

PONTIFÍCIA UNIVERSIDADE CATÓLICA
SISTEMAS DE INFORMAÇÃO

Arthur Henrique de Paula Abreu
João Gontijo de Oliveira Júnior

RELATÓRIO LAB 1

BELO HORIZONTE
2017

SUMÁRIO

1. Exercício 3.1	2
2. Exercício 3.2	3
3. Exercício 3.3	7
4. Exercício 3.4	9
5. Exercício 3.5	13
6. Exercício 3.6	17
7. Exercício 3.7	20
8. Exercício 3.8	22
9.1. Exercício 5.1	27
9.2. Classe do Exercício 5.1	30
10.1. Exercício 5.2	33
10.2. Classe do Exercício 5.2	35
11.1. Exercício 5.3.....	37
11.2. Classe do Exercício 5.3	39
12.1. Exercício 5.4	40
12.2. Classe do Exercício 5.4	43

1. Exercício 3.1 (Tela)

C:\Users\Arthur\Desktop\Faculdade>prova1 565523 Arthur Henrique de Paula Abreu

NOME1 = Arthur NOME2 = Henrique

A = 8 B = 11 C = 11

x = 8 y = 11 z = 11

x = 18 y = 31 z = 90

A = 8 B = 11 C = 11 r = 0

x = 8 y = 11 z = 11

x = 18 y = 31 z = 90

x = 18 y = 31 z = 90 w = 139

A = 18 B = 31 C = 90 r = 139

1.2. Cálculos

variaveis : i , j , x , r = 0, A = 0 ,B = 0,C = 0, NMAT, NOME1,NOME2,vetor[10];

i , j = 6, x = 565523 , r = 0, A = 0 ,B = 0, C= 0,NMAT = 565523, NOME1 = Arthur, NOME2 = Henrique;

i = (0)(1)(2)(3)(4)(5)(6),x = (565523)(56552)(5655)(565)(56)(5)(0),j = 6, A = 0, B = 0 , C = 0, NMAT = 565523, NOME1 = ARTHUR, NOME2 = HENRIQUE, vetor[1] = 8 ,vetor[2] = 11, vetor[3] = 11;

i = 6, j = 6,r = 0,A = 8 , B = 11 , C = 11, x = 18, y = 31, z = 90,NMAT = 565523, NOME1 = ARTHUR, NOME2 = HENRIQUE;

i = 6, j = 6,r = 139, A = 18, B = 31, C = 90, x = 18 , y = 31, c = 90,NMAT = 565523, NOME1 = ARTHUR, NOME2 = HENRIQUE;

2. Exercício 3.2

```
// Nome do programa: Ex3.2_Valores.cs
// Programador(es): João Gontijo e Arthur Henrique
// Data: 17/08/2017
// Entrada(s): N numeros inteiros através da linha de comando
// Saída(s): Imprime o vetor de números ordenado, o menor, o maior e a média
dos valores.
// Para executar digite: csc.bat Ex3.2_Valores.cs
// Para testar digite: Ex3.2_Valores.exe 9 3 8 5
// Descricao: Recebe n numeros inteiros através da linha de comando,
// armazena em um vetor e chama um método estático que receba estes
// números inteiros (armazenados no vetor), e retorna o maior
// deles e a média de seus valores.
// Faltava: Ordenar o vetor na ordem crescente de valores, obter o menor valor.
```

```
using System;
```

```
using System.Text;
```

```
namespace Ex3_2_Valores
```

```
{
```

```
    class Program
```

```
    {
```

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

```
        {
```

```
            int contador = 0;
```

```
            bool teste = true;
```

```
            int[] teste2 = new int[args.Length];
```

```
            for (int i = 0; i < args.Length; i++) // Repetição
```

```
            {
```

```
                teste = int.TryParse(args[i], out teste2[i]); // Teste de erro
```

```

        if (teste == true)
        {
            contador++;
        }
    }

    if ((contador == args.Length) && args.Length > 0)
    {
        int maior = 0, media = 0, menor = 0, aux = 0; // Inicializacao de variaveis
        para calculo

        int j = args.Length; // Inicializacao de variaveis contadoras

        int [] vetor = new int[j]; // Alocacao de vetor para receber numeros
        inteiros

        Console.WriteLine("\nPrograma Ex3.2.exe com " + j + " parametros :
        ");

        for (int i = 0; i < j ; i++) // Mostra e converte parametros recebidos na
        LC
        {
            Console.Write(" args[" + i + "] = " + args[i]);
            Console.Write(" args[{0}] = {1}", i, args[i]);
            vetor[i] = int.Parse(args[i]); // Converte parametro recebidos na LC de
            string para inteiro e armazena no vetor

            Console.WriteLine(" vetor[{0}] = {1}", i, vetor[i]); // Mostra vetor
        }

        Operar_valores(vetor, ref maior, ref media, ref menor, ref aux); //
        Chama metodo para fazer as operacoes solicitadas

        Console.WriteLine("Maior : " + maior); // Mostra maior valor

```

```

        Console.WriteLine("Media : " + media); // Mostra media dos
valores
        Console.WriteLine("Menor : " + menor); // Mostra menor valor
        Console.Write("Vetor : ");

        for (int i = 0; i < vetor.Length; i++) // Mostrar valores ordenados
        {
            Console.Write(vetor[i] + " ");
        }
    }
    else
    {
        Console.WriteLine("\n ERRO DE PARÂMETRO\n");
        Console.ReadKey();
    }
}

static void Operar_valores(int[] vetor, ref int maior, ref int media, ref int
menor, ref int aux) // Método de operação
{

    int soma = 0;

    media = 0;
    maior = vetor[0];
    menor = vetor[0];

    for (int i = 0; i < vetor.Length; i++)
    {
        soma += vetor[i]; // Acumular valores

        if (vetor[i] > maior) // atualizar maior valor
        {
            maior = vetor[i];
        }
    }
}

```

```
        if (vetor[i] < menor) // Atualizar menor Valor
        {
            menor = vetor[i];
        }
    }
```

```
media = soma / vetor.Length; // Calcular media dos valores
```

```
    for (int i = 0; i < vetor.Length; i++) // Ordenar o Vetor
    {
        for (int j = 0; j < vetor.Length; j++)
        {
            if (vetor[i] < vetor[j])
            {
                aux = vetor[i];
                vetor[i] = vetor[j];
                vetor[j] = aux;
            }
        }
    }
}
}
```

