



```

Console.WriteLine("\t\t_____ \n");
    } // Fim do while

    // Exibe uma linha na tela
    MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

    Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

Console.WriteLine("\t\t_____");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

    } // Fim do Método Main

    } // Fim da classe Cálculos
}

```

16.03 – Verifica uma escolha

Estrutura de seleção composta (switch) aninhado em do / while. Verifica qual a linguagem escolhida por um desenvolvedor.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Linguagens
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Desenvolvedor
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Informe sua linguagem de programação!",
                "Aprenda .NET!!!",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\t\t_____ \n");

```

```

// Declaração e Inicialização de Variáveis
string nome, // Nome da linguagem
        resposta = "sim";

// Estrutura de repetição do/while
do
{

    // Solicita e lê a linguagem preferida do programador
    Console.WriteLine("\n\t\tInforme sua linguagem preferida: ");
    nome = Console.ReadLine();

    // Estrutura de seleção composta
    switch( nome)
    {
        case "C#" :
        case "c#" :
            MessageBox.Show("A linguagem é escolhida é: " +
nome, "Ótima escolha!!!",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);
            break;

        case "VB.NET" :
        case "Vb.NET" :
        case "vb.NET" :
        case "VB.Net" :
        case "Vb.Net" :
        case "vb.Net" :
        case "VB.net" :
        case "Vb.net" :
        case "vb.net" :
            MessageBox.Show("A linguagem é escolhida é: " +
nome, "Excelente escolha!!!",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);
            break;

        case "C++.NET" :
        case "c++.NET" :
        case "C++.Net" :
        case "c++.Net" :
        case "C++.net" :
        case "c++.net" :
            MessageBox.Show("A linguagem é escolhida é: " +
nome, "Boa escolha!!!",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);
            break;

        case "DELPHI.NET" :
        case "Delphi.NET" :
        case "delphi.NET" :
        case "DELPHI.Net" :
        case "Delphi.Net" :
        case "delphi.Net" :
    }
}

```

```

        case "DELPHI.net" :
        case "Delphi.net" :
        case "delphi.net" :
            MessageBox.Show("A linguagem é escolhida é: " +
nome, "Legal escolha!!!",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Information);
            break;

        case "JAVA" :
        case "Java" :
        case "java" :
            MessageBox.Show("A linguagem é escolhida é: " +
nome, "Você terá dor de cabeça!!!",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Information);
            break;

        default: // Verifica as outras opções
            MessageBox.Show("A linguagem é escolhida é: " +
nome, "Conheça .NET!!!!",
                                MessageBoxButtons.OK,
                                MessageBoxIcon.Information);
            break;

    } // Fim do switch

    // Verifica se o usuário deseja continuar
    Console.WriteLine("\n\t\tVocê deseja continuar?, digite \"Sim\" para
prosseguir: ");
    resposta = Console.ReadLine();

    // Exibe uma linha na tela
    Console.WriteLine("\n\t\t_____");

    }while(resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==
"S" || resposta == "s");

    // Exibe uma linha na tela
    MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

    Console.WriteLine("\n\t\t Grupo de Usuários do Paraná - Plataforma .NET");
    Console.WriteLine("\n\t\t\t\t\thttp://www.gupnet.com.br");

    Console.WriteLine("\n\t\t_____");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

    } // Fim do Método Main

    } // Fim da Classe Desenvolvedor
}

```

16.04 – Mostra como utilizar estruturas de seleção dupla e composta

Mostrar ao usuário como utilizar estruturas de seleção dupla (if/else), composta (switch) e estruturas de repetição como while, do/while e for.

