

<b>Universidade Federal do ABC</b>	<b>Avaliação:</b> Prova 02
<b>Disciplina:</b> Processamento da Informação	
<b>Professor:</b> Jesús P. Mena-Chalco	<b>Data:</b> 22/08/2018
Nome:	RA:

**Instruções para a prova (leia antes de começar):**

- 1) A prova tem a duração de 1h50min.
- 2) A prova é individual.

**Questão 1** (6 pontos)

Escreva uma função que permita juntar (concatenar) 3 vetores de números inteiros

Definição da função: `static int[] concatenar ( int L1[], int L2[], int L3[] )`

Exemplo:

Para os vetores L1=[11, 22], L2=[55, 44, 11], L3=[88, 99, 22, 44] a resposta deve ser [11, 22, 55, 44, 11, 88, 99, 22, 44].

**Questão 2** (4 pontos)

Escreva uma função que permita imprimir os 100 primeiros números inteiros, **pares** maiores que zero e múltiplos de 3 e de 5.

Definição da função: `static void imprimirNumeros( )`

**Questão 3** (6 pontos)

Qual é o resultado da execução do seguinte método?

```
static void triangulo () {
    int i, j;
    int M[][] = new int[6][6];

    for (i=5; i>=0; i=i-1) {
        for (j=i; j<6; j=j+1) {
            if (i==j) {
                M[i][j] = 1;
            }
            else {
                M[i][j] = M[i+1][j] + M[i][j-1];
            }
        }
    }

    for (i=0; i<6; i=i+1) {
        for (j=0; j<6; j=j+1) {
            System.out.print(M[i][j]+" ");
        }
        System.out.print("\n");
    }
}
```

**Questão 4** (6 pontos)

Escreva uma função que permita verificar se existem pelo menos 5 números iguais em um vetor de números inteiros.

Definição da função: `static boolean cinco( int v[] )`

Exemplos:

Para o vetor [ 11, 22, 11, 33 ] a resposta deve ser false
Para o vetor [ 11, 22, 11, 22, 33, 44, 55, 22, 66, 22, 22 ] a resposta deve ser true
Para o vetor [ 55, 55, 55, 55, 55 ] a resposta deve ser true

**Questão 5** (6 pontos)

Crie uma função para determinar o menor valor presente na diagonal secundária de uma matriz (quadrada ou retangular) dada como entrada.

Definição da função: `static int menorValor( int M[][] )`

**Limitação:** Sua implementação deve ser eficiente e somente os elementos que estão na diagonal deverão ser consultados. Não serão aceitas soluções que considerem laços aninhados (apenas 1 laço é suficiente).

**Questão 6** (6 pontos)

Escreva uma função que permita verificar se existem pelo menos duas linhas idênticas.

Definição da função: `static boolean linhasIdenticas( int M[][] )`

Exemplos:

1	1	1	1	1
2	2	2	2	2
3	3	3	3	3

false

1	2	3
1	2	3
1	2	3

true

**Questão 7** (6 pontos)

Considere o método `particiona`. Avalie o método com os seguintes elementos

`p = 0`

`r = 9`

`A = [99, 33, 55, 77, 11, 22, 88, 66, 33, 44]`

```

static int particiona ( int A[], int p, int r ) {
    int i, j, x, aux;

    x = A[r];
    i = p-1;

    for (j=p; j<=r-1; j=j+1) {
        if (A[j]>=x) {
            i = i+1;
            aux = A[i];
            A[i] = A[j];
            A[j] = aux;
        }
    }

    aux = A[i+1];
    A[i+1] = A[r];
    A[r] = aux;

    return i+1;
}

```

(a) Qual a resposta do método (i.e., indique o número inteiro que o método devolve): [2 pontos]

.....

(b) Apresente o vetor A após a execução do método. [4 pontos]

A = .....