

Prova 2

02/12/2017

08:00 –10:00h

Questão 1

(6 pontos)

Considere o seguinte programa em Python:

```
import numpy as np

def montaMatriz( m, n ):
    x = np.empty( (m,n), dtype=int )
    k = 1
    for i in range( 0, m ):
        for j in range( 0, n ):
            x[i][j] = k
            k = k + 1
    return x

def s1Matriz( x, i ):
    m, n = x.shape # Pega número de linhas e colunas da matriz
    a = 0
    for j in range( 0, n ):
        a = a + x[i][j]
    return a

def s2Matriz( x, j ):
    m, n = x.shape # Pega número de linhas e colunas da matriz
    a = 0
    for i in range( 0, m ):
        a = a + x[i][j]
    return a

A = montaMatriz( 2, 2 )
for i in range( 0, 2 ):
    print( s1Matriz( A, i ))
for j in range( 0, 2 ):
    print( s2Matriz( A, j ))
```

Escreva abaixo qual será a saída exata fornecida pelo programa:

Tela do Computador
3 7 4 6

Rascunho (rastreo das variáveis)

Questão 2**(10 pontos)**

O MMC – Mínimo Múltiplo Comum – entre dois números inteiros A e B é o menor número inteiro que é múltiplo tanto de A quanto de B, excluindo o zero. Exemplos:

A	B	MMC(A, B)
4	6	12
30	3	30
8	6	24
4	15	60

O programa a seguir faz a leitura de dois números inteiros pelo teclado e imprima o MMC entre os dois. Você deve completar o código da função `mmc()` para que ele funcione corretamente.

```
def mmc( x, y ):  
  
    mc = m = max( x, y )  
    while mc % x > 0 or mc % y > 0:  
        mc = mc + m  
    return mc  
  
ou...  
  
    mc = max( x, y )  
    while mc % x > 0 or mc % y > 0:  
        mc = mc + 1  
    return mc  
  
def programa_principal():  
    a = int( input('A = '))  
    b = int( input('B = '))  
    print( mmc(a, b) )  
  
programa_principal()
```

Questão 3

(12 pontos)

Em um determinado concurso, vários candidatos fizeram uma prova de múltipla escolha. O edital do concurso previa que a vaga seria preenchida pelo candidato que obtivesse a maior nota. Mas o estagiário que redigiu o edital se esqueceu de incluir a regra de desempate! Agora que o concurso já foi realizado, os responsáveis precisam saber o tamanho do problema que tem em mãos... descobrir quantas pessoas empataram com a maior nota!

Dentro do programa você já dispõe do código (100% correto) que lê o arranjo **v** contendo a nota de cada candidato, e determina a variável **n** contendo o número de valores dentro do arranjo **v** (veja abaixo).

Em seguida, o programa deve imprimir dois valores como resposta: a maior nota e quantos candidatos obtiveram essa nota.

Exemplo 1:

Entre com a nota de cada candidato (-1 termina):

20
25
2
12
25
-1

Maior nota: 25

Empates: 2

Exemplo 2:

Entre com a nota de cada candidato (-1 termina):

7
5
2
4
6
8
9
4
-1

Maior nota: 9

Empates: 1

Exemplo 3:

Entre com a nota de cada candidato (-1 termina):

5
5
5
5
-1

Maior nota: 5

Empates: 4

SOLUÇÃO:

```
# Leitura do vetor de notas e cálculo de n:
v = np.empty( 0, dtype=int )
print('Entre com a nota de cada candidato (-1 termina):')
n = 0
while True:
    x = int( input() )
    if x < 0:
        break
    v = np.append( v, x )
    n = n + 1
```

```
# Determinação do resultado:
```

Implementação 1

```
maior = notas[0]
for i in range( 1, n ):
    if notas[i] > maior:
        maior = notas[i]

nm = 0
for i in range( 0, n ):
    if notas[i] == maior:
        nm = nm + 1

print( maior, nm )
```