```
using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show.

namespace Diversos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Testes
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("  Teste de estruturas!", "Testando as estruturas aninhadas!",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\n\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            int opc; // Opção a ser informada pelo usuário

            string mensagem = " ";
            string resposta = "sim";

            // Estrutura de repetição do/while
            do
            {
                // Solicita e lê a opção do usuário
                Console.Write("\n\t\tEscolha uma opção do menu: " +
                    "\n\n\t\t\t1 - Efetuar a soma de números ímpares de 1 a 20" +
                    "\n\t\t\t2 - Calcula a potência de um número elevado ao cubo" +
                    "\n\t\t\t3 - Imprime 5 números em cada linha" +
                    "\n\t\t\t_____ \n" +
                    "\n\t\t\t\tQual sua opção?: ");

                opc = Int32.Parse(Console.ReadLine());

                // Exibe uma linha na tela

                Console.WriteLine("\n\t\t_____ \n");
```

```

// Estrutura de seleção composta (múltipla) switch
switch (opc)
{
    case 1:
        // Declaração e Inicialização de Variáveis
        int sum = 0;

        // Estrutura de repetição for
        for (int cont = 1; cont <= 99; cont += 2)
        {
            sum += cont;
        }

        // Exibe o resultado
        MessageBox.Show("A soma é: " + sum, "Calculando a
soma",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);

        break;

    case 2:
        // Declaração e Inicialização de Variáveis
        double num, // Número a ser informado pelo usuário
        resultado = 0; // Resultado da potência do
número

        // Estrutura de repetição do/while
        do
        {
            // Solicita e lê um número informado pelo usuário
            Console.WriteLine("\n\t\tInforme um número
qualquer: ");

            num = Double.Parse(Console.ReadLine());

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");

            // Calcula a potência do número elevado a 3
            resultado = Math.Pow(num,3);

            mensagem = "A potência do número " + num + "
é: " + String.Format("{0}", resultado);

            // Exibe o resultado
            MessageBox.Show(mensagem, "Calculando a
potência", MessageBoxButtons.OK, MessageBoxIcon.Information);

            // Verifica se o usuário deseja continuar
            Console.WriteLine("\n\t\tDeseja recalcular a
potência?, digite \"Sim\" para prosseguir: ");

            resposta = Console.ReadLine();

            // Exibe uma linha na tela

```

```

Console.WriteLine("\n\t\t_____");

}while(resposta == "SIM" || resposta == "Sim" || resposta
== "sim" || resposta == "S" || resposta == "s");
break;

case 3:
// Estrutura de repetição while.
while (resposta == "SIM" || resposta == "Sim" || resposta
== "sim" || resposta == "S" || resposta == "s")
{
// Declaração e Inicialização de Variáveis
int x;
string saida = " ";

// Solicita e lê o número informado pelo usuário
Console.Write("\n\t\tInforme um número
qualquer: ");

x = Int32.Parse(Console.ReadLine());

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");

// Estrutura de repetição for
for ( ;x <= 20; x++)
{
if ( x % 5 == 0) // Estrutura de seleção
{
saida += x + " " + "\n";
MessageBox.Show(saida,
"Imprime os inteiros de 1 até 20", MessageBoxButtons.OK, MessageBoxIcon.Information);
}
else
{
saida += x + " " + "\t";
MessageBox.Show(saida,
"Imprime os inteiros de 1 até 20", MessageBoxButtons.OK, MessageBoxIcon.Information);
}

} // Fim do for

// Reinicialização de Variáveis
saida = " ";

// Verifica se o usuário deseja continuar
Console.Write("\n\t\tDeseja reimprimir?, digite
'Sim' para prosseguir: ");

resposta = Console.ReadLine();

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");

```

```

        } // Fim do while
        break;

        case 4: // Sair do Programa
            break;

        default: // Mensagem para opção incorreta
            // Exibe uma linha na tela
            MessageBox.Show("Opção Inválida!!!", "Tente outra
vez!",
                            MessageBoxButtons.OK,
MessageBoxIcon.Warning);
            break;
    } // Fim do switch

    // Exibe uma linha na tela

    Console.WriteLine("\n\t\t_____");

    // Verifica se o usuário deseja continuar
    Console.Write("\n\t\tDeseja continuar?, digite \"Sim\" para prosseguir: ");
    resposta = Console.ReadLine();

    // Exibe uma linha na tela

    Console.WriteLine("\n\t\t_____");

    // Fim da estrutura do / while
}while(resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==
"S" || resposta == "s");

    // Exibe uma linha na tela
    MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
        MessageBoxButtons.OK, MessageBoxIcon.Hand);

    Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
    Console.WriteLine("\t\t\t\t\thttp://www.gupnet.com.br");

    Console.WriteLine("\t\t_____");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

    } // Fim do Método Main

    } // Fim da classe Testes
}

```

16.05 – Testando a instrução *break*

Estrutura de repetição do while com for. Testando a instrução break.