TELA:

```

C:\Windows\system32\cmd.exe
C:\Users\Arthur\Desktop\Faculdade>C:\Windows\Microsoft.NET\Framework\v4.0.30319\
csc.exe Ex3.2_Valores.cs
Microsoft (R) Visual C# Compiler version 4.7.2053.0
for C# 5
Copyright (C) Microsoft Corporation. All rights reserved.

This compiler is provided as part of the Microsoft (R) .NET Framework, but only
supports language versions up to C# 5, which is no longer the latest version. Fo
r compilers that support newer versions of the C# programming language, see http
://go.microsoft.com/fwlink/?LinkID=533240

C:\Users\Arthur\Desktop\Faculdade>Ex3.2_Valores 5 4 3 2 6 8

Programa Ex3.2.exe com 6 parametros :
args[0] = 5 args[0] = 5 vetor[0] = 5
args[1] = 4 args[1] = 4 vetor[1] = 4
args[2] = 3 args[2] = 3 vetor[2] = 3
args[3] = 2 args[3] = 2 vetor[3] = 2
args[4] = 6 args[4] = 6 vetor[4] = 6
args[5] = 8 args[5] = 8 vetor[5] = 8
Maior : 8
Media : 4
Menor : 2
Vetor : 2 3 4 5 6 8
C:\Users\Arthur\Desktop\Faculdade>

```

3. Exercício 3.3

// Nome do programa: Ex3.3_Data.cs

// Programador(es): João Gontijo e Arthur Henrique

// Data: 18/08/2017

// Entrada(s): Uma data na linha de comando dd/mm/aaaa

// Saída(s): Imprime o número de dias decorridos no ano.

// Para executar digite: csc.bat Ex3.3_Data.cs

// Para testar digite: Ex3.3_Data.exe 15/01/2010

// Descrição: Recebe uma data e verifica quantos dias do ano já se passaram.

```
using System;
```

```
using System.Text;
```

```
namespace Ex3_3_Data
```

```
{
```

```
    class Program
```

```
    {
```

```
        static void quantosDias (int dia, int mes, int ano) // Método para calcular
```

```
a quantidade de dias
```

```
    {
```


DateTime data = new DateTime(ano, mes, dia); // Registra a data digitada pelo usuario

DateTime primeiroDia = new DateTime(ano, 1, 1); // Registra o primeiro dia do ano

TimeSpan diferenca = data - primeiroDia; // Calcula a quantidade de dias/datas/ano de um ano pelo outro

int resultado = diferenca.Days; // Transforma a conta para dias

Console.WriteLine("\n Se passaram {0} dias do ano.", resultado); // Imprime na tela o resultado

}

static void Main(string[] args)

{

if (args.Length != 1)

{

Console.WriteLine("\n ERRO DE PARÂMETRO\n"); //

Verificação de erro

Console.ReadKey();

}

else

{

char delimitador = '/';

string[] data = args[0].Split(delimitador); // Divide a data

int dia = int.Parse(data[0]); // Variáveis

int mes = int.Parse(data[1]);

int ano = int.Parse(data[2]);

if ((ano % 4 == 0) && (dia > 0 && dia <= 31) && (mes > 0 && mes <= 12) && ano > 0) // Condições / Validações

{

quantosDias(dia, mes, ano);

// Descricao: Recebe o número de eleitores e contabiliza os votos.

```
using System;
```

```
using System.Text;
```

```
namespace Ex3_4_Votacao
```

```
{
```

```
    class Program
```

```
    {
```

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

```
        {
```

```
            if (args.Length != 1)
```

```
            {
```

```
                Console.WriteLine("\n ERRO DE PARÂMETRO\n"); //
```

Verificação de erro

```
                Console.ReadKey();
```

```
            }
```

```
            else
```

```
            {
```

```
                // Variaveis
```

```
                string teste2;
```

```
                int numEleitores;
```

```
                int voto = 0;
```

```
                int machado = 0, guimaraes = 0, cecilia = 0;
```

```
                int i = 0;
```

```
                bool teste = int.TryParse(args[0], out numEleitores); //
```

Primeiro teste de erro

```
                if (teste == true)
```

```
                {
```

```
                    numEleitores = int.Parse(args[0]); // Teste de erro
```

```
                }
```

```
            else
```

```

    {
        numEleitores = 0; // Teste de erro
    }

    if (numEleitores > 0)
    {
        while (i < numEleitores)
        {
            Console.WriteLine("\n Digite o voto do {0}º eleitor (19,21,33): ", i
+ 1); // Solicita os votos
            teste2 = Console.ReadLine();

            bool teste3 = int.TryParse(teste2, out voto); //

