

EXERCÍCIOS

Nota	
Engenharia Civil	Resolver os problemas utilizando linguagem C.
Engenharia de Computação	Resolver os problemas utilizando linguagem Java.

1. Desenvolva uma função com a seguinte assinatura:

```
public static int megasena(int[] cartela, int[] sorteio)
```

Desenvolva uma função que receba dois vetores: o primeiro representando os números escolhidos em uma cartela da Mega Sena, e o segundo representando os números sorteados. A função deverá retornar um inteiro indicando a quantidade de números que foram sorteados na cartela.

2. Desenvolva uma função com a seguinte assinatura:

```
public static double[] calcularX(int n)
```

Desenvolva uma função que receba um inteiro N . A função deverá retornar um vetor *double* contendo em cada elemento o resultado da seguinte expressão:

$$x(i) = 0.9^i$$

onde i varia de $[-\pi, \pi]$ com intervalo de n .

Exemplo: se $n = 0.1$ então o intervalo $[-\pi, \pi]$ irá variar de 0.1 em 0.1.

3. Desenvolva uma função com a seguinte assinatura:

```
public static double[] calcularY(double[] x)
```

Desenvolva uma função que receba um vetor *double* contendo os valores de x . A função deverá retornar um vetor *double* contendo os valores de y .

O valor de y pode ser obtido pela seguinte expressão:

EXERCÍCIOS

$$y(k) = \sum_{n=0}^{N-1} \frac{x(n)}{|k-n|}$$

Onde:

- $k \rightarrow$ varia de $[1, 10]$ com intervalo de 1.

4. Desenvolva uma função com a seguinte assinatura:

```
public static int[] listaEmails(String[] emails)
```

Desenvolva uma função que receba um vetor de String contendo uma lista de e-mails. A função deverá retornar um vetor contendo a quantidade de e-mails pertencentes ao: Gmail; Hotmail; Yahoo; Demais serviços.

5. Desenvolva uma função com a seguinte assinatura:

```
public static boolean celularValido(String numero)
```

Desenvolva uma função que receba uma String, essa entrada corresponde a um número de celular. A função deverá retornar se este é um número válido considerando que o número pertence ao estado de Goiás. Retornará **true** se o número for válido. Do contrário **false**.

Para que o número seja válido é necessário verificar:

- Se o primeiro dígito começa com o número 9;
- Se a sequência após primeiro dígito do telefone começa com 6, 7, 8, 9, 96, 97, 98 ou 99;
- Se todos os caracteres são números;
- Se a quantidade de números é igual a 9.

6. RETIRADO PARA AVALIAÇÃO

7. Desenvolva uma função com a seguinte assinatura:

```
public static int[] justMax(int[] a, int[] b)
```

EXERCÍCIOS

Desenvolva uma função que receba dois vetores do tipo inteiro. A função deverá retornar um novo vetor que será composto pelo maior elemento da posição i de A ou B. Exemplo:

$a[0] = 1500$ $b[0] = 1850$ $c[0] = 1850$
 $a[1] = 5000$ $b[1] = 3215$ $c[1] = 5000$

8. Desenvolva uma função com a seguinte assinatura:

```
public static int[] fetuccine(int n, int a0, int a1)
```

A função deverá receber três parâmetros: a quantidade e termos, o primeiro e segundo termo. A função deverá retornar um vetor contendo em seus elementos cada termo da série de FETUCCINE.

A série de FETUCCINE é gerada da seguinte forma: os dois primeiros termos são fornecidos pelo usuário; a partir daí, os termos são gerados com a soma ou subtração dos dois termos anteriores, ou seja:

$$A_i = A_{i-1} + A_{i-2} \text{ para } i \text{ ímpar}$$

$$A_i = A_{i-1} - A_{i-2} \text{ para } i \text{ par}$$

Sabe-se que para existir esta série serão necessários pelo menos três termos, ou seja, a função retornará um vetor vazio caso n seja menor que 3.

9. Desenvolva uma função com a seguinte assinatura:

```
public static int quantDias(int anoInicial, int anoFinal)
```

A função deverá receber dois valores inteiros correspondendo o ano inicial e ano final. Ela irá retornar o número de dias decorridos entre esses dois anos.

Considere a ocorrência de anos bissextos.

10. RETIRADO PARA AVALIAÇÃO

11. Desenvolva uma função com a seguinte assinatura:

EXERCÍCIOS

```
public static void repeticoes(int[] sequencia)
```

A função deverá receber um vetor de inteiro. Não haverá retorno. Ela irá determinar e imprimir os números que compõem a sequência e o número de vezes que cada um deles ocorre na mesma.

Exemplo:

Entrada →

sequencia = [4, 5, 6, 3, 7, 4, 5, 5, 5, 9, 1, 6]

Saída →

4 ocorre 2 vezes
5 ocorre 4 vezes
6 ocorre 2 vezes
3 ocorre 1 vez
7 ocorre 1 vez
9 ocorre 1 vez
1 ocorre 1 vez

12. Desenvolva uma função com a seguinte assinatura:

```
public static int[] pascal(int n)
```

A função deverá receber um valor inteiro representando a linha do Triângulo de Pascal a ser retornada. A função deverá retornar um vetor contendo em cada elemento os valores da linha escolhida.

O Triângulo de Pascal funciona da seguinte maneira:

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
```