```
using System;
```

```

using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace BreakTest
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class BreakTest
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Testando a instrução continue", "Laço de repetição com
continue!",
                            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            string output = "";
            string resposta = "sim";
            int count;

            do // Estrutura de repetição do/while
            {
                for (count = 1; count <= 10; count++)
                {
                    if (count == 5) // Pula o código restante no laço
                        continue; // apenas se count == 5

                    output += count + " ";
                } // fim do laço for

                output += "\nContinua o laço, mas não imprime quando o contador = 5";

                // Exibindo a mensagem
                MessageBox.Show(output, "Demonstrando a instrução break",
                                MessageBoxButtons.OK, MessageBoxIcon.Information);

                // Exibe uma linha na tela
                MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de
estudos!",
                                MessageBoxButtons.OK, MessageBoxIcon.Information);

                Console.WriteLine("\t\t Grupo de Usuários do Paraná -
Plataforma .NET");

                Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");
            }
        }
    }
}

```



```

Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

        // Verifica se o usuário deseja continuar
        Console.Write("\n\t\tDeseja continuar?, \n\t\t\tdigite \"Sim\" para
prosseguir: ");

        resposta = Console.ReadLine();

        // Exibe uma linha na tela

Console.WriteLine("\t\t_____");

        // Reinicialização de Variáveis
        count = 1;
        output = "";

        }while(resposta == "SIM"|| resposta == "Sim"|| resposta == "sim"|| resposta ==
"S"|| resposta == "s");

    } // fim do método Main

} // Fim da classe ContinueTest
}

```

16.06 – Compara números

Utiliza estrutura de seleção simples (if /else), seleção composta (switch) e repetição (while). Compara dois números e verifica qual o maior, menor e se são iguais ou diferentes.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace Números
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Compara
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            MessageBox.Show("Informe os números!", "Compara 2 números quaisquer!",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

```

```

Console.WriteLine("\t\t_____ \n");

// Declaração e Inicialização de Variáveis

double num1, num2;           // Dois números a serem fornecidos pelo usuário

char opc;

string resposta = "sim";

// Verifica se o usuário deseja continuar a comparação entre os dois números
while ( resposta == "sim" || resposta == "SIM" || resposta == "Sim" || resposta ==
"s" || resposta == "S")
{

    // Solicita e lê os números a serem fornecidos pelo usuário
    Console.Write("\t\tInforme o 1º número: ");
    num1 = Double.Parse( Console.ReadLine() );

    Console.Write("\t\tInforme o 2º número: ");
    num2 = Double.Parse( Console.ReadLine() );

    Console.Write("\t\tO que você deseja fazer?" +
        "\t\t\t1 - Verificar o maior número fornecido?" +
        "\t\t\t2 - Verificar o menor número fornecido?" +
        "\t\t\t3 - Verificar se os números são iguais?" +
        "\n\t\t\t\tInforme sua opção: ");

    opc = Char.Parse( Console.ReadLine() );

    // Exibe uma linha na tela

    Console.WriteLine("\t\t_____");

    switch ( opc )
    {
        case '1': // Verifica qual o maior número
            if (num1 > num2)
                Console.WriteLine("\t\t\tO número {0} é o maior
número!", num1);
            else
                Console.WriteLine("\t\t\tO número {0} é o maior
número!", num2);
            break;

        case '2': // Verifica qual o menor número
            if (num1 < num2)
                Console.WriteLine("\t\t\tO número {0} é o
menor número!", num1);
            else
                Console.WriteLine("\t\t\tO número {0} é o
menor número!", num2);
            break;

        case '3': // Verifica se os números são iguais ou diferentes