```

Segundo teste de erro

```

            if (teste3 == true)
            {
                voto = int.Parse(teste2); // Segundo teste de
erro
            }
            else
            {
                voto = 0;
            }

            // Registro de votos
            if (voto == 19)
            {
                machado++;
                i++;
            }
            else if (voto == 21)
            {

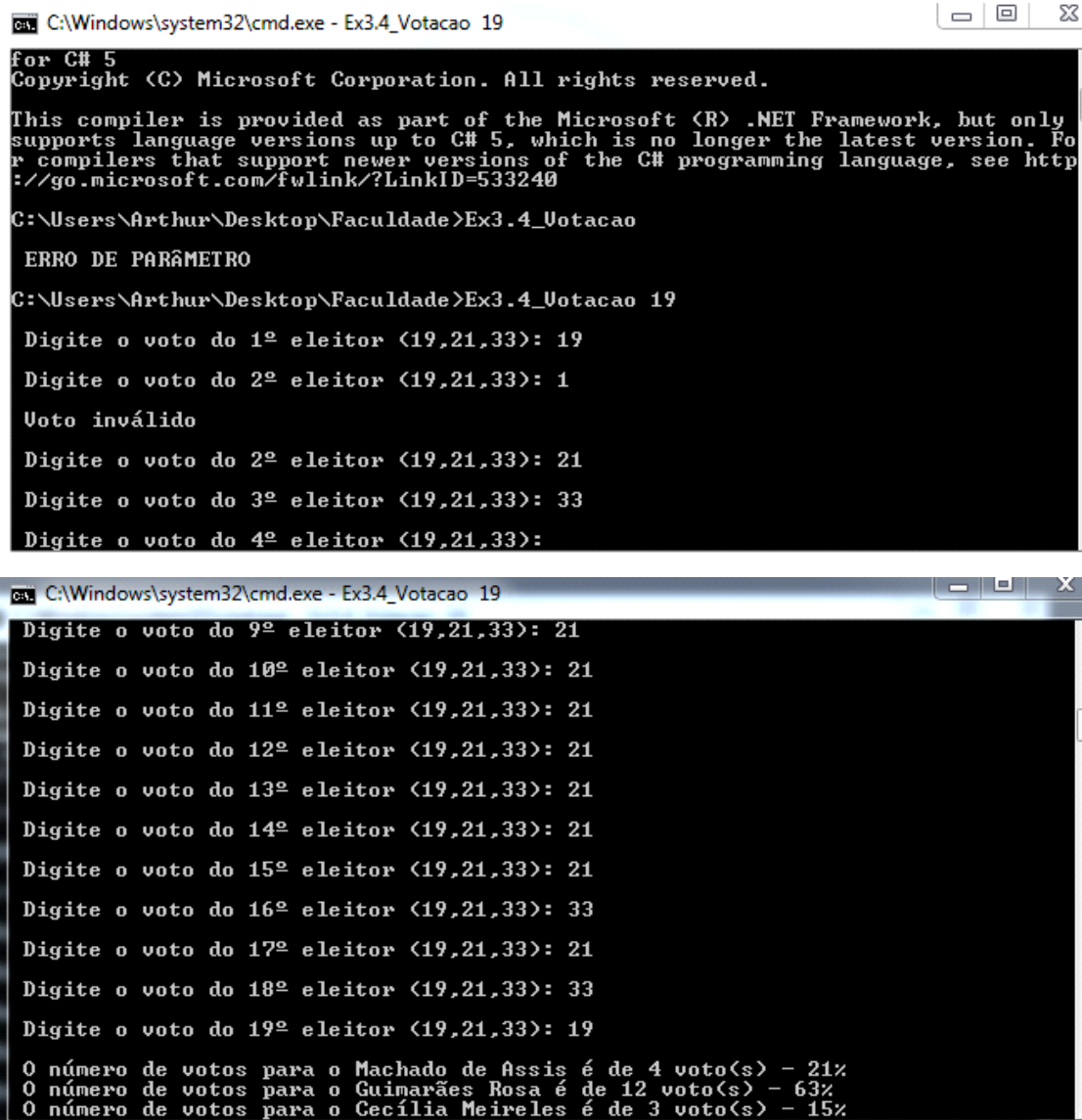
```

```

        guimaraes++;
        i++;
    }
    else if (voto == 33)
    {
        cecilia++;
        i++;
    }
    else
    {
        Console.WriteLine("\n Voto inválido\n"); // Verificação de
erro
    }
}
// Imprime os resultados
Console.WriteLine("\n O número de votos para o Machado de
Assis é de {0} voto(s) - {1}%", machado, (machado * 100) / numEleitores);
Console.WriteLine("\n O número de votos para o Guimarães
Rosa é de {0} voto(s) - {1}%", guimaraes, (guimaraes * 100) / numEleitores);
Console.WriteLine("\n O número de votos para o Cecília Meireles
é de {0} voto(s) - {1}%", cecilia, (cecilia * 100) / numEleitores);

Console.ReadKey();
}
else
{
    Console.WriteLine("\n ERRO DE PARÂMETRO\n"); //
Verificação de erro
    Console.ReadKey();
}
}
}
}
}
}
}

```



```

C:\Windows\system32\cmd.exe - Ex3.4_Votacao 19
for C# 5
Copyright (C) Microsoft Corporation. All rights reserved.

This compiler is provided as part of the Microsoft (R) .NET Framework, but only
supports language versions up to C# 5, which is no longer the latest version. Fo
r compilers that support newer versions of the C# programming language, see http
://go.microsoft.com/fwlink/?LinkID=533240

C:\Users\Arthur\Desktop\Faculdade>Ex3.4_Votacao

ERRO DE PARÂMETRO

C:\Users\Arthur\Desktop\Faculdade>Ex3.4_Votacao 19

  Digite o voto do 1º eleitor <19,21,33>: 19
  Digite o voto do 2º eleitor <19,21,33>: 1
  Voto inválido
  Digite o voto do 2º eleitor <19,21,33>: 21
  Digite o voto do 3º eleitor <19,21,33>: 33
  Digite o voto do 4º eleitor <19,21,33>:

C:\Windows\system32\cmd.exe - Ex3.4_Votacao 19

  Digite o voto do 9º eleitor <19,21,33>: 21
  Digite o voto do 10º eleitor <19,21,33>: 21
  Digite o voto do 11º eleitor <19,21,33>: 21
  Digite o voto do 12º eleitor <19,21,33>: 21
  Digite o voto do 13º eleitor <19,21,33>: 21
  Digite o voto do 14º eleitor <19,21,33>: 21
  Digite o voto do 15º eleitor <19,21,33>: 21
  Digite o voto do 16º eleitor <19,21,33>: 33
  Digite o voto do 17º eleitor <19,21,33>: 21
  Digite o voto do 18º eleitor <19,21,33>: 33
  Digite o voto do 19º eleitor <19,21,33>: 19

O número de votos para o Machado de Assis é de 4 voto(s) - 21%
O número de votos para o Guimarães Rosa é de 12 voto(s) - 63%
O número de votos para o Cecília Meireles é de 3 voto(s) - 15%

```

5. Exercício 3.5

// Nome do programa: Ex3.5_Calculadora.cs
 // Programador(es): João Gontijo e Arthur Henrique
 // Data: 18/08/2017
 // Entrada(s): Um cálculo na linha de comando
 // Saída(s): Imprime o resultado do cálculo
 // Para executar digite: csc.bat Ex3.5_Calculadora.cs
 // Para testar digite: Ex3.5_Calculadora.exe 2.5 x 3
 // Descrição: Recebe um cálculo e o executa

using System;

```

using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ex3_5_Calculadora
{
    class Program
    {
        static void adicao (double num1, double num2) // Módulo de adição
        {
            Console.WriteLine("\n {0} + {1} = {2}\n", num1, num2, num1 + num2);
            Console.ReadKey();
        }

        static void subtracao (double num1, double num2) // Módulo de
subtração
        {
            Console.WriteLine("\n {0} - {1} = {2}\n", num1, num2, num1 - num2);
            Console.ReadKey();
        }

        static void multiplicacao (double num1, double num2) // Módulo de
multiplicacao
        {
            Console.WriteLine("\n {0} x {1} = {2}\n", num1, num2, num1 * num2);
            Console.ReadKey();
        }

        static void divisao (double num1, double num2) // Módulo de divisão
        {
            Console.WriteLine("\n {0} / {1} = {2}\n", num1, num2, num1 / num2);
            Console.ReadKey();
        }

        static void Main(string[] args)
        {
            if (args.Length != 3)

