

Algoritmos

Roteiro 8 para Laboratório

Professor: Humberto Nigri

Alunos: João Pedro Botelho e Kamilly Oliveira

1. Faça um programa que contenha uma função que receba dois números inteiros como parâmetro de entrada e leia um valor do teclado que esteja entre os dois parâmetros passados. A função deve retornar o número correto para o programa principal e este deve imprimir o valor lido.

Exemplo:

```
// digite aqui a função le_no_intervalo()
```

```
void Main(String[] args)
```

```
{
    int dia, mes;
    Console.WriteLine("Informe o dia entre 1 e 31: ");
    dia = le_no_intervalo(1, 31);
    Console.WriteLine("Informe o mes entre 1 e 12: ");
    mes = le_no_intervalo(1, 12);
    Console.WriteLine("Data: {0}/{1}", dia, mes);
}
```

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

```
{
    int dia, mes;
    Console.WriteLine("\nInforme o dia entre 1 e 31: ");
    dia = le_no_intervalo(1, 31);
    Console.WriteLine("Informe o mes entre 1 e 12: ");
    mes = le_no_intervalo(1, 12);
    Console.WriteLine("Data: {0}/{1}", dia, mes);
}
```

```
static int le_no_intervalo(int n1, int n2) {
```

```
    int valor=0;
    do{
        valor = int.Parse(Console.ReadLine());
        if(valor<n1 || valor>n2)
            Console.WriteLine("Valor inválido. Insira outro número");
    } while (valor<n1 || valor>n2);
    return valor;
}
```

2. Faça uma função que receba um número inteiro positivo como parâmetro, e caso este número seja primo retorne 1, caso contrário retorne 0. Número primo é aquele que só é divisível por 1 e por ele mesmo.

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

```
{
    Console.WriteLine("\nInsira um número inteiro");
    int result = isPrimo(int.Parse(Console.ReadLine()));

    if(result == 1)
        Console.WriteLine("É primo");
    else
        Console.WriteLine("Não é primo");
}
```

```
static int isPrimo(int n1) {
    for(int i=n1-1; i>1; i--) {
        if(n1%i == 0)
            return 0;
    }
    return 1;
}
```

3. Faça uma função que receba três números inteiros como parâmetros, representando horas, minutos e segundos, e retorne o seu valor convertido em segundos.

Exemplo: 2h, 40 min e 10 seg = 9.610 segundos

```
static void Main(string[] args)
{
    Console.WriteLine("\n{0}h, {1}min e {2}seg = ", DateTime.Now.Hour, DateTime.Now.Minute,
DateTime.Now.Second);

    Console.WriteLine("{0} segundos", convertToSegs(DateTime.Now.Hour, DateTime.Now.Minute,
DateTime.Now.Second));
}
```

```
static int convertToSegs(int hh, int mm, int ss) {
    return ss + (mm*60) + (hh*60*60);
}
```

4. Faça uma função que receba como parâmetro de entrada, a quantidade de segundos e, calcule e retorne como parâmetros de saída, três inteiros representando a: hora, minutos e segundos equivalentes.

Exemplo: 9.610 segundos = 2h, 40 min e 10 seg.

Use como parâmetros de saída parâmetros do tipo: (ref int var)

```
static void Main(string[] args)
{
    Console.WriteLine("\n{0} segundos = ", DateTime.Now.Hour + (DateTime.Now.Minute*60) +
(DateTime.Now.Hour*60*60));

    int hora=0, min=0, seg=0;
    convertToHourMinSeg(DateTime.Now.Hour + (DateTime.Now.Minute*60) +
(DateTime.Now.Hour*60*60), ref hora, ref min, ref seg);

    Console.WriteLine("{0}h, {1}min e {2}seg = ", hora, min, seg);
}
```

```
static void convertToHourMinSeg(int segundos, ref int hh, ref int mm, ref int ss) {
    hh = segundos/3600;
    mm = (segundos-(hh*60*60))/60;
    ss = (segundos-(hh*60*60)-(mm*60));
}
```

5. Faça uma função que receba três números representando o dia, mês e ano de uma data. A função deverá calcular quantos dias se passaram entre esta data e o dia 01/01/1900. Para simplificar as coisas considere os meses: 1, 3, 5, 7, 8, 10 e 12 como sendo de 31 dias e os demais 30. Para os “avançadinhos” façam para o caso real, ou seja: fevereiro tem 28 dias exceto em anos bissextos (a cada 4 anos)

```
static void Main(string[] args)
{
    int dias = calcDias(DateTime.Now.Day, DateTime.Now.Month, DateTime.Now.Year);
    Console.WriteLine("\nDesde o dia 01/01/1900 já se passaram {0} dias", dias);
}
```

```
static int calcDias(int dia, int mes, int ano) {
```

```

    int dias=dia, bi=4;
    for(int i=1900; i<ano; i++) {
        dias = bi%4==0 ? dias+366 : dias+365;
        bi++;
    }
    for(int i=1; i<mes; i++){
        if(i==2){
            dias = bi%4==0 ? dias+29 : dias+28;
            continue;
        }
        dias = i==4||i==6||i==9||i==11 ? dias+30 : dias+31;
    }
    return dias;
}

```

6. Faça agora uma função que receba uma quantidade de dias corridos e os transforme em dias, meses e anos (retornados como parâmetros de saída), tendo como dia Zero o dia 01/01/1900. Datas negativas são anteriores a este dia.
Para testar essa função utilize a função do exercício anterior para calcular quantos dias, meses e anos entre o dia de hoje e algumas datas especiais tais como:

```

static void Main(string[] args)
{
    Console.WriteLine("\nInsira o valor dos dias");
    int dias = int.Parse(Console.ReadLine());

    int dia=0, mes=0, ano=0;
    convertToDayMonthYear(dias, ref dia, ref mes, ref ano);

    Console.WriteLine("{0} dias, a partir do dia 01/01/1900 é equivalente a: {1}/{2}/{3}", dias, dia, mes,
ano);
}

static void convertToDayMonthYear(int dias, ref int dd, ref int mm, ref int aa) {
    if(dias >= 0) {
        aa = 1900 + dias/365;
        mm = 1 + (dias%365)/30;
        dd = 1 + (dias%365)%30;
    } else {
        aa = 1899 - (Math.Abs(dias)/365);
        mm = 12 - (Math.Abs(dias)%365)/30;
        dd = 30 - (Math.Abs(dias)%365)%30;
    }
}

```

- Dia do seu nascimento
- Dia em que você começou a namorar
- Dia em que você começou estudar para essa disciplina (ontem ??)
- Ultima vez que viu o seu time ganhar (já faz tempo ? rsrs)
- Etc