```

```

        if (num1 == num2)
            Console.WriteLine("\n\t\t\tOs números {0} e {1}
são iguais!!!", num1, num2);
        else
            Console.WriteLine("\n\t\t\tOs números {0} e {1}
são diferentes!!!", num1, num2);
        break;

        default: // Verifica as outras possíveis opções
            Console.WriteLine("\n\t\t\tOpção inválida!!!");
            break;
    }

    Console.WriteLine("\n\t\tVocê deseja continuar?" +
        "\n\t\tDigite \"Sim\" para continuar a pesquisa: ");
    resposta = Console.ReadLine();

    // Exibe uma linha na tela

    Console.WriteLine("\n\t\t_____");

    } // Fim do while

    // Exibe uma linha na tela
    MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

    Console.WriteLine("\n\t\t\thttp://www.gupnet.com.br");

    Console.WriteLine("\n\t\t_____");

    // Exibe uma linha em branco na tela
    Console.WriteLine();

    } // Fim do Método Main

    } // Fim da classe Comparar
}

```

16.07 – Bônus salarial

Estrutura de seleção múltipla switch. Cálculo de bônus salarial de 3 funcionários do seguinte modo:

- * os funcionários do sexo masculino com tempo de casa superior a 15 anos terão direito a um bônus de 20% de seu salário.
- * as funcionárias com tempo de serviço superior a 10 anos terão direito a um bônus de 25% de seu salário.
- * os demais funcionários terão direito a um bônus de R\$ 50,00

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

```

```
namespace Bonus
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Salarial
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);


            // Exibe uma linha na tela
            MessageBox.Show("\tEscolha a categoria!", "Cálcula o bônus salarial de alguns funcionários",
                MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\t\t_____ \n");


            // Declaração e Inicialização de Variáveis
            string mensagem = "Os dados do funcionário(a) são:\n ";
            string resposta = "sim";


            int opc;


            int c_anos; // tempo de casa do funcionário


            double salario; // Salário do funcionário
            salarion = 0; // Novo salário do funcionário
            bonus = 0;      // Bônus do funcionário
            montantep = 0; // Montante parcial
            montante = 0; // Montante total


            // Estrutura de repetição do/while
            do
            {
                // Informa o menu ao usuário
                Console.Write("\t\tCalcula o bônus salarial para: " +
                    "\n\n\t\t\t1 - Adultos Homens" + "\n\t\t\t2 - Adultos Mulheres" +
                    "\n\t\t\t3 - Outros..." + "\n\t\t\t\t\t\t\t\t\t\tDigite sua opção: ");


                opc = Int32.Parse(Console.ReadLine());


                // Exibe uma linha na tela
                Console.WriteLine("\t\t\t\t\t\t\t\t\t\t_____");


                // Estrutura de seleção switch
                switch (opc)
                {
                    case 1:
```

```
// Solicita e lê os dados dos funcionários
Console.WriteLine("Informe os dados do funcionário: ");

Console.WriteLine("\n\t\tTempo de casa: ");
c_anos = Int32.Parse(Console.ReadLine());

Console.WriteLine("\n\t\tSalário: R$ ");
salario = Double.Parse(Console.ReadLine());

// Exibe uma linha na tela
Console.WriteLine("\t_____");

if (c_anos > 15) // Bônus salarial de 20%
{
    bonus = salario * 0.20;
    salarion = salario + bonus;
    mensagem = "\n\tTempo de Serviço: " +
                "\n\tSalário: " + String.Format("{0:c}",
                "\n\tBônus: " + String.Format("{0:c}",
                "\n\tSalário reajustado: " +
                String.Format("{0:c}", salarion);

    MessageBox.Show(mensagem, "Calculando o
                        MessageBoxButtons.OK,
                        MessageBoxIcon.Information);