```

```

        {
            Console.WriteLine("\n QUANTIDADE DE PARÂMETROS
INVÁLIDA! \n"); // Verificação de erro
            Console.ReadKey();
        }
        else
        {
            double num1;
            bool teste1 = double.TryParse(args[0], out num1); //
Variável de erro

            char operacao;
            bool teste2 = char.TryParse(args[1], out operacao); //
Variável de erro

            double num2;
            bool teste3 = double.TryParse(args[2], out num2); //
Variável de erro

            if (teste1 == true && teste2 == true && teste3 == true) //
Teste de erro

            {
                if (operacao == '+') // Condições
                {
                    adicao(num1, num2); // Chama método de adição
                }
                else if (operacao == '-') // Condições
                {
                    subtracao(num1, num2); // Chama método de subtração
                }
                else if (operacao == 'x' || operacao == 'X' || operacao == '*') //
Condições

                {

```



```

C:\Windows\system32\cmd.exe - Ex3.5_Calculadora 10 - 5
This compiler is provided as part of the Microsoft (R) .NET Framework, but only
supports language versions up to C# 5, which is no longer the latest version. Fo
r compilers that support newer versions of the C# programming language, see http
://go.microsoft.com/fwlink/?LinkID=533240
C:\Users\Arthur\Desktop\Faculdade>Ex3.5_Calculadora 5x5
QUANTIDADE DE PARÂMETROS INVÁLIDA!
C:\Users\Arthur\Desktop\Faculdade>Ex3.5_Calculadora 5 x 5
5 x 5 = 25
C:\Users\Arthur\Desktop\Faculdade>Ex3.5_Calculadora 4 + 4
4 + 4 = 8
C:\Users\Arthur\Desktop\Faculdade>Ex3.5_Calculadora 4 / 2
4 / 2 = 2
C:\Users\Arthur\Desktop\Faculdade>Ex3.5_Calculadora 10 - 5
10 - 5 = 5

```

6. Exercício 3.6

// Nome do programa: Ex3.6_Fatorial
 // Programador(es): João Gontijo e Arthur Henrique
 // Data: 19/08/2017
 // Entrada(s): Um número para fatorar
 // Saída(s): Um número fatorado
 // Para executar digite: csc.bat Ex3.6_Fatorial.cs
 // Para testar digite: Ex3.6_Fatorial.exe 4
 // Descrição: Recebe um número e o fatora

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ex3_6_Fatorial
{
    class Program
    {
        static void fatorial (int num1) // Método de fatorial

```

```

    {
        int mostrar = num1;

        for (int i = num1 - 1; i > 1; i--) // Fatoração
        {
            num1 *= i;
        }

        if (num1 == 0)
        {
            Console.WriteLine("\n {0}! = {1}\n", mostrar, 1); // Imprime
resultado
            Console.ReadKey();
        }
        else
        {
            Console.WriteLine("\n {0}! = {1}\n", mostrar, num1); // Imprime
resultado
            Console.ReadKey();
        }
    }

    static void Main(string[] args)
    {
        if (args.Length != 1)
        {
            Console.WriteLine("\n PARÂMETRO INVÁLIDO!\n"); // Erro de
parâmetro
            Console.ReadKey();
        }
        else
        {
            int num1;
            bool teste = int.TryParse(args[0], out num1); // Verificação
de erro

```



```
// Nome do programa: Ex3.7_Conversor.cs
// Programador(es): João Gontijo e Arthur Henrique
// Data: 21/08/2017
// Entrada(s): Uma temperatura em Celsius ou Fahrenheit
// Saída(s): A temperatura convertida para Celsius ou Fahrenheit
// Para executar digite: csc.bat Ex3.7_Conversor.cs
// Para testar digite: Ex3.7_Conversor.exe 10 c
// Descrição: Conversor de temperaturas, Celsius e Fahrenheit
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ex3_7_Conversor
{
    class Program
    {
        static void conversor (string num1, string char1) // Método de conversão
        {
            double temp = double.Parse(num1); // Variáveis
            char tipo = char.Parse(char1); // Variáveis

            if ((tipo == 'f' || tipo == 'F') && temp > -459.76)
            {
                Console.WriteLine("\n {0} graus Fahrenheit = {1} graus Celsius\n", temp, (temp - 32) / 1.8); // Conversão de F para C
                Console.ReadKey();
            }
            else if ((tipo == 'c' || tipo == 'C') && temp > -273.15)
            {

```

```

        Console.WriteLine("\n {0} graus Celsius = {1} graus
Fahrenheit\n", temp, (temp * 1.8) + 32); // Conversão de C para F
        Console.ReadKey();
    }
    else
    {
        Console.WriteLine("\n TEMPERATURA INVÁLIDA\n"); //
Mensagem de erro
        Console.ReadKey();
    }
}
static void Main(string[] args)
{
    if (args.Length != 2)
    {
        Console.WriteLine("\n NÚMERO DE PARÂMETROS
INVÁLIDO!\n"); // Verificador de entradas
        Console.ReadKey();
    }
    else
    {
        double temp;
        bool teste = double.TryParse(args[0], out temp); // Verificar
se é possível a conversão

        char tipo;
        bool teste2 = char.TryParse(args[1], out tipo); // Verificar se
é possível a conversão

        if (teste == true && teste2 == true)
        {
            conversor(args[0], args[1]); // Chamar o método
        }
    }
    else

```



```

using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ex3_8_Saldo
{
    class Program
    {
        static void depositar (ref double saldo) // Método de deposito
        {
            string auxTeste; // Variáveis
            double auxDeposito;

            Console.WriteLine("\n Digite o valor à depositar: "); // Valor à depositar
            auxTeste = Console.ReadLine();

            bool teste = double.TryParse(auxTeste, out auxDeposito); // Teste
de erro

            if (teste == true && auxDeposito > 0)
            {
                saldo += auxDeposito; // Alterar o saldo
            }
            else
            {
                Console.WriteLine("\n VALOR INVÁLIDO!\n"); // Imformar erro
                Console.ReadKey();
            }
        }
        static void sacar (ref double saldo) // Método de saque

```



```

{
    string auxTeste; // Variáveis
    double auxSaque;

    Console.WriteLine("\n Digite o valor à sacar: "); // Solicitar valor
    auxTeste = Console.ReadLine();

    bool teste = double.TryParse(auxTeste, out auxSaque); // Teste
de erro

    if (teste == true && auxSaque < saldo)
    {
        saldo -= auxSaque; // Alterar saldo
    }
    else
    {
        Console.WriteLine("\n VALOR INVÁLIDO!\n"); // Informar erro
        Console.ReadKey();
    }
}

static void imprimirSaldo (ref double saldo) // Método para informar saldo
{
    Console.WriteLine("\n Seu saldo é de {0} reais\n", saldo); // Informar
saldo
    Console.ReadKey();
}

static void Main(string[] args)
{
    double saldo = 0; // Variáveis
    string suporte;

```

