



Fundação CECIERJ - Vice-Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**  
**Disciplina Fundamentos de Programação**

**AP3 2º semestre de 2016**

---

**IMPORTANTE**

- Prova sem consulta e sem uso de qualquer aparato eletrônico.
  - Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
  - Você pode usar lápis para responder as questões.
  - Ao final da prova, devolva as folhas de questões e as de respostas.
  - Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
- 

**1ª Questão (3,5 pontos)**

Utilizando subprogramação, faça o programa a seguir:

- (1) Peça ao usuário as dimensões, quantidade de linhas e quantidade de colunas, de uma matriz bidimensional de números inteiros, de conteúdo a ser gerado aleatoriamente, num intervalo também pedido ao usuário;
- (2) Produza a matriz com números gerados pela função `randint(mínimo, máximo)`, que retorna um número inteiro aleatório dentro do intervalo mínimo e máximo;
- (3) Mostre a matriz de forma amigável, isto é, cada linha escrita deve ser composta apenas pelos números na respectiva coluna, separados por espaço em branco;
- (4) Mostre o conteúdo da linha que inclui o maior valor (em caso de empate, mostre uma delas); e
- (5) Mostre o conteúdo da coluna com maior soma (em caso de empate, mostre uma delas) e o valor dessa soma.

## Exemplo

Entradas	Saída
Dimensões da matriz: 3 4 Intervalo para geração aleatória: 20 95	Matriz: 35 82 23 13 26 80 44 95 21 92 85 47  Conteúdo da linha com maior valor: 26 80 44 95  Conteúdo da coluna com maior soma: 82 80 92

### 2ª Questão (3,0 pontos)

Utilize **recursão** para resolver os seguintes problemas:

- a) Escreva um programa que receba, via entrada padrão (teclado), uma coleção de  $n$  valores inteiros e ative a função `analisarPadrao(x)` que retorna `True` caso os valores colocado na lista `x` atendam à seguinte regra:

$$x[n-i-1] == \text{mod}(2*x[i], 10)$$

para todo índice  $i$ , onde  $i \leq n-i-1$ , e `False` caso contrário. Seu programa deverá imprimir, na saída padrão (vídeo), as mensagens “Padrão válido” ou “Padrão inválido”, conforme o retorno de `analisarPadrao`.

Por exemplo, tanto a lista `x = [7 0 4]` quanto a lista `x = [1 3 4 8 6 2]` levam à impressão da mensagem “Padrão válido”.

Você deve implementar a função `analisarPadrao` como uma função recursiva.

- b) Escreva um programa recursivo que calcule o máximo divisor comum (MDC) entre dois números inteiros. O MDC de dois ou mais números inteiros é o maior número inteiro que é fator de tais números.

O Algoritmo Euclidiano para cálculo do MDC de dois números  $a$  e  $b$  mostra que:

$$\text{mdc}(a, b) == \text{mdc}(b, a \% b), \text{ para } b > 0 \text{ e}$$

$$\text{mdc}(a, 0) == a, \text{ caso contrário.}$$

Seu programa deverá receber, via entrada padrão (teclado), os valores inteiros  $a$  e  $b$  e ativar a função `mdc(a, b)` que retorna o máximo divisor comum. O valor retornado deverá ser impresso pelo programa principal na saída padrão (vídeo).

Por exemplo, `mdc(12, 18) == mdc(18, 12) == mdc(12, 6) == mdc(6, 0) == 6`.

Você deve implementar a função `mdc` como uma função recursiva.

### 3ª Questão (3,5 pontos)

Dado um arquivo texto de nome informado pelo usuário, contendo em cada linha um par de coordenadas que definem a localização de pontos (x, y), implemente um programa que atenda à seguinte especificação:

- (1) Peça para o usuário informar via entrada padrão (teclado) o nome do arquivo texto contendo as coordenadas dos pontos;

- (2) Declare e ative o subprograma

```
def centroideDosPontos(arquivo):
```

que abre o arquivo texto de nome informado, lê o conteúdo desse arquivo e calcula o ponto médio (centroide) dos pontos, imprime as coordenadas do centroide na saída padrão (tela) mostrando duas casas decimais, fecha o arquivo e retorna a tupla com o par de coordenadas médias calculadas;

- (3) Repetidamente, no programa principal, receba do usuário o tamanho de um raio, até que um raio negativo seja informado. Para cada raio não negativo informado, invoque o subprograma

```
def pontosNaCircunferencia(arquivo, centroide, raio):
```

que abre e processa o arquivo novamente escrevendo na saída padrão (vídeo) as coordenadas dos pontos dentro da circunferência definida pelo centroide e pelo raio atual. Lembre-se de fechar o arquivo antes de sair do subprograma. As coordenadas dos pontos devem ser impressas mostrando duas casas decimais.

Não é permitido manter o conteúdo de todo o arquivo armazenado ao mesmo tempo na memória principal, isto é, apenas uma linha do arquivo pode estar em processamento em um dado instante.

#### Exemplo

Entrada	Saída
pontos.txt 3.0 5.0 -1.0	Centroide: x = 5.40, y = 2.28  Pontos na circunferência de raio 3.00 x = 5.00, y = 3.20  Pontos na circunferência de raio 5.00 x = 5.00, y = 3.20 x = 9.50, y = 4.70 x = 9.00, y = 2.00

#### Conteúdo do Arquivo

pontos.txt
5.0 3.2
4.8 -6.3
9.5 4.7
9.0 2.0
-1.3 7.8

**Dica:** Em Python, a função que calcula a raiz quadrada de um determinado valor está declarada no módulo `math` (`math.sqrt(valor)`).

**Boa Avaliação!**