    // Cálculo do reajuste salarial parcial
    montante += bonus;

    // Exibe o valor total gasto com o bônus para
    Console.WriteLine("\n\t\t\t\tBônus parcial de:
{0:c}", montante);

    Console.WriteLine("\t\t\t\t\t\n");
}
else // Bônus de R$ 50,00
{
    bonus = 50.00;
    salarion = salario + bonus;
    mensagem = "\n\tTempo de Serviço: " +
                "\n\tSalário: " + String.Format("{0:c}",
                "\n\tBônus: " + String.Format("{0:c}",
                "\n\tSalário reajustado: " +
                String.Format("{0:c}", salarion);

    MessageBox.Show(mensagem, "Calculando o
bônus salarial...",
```



```
Console.WriteLine("\t\t\t\t\t");  
    }  
    else // Bônus de R$ 50,00  
    {  
  
        bonus = 50.00;  
        salarion = salario + bonus;  
        mensagem = "\n\tTempo de Serviço: " +  
  
            "\n\tSalário: " + String.Format("{0:c}",  
  
                "\n\tBônus: " + String.Format("{0:c}",  
  
                    "\n\tSalário reajustado: " +  
  
                        MessageBox.Show(mensagem, "Calculando o  
  
                            MessageBoxButtons.OK,  
  
                                // Cálculo do reajuste salarial parcial  
                                montantep += bonus;  
  
                                    // Exibe o valor total gasto com o bônus para  
todos os funcionários  
  
                                        Console.WriteLine("\n\t\t\t\t\tBônus parcial de:  
{0:c}", montantep);  
  
                                            Console.WriteLine("\t\t\t\t\t");  
                                                }  
                                                break;  
  
case 3: // Bônus salarial de R$ 50,00  
  
        // Solicita e lê os dados dos funcionários  
        Console.Write("\n\tInforme os dados do funcionário(a):  
");  
  
        Console.Write("\n\t\tTempo de casa: ");  
        c_anos = Int32.Parse(Console.ReadLine());  
  
        Console.Write("\n\t\tSalário: R$ ");  
        salario = Double.Parse(Console.ReadLine());  
  
        // Exibe uma linha na tela  
  
        Console.WriteLine("\t\t\t\t\t");  
        if (c_anos <= 50)  
        {  
  
            bonus = 50.00;  
            salarion = salario + bonus;  
            mensagem = "\n\tTempo de Serviço: " +  
  
                "\n\tSalário: " + String.Format("{0:c}",
```

```

bonus) +
String.Format("{0:c}", salarion);

bônus salarial...",
MessageBoxIcon.Information);

// Cálculo do reajuste salarial parcial
montantep += bonus;

// Exibe o valor total gasto com o bônus para
todos os funcionários
Console.WriteLine("\n\t\t\t\t\tBônus parcial de:
{0:c}", montantep);

Console.WriteLine("\t\t\t\t\t_____ \n");
}
break;

default: // Considera todas as outras opções
MessageBox.Show("Opção incorreta!", "Verificando sua
opção....",
MessageBoxButtons.OK, MessageBoxIcon.Error);
break;

} // Fim do switch

// Verifica se o usuário deseja continuar
Console.Write("\n\t\tDeseja continuar?, \"Sim\" para prosseguir: ");
resposta = Console.ReadLine();

// Exibe uma linha na tela
Console.WriteLine("\t\t_____");

// Cálculo do reajuste salarial global
montante += montantep;

// Reinicialização do valor do montante parcial
montantep = 0;

}while(resposta == "Sim" || resposta == "sim" || resposta == "SIM" || resposta ==
"S" || resposta == "s");

// Exibe o valor total gasto com o bônus para todos os funcionários
Console.WriteLine("\n\t\t\t\t\tBônus global de: {0:c}", montante);

Console.WriteLine("\t\t_____");

// Exibe uma linha na tela
MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!");

```



```

        MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t Grupo de Usuários do Paraná - Plataforma .NET");
        Console.WriteLine("\t\t\t\t http://www.gupnet.com.br");

        Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do Método Main

} // Fim da Classe Salarial
}

```

16.08 – Testando instrução *break*

Estrutura de repetição do while com for. Testando a instrução break.