```
int opcao = 0;

while (opcao != 4) // Repetição
{

    Console.WriteLine("\n 1. Depositar\n 2. Sacar\n 3. Imprimir
saldo\n 4. Sair\n Digite sua opção: "); // Menu
    suporte = Console.ReadLine();

    bool teste = int.TryParse(suporte, out opcao); // Teste de erro

    if (teste == true)
    {
        switch(opcao)
        {
            case 1:

                depositar(ref saldo); // Chama método de
depósito

                break;

            case 2:

                sacar(ref saldo); // Chama método de saque

                break;

            case 3:
```

```

        imprimirSaldo(ref saldo); // Chama método de
imprimir na tela

        break;

        case 4: // Usado para evitar mensagem de
erro

        break;

        default:

        Console.WriteLine("\n ERRO OPÇÃO
INVÁLIDA!\n"); // Informa erro
        Console.ReadKey();

        break;
    }
}

else
{
    Console.WriteLine("\n ERRO OPÇÃO INVÁLIDA!\n"); //
Informa erro
    Console.ReadKey();

}

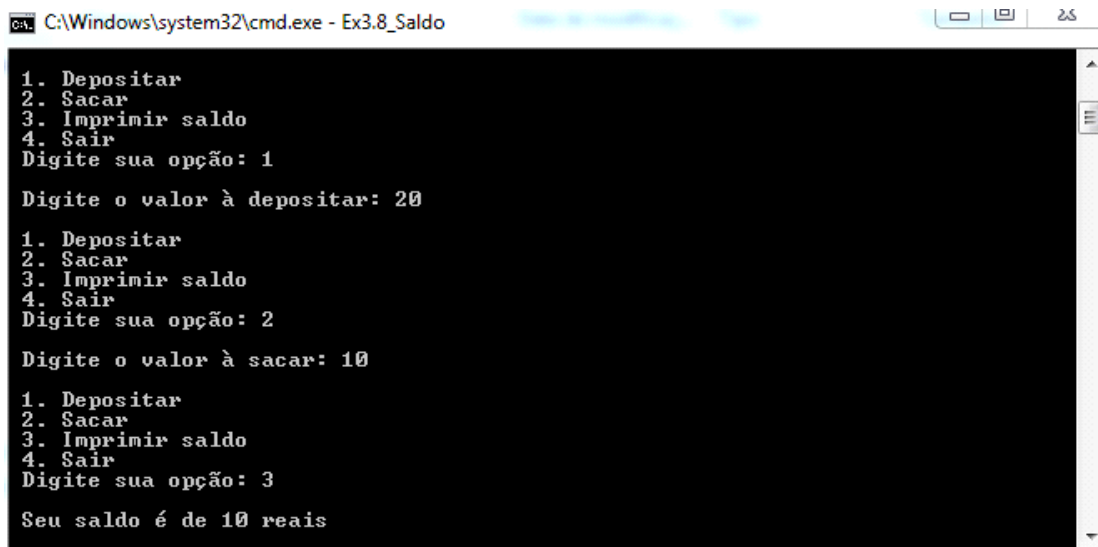
}

}

}
}
}
}

```

TELA:



```

C:\Windows\system32\cmd.exe - Ex3.8_Saldo
1. Depositar
2. Sacar
3. Imprimir saldo
4. Sair
Digite sua opção: 1

Digite o valor à depositar: 20

1. Depositar
2. Sacar
3. Imprimir saldo
4. Sair
Digite sua opção: 2

Digite o valor à sacar: 10

1. Depositar
2. Sacar
3. Imprimir saldo
4. Sair
Digite sua opção: 3

Seu saldo é de 10 reais
  
```

9.1. Exercício 5.1

// Nome do programa: Ex5.1_Saldo.cs

// Programador(es): João Gontijo e Arthur Henrique

// Data: 24/08/2017

// Entrada(s): Opção no menu, Depósito, Saque

// Saída(s): Mostrar saldo

// Para executar digite: csc.bat Ex5.1_Saldo.cs Ex5.1_ClassSaldo.cs

// Para testar digite: Ex5.1_Saldo.exe

// Descrição: Um programa que registra e altera o saldo de uma pessoa

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Linq;
```

```
using System.Text;
```

```
using System.Threading.Tasks;
```

```
namespace Ex5_1_Saldo
```

```
{
```

```
    class Program
```

```
    {
```

```

static void Main(string[] args)
{
    double saldo = 0; // Variáveis
    string suporte;
    int opcao = 0;

    ClassSaldo conta = new ClassSaldo(); // Criar classe

    while (opcao != 4) // Repetição
    {
        Console.Clear();

        Console.WriteLine("\n 1. Depositar\n 2. Sacar\n 3. Imprimir
saldo\n 4. Sair\n Digite sua opção: "); // Menu
        suporte = Console.ReadLine();

        Console.Clear();

        bool teste = int.TryParse(suporte, out opcao); // Teste de erro

        if (teste == true)
        {
            switch(opcao)
            {
                case 1:

                    saldo = conta.depositar(saldo); // Chama
classe de depósito

                    break;

                case 2:

```

```

    saldo = conta.sacar(saldo); // Chama classe
de saque

    break;

    case 3:

        conta.imprimirSaldo(saldo); // Chama classe
de imprimir na tela

    break;

    case 4: // Usado para evitar mensagem de
erro

    break;

    default:

        Console.WriteLine("\n ERRO OPÇÃO
INVÁLIDA!\n"); // Informa erro

        Console.ReadKey();
        Console.Clear();

        break;
    }
}
else
{
    Console.WriteLine("\n ERRO OPÇÃO INVÁLIDA!\n"); //
Informa erro

    Console.ReadKey();
    Console.Clear();

}
}

```

```

    }
}
}

```

9.2. Classe do Exercício 5.1

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ex5_1_Saldo
{
    class ClassSaldo
    {
        public double depositar (double saldo) // Classe de deposito
        {
            string auxTeste; // Variáveis
            double auxDeposito;

            Console.WriteLine("\n Digite o valor à depositar: "); // Valor à depositar
            auxTeste = Console.ReadLine();

            bool teste = double.TryParse(auxTeste, out auxDeposito); // Teste de
erro

            Console.Clear();

            if (teste == true && auxDeposito > 0)
            {
                saldo += auxDeposito; // Alterar o saldo
                return saldo; // Retorna novo saldo
            }

```