```

using System;
using System.Windows.Forms;
// Adicione uma referência para MessageBox.Show

namespace BreakTest
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class BreakTest
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {

            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

            // Exibe uma linha na tela
            MessageBox.Show("Testando a instrução break", "Laço de repetição com
break!",
                            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

            Console.WriteLine("\t\t_____ \n");

            // Declaração e Inicialização de Variáveis
            string output = "";
            string resposta = "sim";
            int count;

            do        // Estrutura de repetição do/while
            {

```

```

        for (count = 1; count <= 10; count++)
        {
            if (count == 5) // Pula o código restante no laço
                break;      // se count == 5

            output += count + " ";
        } // fim do laço for

        output += "\nSai do laço de repetição quando contador = " + count;

        // Exibindo a mensagem
        MessageBox.Show(output, "Demonstrando a instrução break",
            MessageBoxButtons.OK, MessageBoxIcon.Information);

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de
estudos!",
            MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t Grupo de Usuários do Paraná -
Plataforma .NET");
        Console.WriteLine("\t\t\t\t\t http://www.gupnet.com.br");

        Console.WriteLine("\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

        // Verifica se o usuário deseja continuar
        Console.WriteLine("\n\t\tDeseja continuar?, \n\t\t\t\t\t digite \"Sim\" para
prosseguir: ");

        resposta = Console.ReadLine();

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");

        // Reinicialização de Variáveis
        count = 1;
        output = "";

    } while (resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta ==
"S" || resposta == "s");

    } // fim do método Main

    } // Fim da classe BreakTest
}

```

16.09 – Cálculo de áreas de figuras geométricas

Estrutura de seleção composta switch. Calcula as áreas de algumas figuras geométricas

```
using System;
using System.Windows.Forms;

// Adicione uma referência para MessageBox.Show

namespace Calcula
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Areas
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe uma linha na tela
            MessageBox.Show("Pressione \"OK\" para prosseguir...", "Calcula área de algumas figuras geométricas",
                MessageBoxButtons.OK, MessageBoxIcon.Question);

            Console.WriteLine("_____\n");

            // Declaração e inicialização de variáveis
            double area = 0;

            string resposta = "sim";

            // Verifica se o usuário deseja calcular mais uma vez
            while (resposta == "SIM" || resposta == "Sim" || resposta == "sim" || resposta == "S" || resposta == "s")
            {
                // Solicita e apresenta um menu de escolha ao usuário
                Console.WriteLine("Escolha a sua opção abaixo: " +
                    "\n\t\t1 - Calcular área do quadrado" +
                    "\n\t\t2 - Calcular área do retângulo" +
                    "\n\t\t3 - Calcular área da círculo" +
                    "\n\t\t4 - Calcular área do triângulo equilátero" +
                    "\n\t\t5 - Calcular área do cubo" +
                    "\n\t\t6 - Calcular área do cilindro" +
                    "\n\t\t7 - Calcular área do paralelogramo" +
                    "\n\t\t8 - Calcular área do triângulo qualquer" +
                    "\n\t\t9 - Calcular área do losango" +
                    "\n\t\t10 - Calcular área do trapézio");

                // Exibe uma linha na tela

                Console.WriteLine("_____\n");

                Console.WriteLine("Informe sua opção: ");

                // Lê a opção que o usuário deseja calcular
                int opc = Int32.Parse( Console.ReadLine());
            }
        }
    }
}
```

```

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");

// Estrutura de seleção composta switch
switch ( opc )
{
    case 1: // Calcula a área do quadrado
        Console.WriteLine("\n\t\tInforme a medida do lado do
quadrado: ");

        double lado1 = Double.Parse( Console.ReadLine() );
        area = Math.Pow(lado1,2);
        // Exibe o resultado
        Console.WriteLine("\n\t\tA área do quadrado é: {0:n}",
area);

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");
        break;

    case 2: // Calcula a área do retângulo
        Console.WriteLine("\n\t\tInforme a medida da base do
retângulo: ");

        double base1 = Double.Parse( Console.ReadLine() );
        Console.WriteLine("\n\t\tInforme a medida da altura do
retângulo: ");

        double altura1 = Double.Parse( Console.ReadLine() );
        area = (base1 * altura1)/2;
        // Exibe o resultado
        Console.WriteLine("\n\t\tA área do retângulo : {0:n}",
area);

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");
        break;

    case 3: // Calcula a área da círculo
        Console.WriteLine("\n\t\tInforme a medida do raio da
circunferência: ");

        double raio1 = Double.Parse( Console.ReadLine() );
        area = (3.1415 * Math.Pow(raio1,2));
        // Exibe o resultado
        Console.WriteLine("\n\t\tA área do retângulo : {0:n}",
area);

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");
        break;

    case 4: // Calcula a área do triângulo equilátero
        Console.WriteLine("\n\t\tInforme a medida do lado do
triângulo equilátero: ");

        double lado2 = Double.Parse( Console.ReadLine() );

```

```

        area = ((Math.Pow(lado2,2))* Math.Sqrt(3.0))/4;
        // Exibe o resultado
        Console.WriteLine("\n\t\tA área do triângulo equilátero:
{0:n}", area);

        // Exibe uma linha na tela

        Console.WriteLine("\t\t_____");
        break;

        case 5: // Calcula a área do cubo
            Console.Write("\n\t\tInforme a medida do lado do cubo:
");

            double lado3 = Double.Parse( Console.ReadLine() );
            area = Math.Pow(lado3,3);
            // Exibe o resultado
            Console.WriteLine("\n\t\tA área do cubo é: {0:n}", area);

            // Exibe uma linha na tela

            Console.WriteLine("\t\t_____");
            break;

            case 6: // Calcula a área do cilindro
                Console.Write("\n\t\tInforme a medida da altura do
cilindro: ");

                double altura2 = Double.Parse( Console.ReadLine() );
                Console.Write("\n\t\tInforme a medida do raio da base do
cilindro: ");

                double raio2 = Double.Parse( Console.ReadLine() );
                area = ((2* (Math.PI)* raio2)* (altura2 + raio2));
                // Exibe o resultado
                Console.WriteLine("\n\t\tA área do cilindro é: {0:n}", area);

                // Exibe uma linha na tela

                Console.WriteLine("\t\t_____");
                break;

                case 7: // Calcula a área do paralelogramo
                    Console.Write("\n\t\tInforme a medida da base do
paralelogramo: ");

                    double base2 = Double.Parse( Console.ReadLine() );
                    Console.Write("\n\t\tInforme a medida da altura do
paralelogramo: ");

                    double altura3 = Double.Parse( Console.ReadLine() );
                    area = (base2 * altura3);
                    // Exibe o resultado
                    Console.WriteLine("\n\t\tA área do paralelogramo é:
{0:n}", area);

                    // Exibe uma linha na tela

                    Console.WriteLine("\t\t_____");
                    break;

                    case 8: // Calcula a área de um triângulo qualquer

```

```

triângulo: ");

Console.WriteLine("\n\t\tInforme a medida da base do

double base3 = Double.Parse( Console.ReadLine() );
Console.WriteLine("\n\t\tInforme a medida da altura do

double altura4 = Double.Parse( Console.ReadLine() );
area = (base3 * altura4)/2;
// Exibe o resultado
Console.WriteLine("\n\t\tA área do triângulo é: {0:n}",

area);

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");
break;

case 9: // Calcula a área do losango
Console.WriteLine("\n\t\tInforme a medida da diagonal maior:

");

double D = Double.Parse( Console.ReadLine() );
Console.WriteLine("\n\t\tInforme a medida da diagonal

menor: ");

double d = Double.Parse( Console.ReadLine() );
area = (D * d)/2;
// Exibe o resultado
Console.WriteLine("\n\t\tA área do losango é: {0:n}",

area);