```

else
{
    Console.WriteLine("\n VALOR INVÁLIDO!\n"); // Informar erro
    Console.ReadKey();
    Console.Clear();
    return saldo; // Retorna saldo sem alteração
}
}
public double sacar (double saldo) // Classe de saque
{
    string auxTeste; // Variáveis
    double auxSaque;

    Console.WriteLine("\n Digite o valor à sacar: "); // Solicitar valor
    auxTeste = Console.ReadLine();

    bool teste = double.TryParse(auxTeste, out auxSaque); // Teste de erro

    Console.Clear();

    if (teste == true && auxSaque < saldo)
    {
        saldo -= auxSaque; // Alterar saldo
        return saldo; // Retorna novo saldo
    }
    else
    {
        Console.WriteLine("\n VALOR INVÁLIDO!\n"); // Informar erro
        Console.ReadKey();
        Console.Clear();
        return saldo; // Retorna saldo sem alteração
    }
}
}
public void imprimirSaldo (double saldo) // Classe para informar saldo

```



```

    {
        Console.WriteLine("\n Seu saldo é de {0} reais\n", saldo); // Informar saldo
        Console.ReadKey();
        Console.Clear();
    }
}
}
}

```

TELA:

```

C:\Windows\System32\cmd.exe
C:\Users\joaog\Desktop\Print>csc.bat Ex5.1_Saldo.cs Ex5.1_ClassSaldo.cs
C:\Users\joaog\Desktop\Print>C:\Windows\Microsoft.NET\Framework\v4.0.30319\csc.exe Ex5.1_Saldo.cs Ex5.1_C
lassSaldo.cs
Microsoft (R) Visual C# Compiler version 4.7.2046.0
for C# 5
Copyright (C) Microsoft Corporation. All rights reserved.

This compiler is provided as part of the Microsoft (R) .NET Framework, but only supports language version
s up to C# 5, which is no longer the latest version. For compilers that support newer versions of the C#
programming language, see http://go.microsoft.com/fwlink/?LinkID=533240

C:\Users\joaog\Desktop\Print>Ex5.1_Saldo.exe

1. Depositar
2. Sacar
3. Imprimir saldo
4. Sair
Digite sua opção: 1

Digite o valor à depositar: 20

1. Depositar
2. Sacar
3. Imprimir saldo
4. Sair
Digite sua opção: 2

Digite o valor à sacar: 10

1. Depositar
2. Sacar
3. Imprimir saldo
4. Sair
Digite sua opção: 3

Seu saldo é de 10 reais

1. Depositar
2. Sacar
3. Imprimir saldo
4. Sair
Digite sua opção: 4

C:\Users\joaog\Desktop\Print>

```

10.1. Exercício 5.2

// Nome do programa: Ex5.2_Calculadora.cs

// Programador(es): João Gontijo e Arthur Henrique

// Data: 18/08/2017

// Entrada(s): Um cálculo na linha de comando

// Saída(s): Imprime o resultado do cálculo

// Para executar digite: csc.bat Ex5.2_Calculadora.cs

Ex5.2_ClassCalculadora.cs

// Para testar digite: Ex5.2_Calculadora.exe 2.5 x 3

// Descricao: Recebe um cálculo e o executa

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Linq;
```

```
using System.Text;
```

```
using System.Threading.Tasks;
```

```
namespace Ex5_2_Calculadora
```

```
{
```

```
    class Program
```

```
    {
```

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

```
        {
```

```
            if (args.Length != 3)
```

```
            {
```

```
                Console.WriteLine("\n QUANTIDADE DE PARÂMETROS
```

```
INVÁLIDA! \n"); // Verificação de erro
```

```
                Console.ReadKey();
```

```
            }
```

```
            else
```

```
            {
```

```
                Console.Clear();
```

```
                double num1;
```

```
                bool teste1 = double.TryParse(args[0], out num1); //
```

Variável de erro

```
                char operacao;
```

```
                bool teste2 = char.TryParse(args[1], out operacao); //
```

Variável de erro

```

double num2;
    bool teste3 = double.TryParse(args[2], out num2); //
Variável de erro

    operadores oper = new operadores();

    if (teste1 == true && teste2 == true && teste3 == true) //
Teste de erro
    {
        if (operacao == '+') // Condições
        {
            oper.adicao(num1, num2); // Chama classe de adição
        }
        else if (operacao == '-') // Condições
        {
            oper.subtracao(num1, num2); // Chama classe de
subtração
        }
        else if (operacao == 'x' || operacao == 'X' || operacao == '*') //
Condições
        {
            oper.multiplicacao(num1, num2); // Chama classe de
multiplicação
        }
        else if (operacao == '/' && (num2 != 0)) // Condições
        {
            oper.divisao(num1, num2); // Chama classe de divisão
        }
        else
        {
            Console.WriteLine("\n Cálculo inválido\n"); // Imprime resposta
ao erro

            Console.ReadKey();

```

```

        }
    }
    else
    {
        Console.WriteLine("\n ERRO DE PARÂMETRO! \n"); //
Verificação de erro
        Console.ReadKey();
    }
}
}
}
}
}
}
}
}

```

10.2. Classe do Exercício 5.2

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ex5_2_Calculadora
{
    class operadores
    {
        public void adicao (double num1, double num2) // Classe de adição
        {
            Console.WriteLine("\n {0} + {1} = {2}\n", num1, num2, num1 + num2); //
Faz a conta
            Console.ReadKey();
        }
        public void subtracao (double num1, double num2) // Classe de
subtração
        {

```

```
Console.WriteLine("\n {0} - {1} = {2}\n", num1, num2, num1 - num2); //
```

Faz a conta

```
Console.ReadKey();
```

```
}
```

public void multiplicacao (double num1, double num2) // Classe de multiplicacao

```
{
```

```
Console.WriteLine("\n {0} x {1} = {2}\n", num1, num2, num1 * num2); //
```

Faz a conta