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");
break;

case 10: // Calcula a área do trapézio
Console.WriteLine("\n\t\tInforme a medida da base maior: ");
double B = Double.Parse( Console.ReadLine() );
Console.WriteLine("\n\t\tInforme a medida da base menor: ");
double b = Double.Parse( Console.ReadLine() );
Console.WriteLine("\n\t\tInforme a medida da altura do

trapézio: ");

double altura5 = Double.Parse( Console.ReadLine() );
area = ((B + b)*altura5)/2;
// Exibe o resultado
Console.WriteLine("\n\t\tA área do trapézio é: {0:n}",

area);

// Exibe uma linha na tela

Console.WriteLine("\t\t_____");
break;

default: // Verifica as outras opções
MessageBox.Show("Opção inválida, tente outra vez!!!",

"Verificando sua opção",

MessageBoxButtons.OK,

MessageBoxIcon.Error);

break;

```

```

        } // Fim do switch

        // Verifica se o usuário efetuará mais um cálculo
        Console.WriteLine("\n\t\tDeseja continuar calculando áreas?" +
            "\n\t\tDigite \"Sim\" para continuar: ");
        resposta = Console.ReadLine();

        // Exibe uma linha na tela

        Console.WriteLine("\n\t\t_____");

        } // Fim do while

        // Exibe uma linha na tela
        MessageBox.Show("http://www.gupnet.com.br", "Visite nosso grupo de estudos!",
            MessageBoxButtons.OK, MessageBoxIcon.Information);

        Console.WriteLine("\t\t\t\thttp://www.gupnet.com.br");

        Console.WriteLine("\n\t\t_____");

        // Exibe uma linha em branco na tela
        Console.WriteLine();

    } // Fim do método Main

} // Fim da classe Areas
}

```

16.10 – Números de votos por candidatos

Estrutura de seleção composta (switch). Verifica o número de votos por candidato em uma eleição, e o número de votos brancos e nulos.

```

using System;
using System.Windows.Forms;

// Adicione uma referência para MessageBox.Show

namespace Votos
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Urnas
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            // Exibe a data e hora!
            Console.WriteLine("\n\t\tData: " + DateTime.Now);

```

```

// Exibe uma linha na tela
MessageBox.Show("Vote certo!", "Iniciando a votação...",
    MessageBoxButtons.OK, MessageBoxIcon.Exclamation);

Console.WriteLine("\t_____");

// Declaração e Inicialização de Variáveis
int nvcand1 = 0,      // número de votos do candidato 1
    nvcand2 = 0,      // número de votos do candidato 2
    nvcand3 = 0,      // número de votos do candidato 3
    nvbrancos = 0,    // número de votos em branco
    nvnullos = 0,     // número de votos nulos
    neleitores = 0,   // número de total de eleitores que compareceram à
urnas

    opc;    // Opção para o eleitor escolher no menu

string resposta = "sim";
string mensagem = "Opções\t\tTotal de votos\n\n";

// Estrutura de repetição do/while
do
{
    // Solicita e lê a opção do usuário
    Console.WriteLine("Informe sua opção: \n\n\t\t1 - MSDN Brasil - http://
www.msdnbrasil.com.br\n" +
        "\t\t2 - TechNet Brasil - http://www.technetbrasil.com.br\n" +
        "\t\t3 - The Spoke .Net - http://br.thespoke.net\n" +
        "\t\t4 - Outro site\n" + "\t\t5 - Não conhece!!!\n" +
        "\n\tA opção é: ");

    opc = Int32.Parse(Console.ReadLine());

    // Estrutura de seleção múltipla
    switch (opc)
    {
        case 1:
            nvcand1++;
            break;

        case 2:
            nvcand2++;
            break;

        case 3:
            nvcand3++;
            break;

        case 4:
            nvbrancos++;
            break;

        case 5:
            nvnullos++;
            break;
    }
}

```