```
Console.ReadKey();
```

```
}
```

public void divisao (double num1, double num2) // Classe de divisão

```
{
```

```
Console.WriteLine("\n {0} / {1} = {2}\n", num1, num2, num1 / num2); //
```

Faz a conta

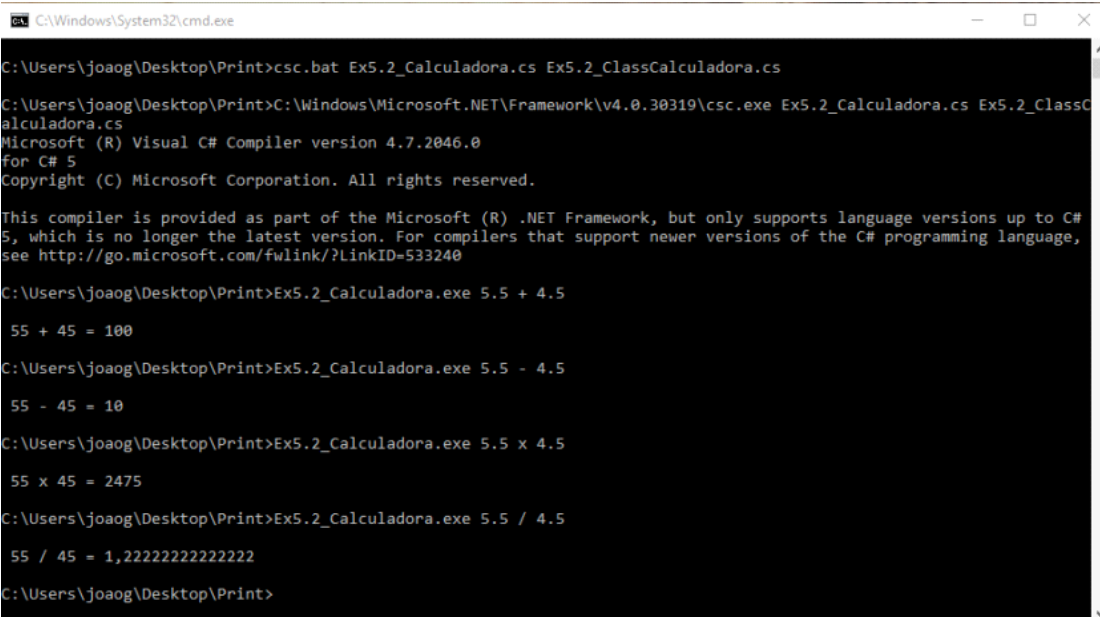
```
Console.ReadKey();
```

```
}
```

```
}
```

```
}
```

TELA:



```
C:\Windows\System32\cmd.exe
C:\Users\joaog\Desktop\Print>csc.bat Ex5.2_Calculadora.cs Ex5.2_ClassCalculadora.cs
C:\Users\joaog\Desktop\Print>C:\Windows\Microsoft.NET\Framework\v4.0.30319\csc.exe Ex5.2_Calculadora.cs Ex5.2_ClassCalculadora.cs
Microsoft (R) Visual C# Compiler version 4.7.2046.0
for C# 5
Copyright (C) Microsoft Corporation. All rights reserved.

This compiler is provided as part of the Microsoft (R) .NET Framework, but only supports language versions up to C# 5, which is no longer the latest version. For compilers that support newer versions of the C# programming language, see http://go.microsoft.com/fwlink/?LinkID=533240

C:\Users\joaog\Desktop\Print>Ex5.2_Calculadora.exe 5.5 + 4.5

55 + 45 = 100
C:\Users\joaog\Desktop\Print>Ex5.2_Calculadora.exe 5.5 - 4.5

55 - 45 = 10
C:\Users\joaog\Desktop\Print>Ex5.2_Calculadora.exe 5.5 x 4.5

55 x 45 = 2475
C:\Users\joaog\Desktop\Print>Ex5.2_Calculadora.exe 5.5 / 4.5

55 / 45 = 1,22222222222222
C:\Users\joaog\Desktop\Print>
```

11.1. Exercício 5.3

// Nome do programa: Ex5.3_Fatorial

// Programador(es): João Gontijo e Arthur Henrique

// Data: 19/08/2017

// Entrada(s): Um número para fatorar

// Saída(s): Um número fatorado

// Para executar digite: csc.bat Ex5.3_Fatorial.cs Ex5.3_ClassFatorial.cs

// Para testar digite: Ex5.3_Fatorial.exe 4

// Descrição: Recebe um número e o fatora

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Ex5_3_Fatorial

{

class Program

{

static void Main(string[] args)

{

if (args.Length != 1)

{

Console.WriteLine("\n PARÂMETRO INVÁLIDO!\n"); // Erro de

parâmetro

Console.ReadKey();

}

else

{

Console.Clear();

```

        int num1;
        bool teste = int.TryParse(args[0], out num1); // Verificação
de erro

        ClassFatorial fat = new ClassFatorial(); // Cria classe

        if (teste == true)
        {

            num1 = int.Parse(args[0]); // Converte args

            fat.fatorial(num1); // Chama a classe

        }
        else
        {
            Console.WriteLine("\n PARÂMETRO INVÁLIDO!\n"); //
Erro de parâmetro
            Console.ReadKey();
        }
    }
}
}
}
}
}
}
}

```

11.2. Classe do Exercício 5.3

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ex5_3_Fatorial

```

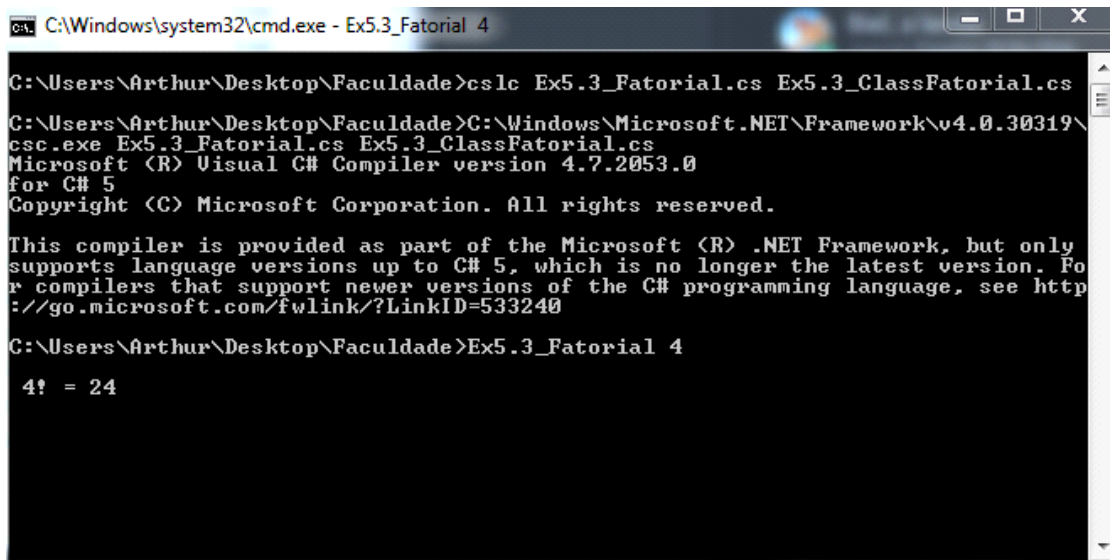
```

{
class ClassFatorial
{
    public void fatorial (int num1) // Classe de fatorial
    {
        int mostrar = num1;

        for (int i = num1 - 1; i > 1; i--) // Fatorial
        {
            num1 *= i; // Cálculo
        }

        if (num1 == 0)
        {
            Console.WriteLine("\n {0}! = {1}\n", mostrar, 1); // Imprime
resultado
            Console.ReadKey();
        }
        else
        {
            Console.WriteLine("\n {0}! = {1}\n", mostrar, num1); // Imprime
resultado
            Console.ReadKey();
        }
    }
}
}
}
TELA:

```

```

C:\Windows\system32\cmd.exe - Ex5.3_Fatorial 4

C:\Users\Arthur\Desktop\Faculdade>csc Ex5.3_Fatorial.cs Ex5.3_ClassFatorial.cs
C:\Users\Arthur\Desktop\Faculdade>C:\Windows\Microsoft.NET\Framework\v4.0.30319\
csc.exe Ex5.3_Fatorial.cs Ex5.3_ClassFatorial.cs
Microsoft (R) Visual C# Compiler version 4.7.2053.0
for C# 5
Copyright (C) Microsoft Corporation. All rights reserved.

This compiler is provided as part of the Microsoft (R) .NET Framework, but only
supports language versions up to C# 5, which is no longer the latest version. Fo
r compilers that support newer versions of the C# programming language, see http
://go.microsoft.com/fwlink/?LinkID=533240

C:\Users\Arthur\Desktop\Faculdade>Ex5.3_Fatorial 4

4! = 24

```

12.1. Exercício 5.4

```

// Nome do programa: Ex5.4_Conversor.cs
// Programador(es): João Gontijo e Arthur Henrique
// Data: 21/08/2017
// Entrada(s): Uma temperatura em Celsius ou Fahrenheit
// Saída(s): A temperatura convertida para Celsius ou Fahrenheit
// Para executar digite: csc.bat Ex5.4_Conversor.cs Ex5.4_ClassConversor.cs
// Para testar digite: Ex5.4_Conversor.exe
// Descrição: Conversor de temperaturas, Celsius e Fahrenheit

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

```

```

namespace Ex5_4_Conversor
{
    class Program
    {

```

```

static void Main(string[] args)
{
    Console.Clear();
    // Variáveis
    int opcao = 0;
    double temperatura;

    string auxOpcao;
    string auxTemperatura;

    ClassConversor cnv = new ClassConversor(); // Criar classe

    while (opcao != 3)
    {
        Console.Clear();
        // Menu
        Console.WriteLine("\n 1. Converter de Celsius para
Fahrenheit\n 2. Converter de Fahrenheit para Celsius\n 3. Sair do programa\n
Digite sua opção: ");
        auxOpcao = Console.ReadLine();

        bool teste = int.TryParse(auxOpcao, out opcao); // Teste de
erro

        if (teste == true && (opcao == 1 || opcao == 2)) //
Verificação
        {
            Console.WriteLine("\n Digite a temperatura à converter: "); //
Pedir a temperatura
            auxTemperatura = Console.ReadLine();

            bool teste2 = double.TryParse(auxTemperatura, out
temperatura); // Teste de erro

```

```

Console.Clear();

if (teste == true && teste2 == true)
{
    switch(opcao)
    {
        case 1:

            cnv.conversor(temperatura,'c'); // Chama a
classe

            break;

        case 2:

            cnv.conversor(temperatura,'f'); // Chama a
classe

            break;

        case 3:
            break;

        default:

            Console.WriteLine("\n OPÇÃO INVÁLIDA!"); //
Informa erro

            Console.ReadKey();
            Console.Clear();

            break;
    }
}
}
}

```

```

        else if (opcao != 3)
        {
            Console.WriteLine("\n OPÇÃO INVÁLIDA!"); // Informa
erro
            Console.ReadKey();
            Console.Clear();
        }

        Console.Clear();
    }
}
}
}

```

12.2. Classe do Exercício 5.4

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Ex5_4_Conversor
{
    class ClassConversor
    {
        public void conversor (double temp, char tipo) // Classe de conversão
        {

            if ((tipo == 'f' || tipo == 'F') && temp > -459.76)
            {
                Console.WriteLine("\n {0} graus Fahrenheit = {1} graus
Celsius\n", temp, (temp - 32) / 1.8); // Conversão de F para C
                Console.ReadKey();
            }
        }
    }
}

```

```

    }
    else if ((tipo == 'c' || tipo == 'C') && temp > -273.15)
    {
        Console.WriteLine("\n {0} graus Celsius = {1} graus
Fahrenheit\n", temp, (temp * 1.8) + 32); // Conversão de C para F
        Console.ReadKey();
    }
    else
    {
        Console.WriteLine("\n TEMPERATURA INVÁLIDA\n"); //
Mensagem de erro
        Console.ReadKey();
    }
}
}
}
TELA:

```

```
C:\Windows\System32\cmd.exe
Microsoft Windows [versão 10.0.15063]
(c) 2017 Microsoft Corporation. Todos os direitos reservados.

C:\Users\joaog\Desktop\Print>csc.bat Ex5.4_Conversor.cs Ex5.4_ClassConversor.cs

C:\Users\joaog\Desktop\Print>C:\Windows\Microsoft.NET\Framework\v4.0.30319\csc.exe Ex5.4_Conv
ersor.cs Ex5.4_ClassConversor.cs
Microsoft (R) Visual C# Compiler version 4.7.2046.0
for C# 5
Copyright (C) Microsoft Corporation. All rights reserved.

This compiler is provided as part of the Microsoft (R) .NET Framework, but only supports lang
uage versions up to C# 5, which is no longer the latest version. For compilers that support n
ewer versions of the C# programming language, see http://go.microsoft.com/fwlink/?LinkID=533240

C:\Users\joaog\Desktop\Print>Ex5.4_Conversor.exe

1. Converter de Celsius para Fahrenheit
2. Converter de Fahrenheit para Celsius
3. Sair do programa
Digite sua opção: 1

Digite a temperatura à converter: 50

50 graus Celsius = 122 graus Fahrenheit

1. Converter de Celsius para Fahrenheit
2. Converter de Fahrenheit para Celsius
3. Sair do programa
Digite sua opção: 2

Digite a temperatura à converter: 122

122 graus Fahrenheit = 50 graus Celsius

1. Converter de Celsius para Fahrenheit
2. Converter de Fahrenheit para Celsius
3. Sair do programa
Digite sua opção: 3

C:\Users\joaog\Desktop\Print>
```