Implementação 2

```
maior = notas[0]
nm = 1
for i in range( 1, n ):
    if notas[i] > maior:
        maior = notas[i]
        nm = 1
    elif notas[i] == maior:
        nm = nm + 1

print( maior, nm )
```

Questão 4

(7 pontos)

Em uma aula prática, foi dada a seguinte tarefa:

Escreva um programa que faça o seguinte:

1. Lê um valor inteiro n pelo teclado e cria uma matriz n x n de valores aleatórios;
2. Calcula e escreve na tela a média e o desvio padrão dos valores da matriz, além da média - desvio padrão e a média + desvio padrão;
3. Calcula e escreve na tela uma matriz C, tal que $C_{ij} = 0$ se $A_{ij} < (\text{Média} - \text{DP})$, $C_{ij} = 0$ se $A_{ij} > (\text{Média} + \text{DP})$, e $C_{ij} = 1$ caso contrário.

onde:
$$DP = \sqrt{\frac{\sum_i \sum_j (x_{ij} - \bar{x})^2}{n^2}}$$
 sendo \bar{x} a média dos valores.

E de acordo com o enunciado, o programa deveria fornecer a seguinte saída:

Entre com a Ordem da matriz: 4

Matriz A:

8.76	15.13	0.00	6.35
3.08	1.94	3.91	7.26
8.33	11.32	8.80	14.39
4.29	18.44	0.58	14.08

Média: 7.92

Desvio padrão: 5.38

Média - DP: 2.53

Média + DP: 13.30

Matriz C:

1	0	0	1
1	0	1	1
1	1	1	0
1	0	0	0

No entanto, após um aluno implementar e executar o programa, ele obteve a seguinte saída:

Entre com a Ordem da matriz: 4

Matriz A:

8.76	15.13	0.00	6.35
3.08	1.94	3.91	7.26
8.33	11.32	8.80	14.39
4.29	18.44	0.58	14.08

Média: 5.04

Desvio padrão: 5.23

Média - DP: -0.19

Média + DP: 10.27

Matriz C:

1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

O aluno então observou os seguintes problemas:

1. O valor da média e do DP estão incorretos;
2. Na montagem da matriz C, alguns valores que deveriam ser iguais a 0 estão como 1.

Segue abaixo o código implementado pelo aluno. Descreva quais foram os erros cometidos e como eles podem ser corrigidos. Use os números das linhas para facilitar a sua resposta.

```
1  import numpy as np
2
3  def imprimeMatriz( M, titulo, formato ):
4      m, n = M.shape
5      if titulo != '': print( titulo )
6      for i in range( 0, m ):
7          for j in range( 0, n ):
8              print( formato % M[i][j], end='')
9          print()
10
11  n = int( input('Entre com a Ordem da matriz: ') )
12  np.random.seed( 1 )
13  A = np.random.uniform( 0, 21, (n,n) )
14  imprimeMatriz( A, '\nMatriz A:', '%8.2f' )
15
16  soma = 0
17  i = 1
18  while i < n:
19      j = 1
20      while j < n:
21          soma = soma + A[i][j]
22          j = j + 1
23      i = i + 1
24  media = soma / n**2
25  soma = 0
26  i = 1
27  while i < n:
28      j = 1
29      while j < n:
30          soma = soma + (A[i][j] - media) ** 2
31          j = j + 1
32      i = i + 1
33  desvpad = (soma / n**2) ** 0.5
34  print()
35  print('Média:          %5.2f' % media )
36  print('Desvio padrão: %5.2f' % desvpad )
37  print('Média - DP:     %5.2f' % (media - desvpad))
38  print('Média + DP:     %5.2f' % (media + desvpad))
39
40  C = np.empty( (n,n), dtype=int )
41  for i in range( 0, n ):
42      for j in range( 0, n ):
43          a = A[i][j]
44          if a < media-desvpad and a > media+desvpad:
45              C[i][j] = 0
46          else:
47              C[i][j] = 1
48  imprimeMatriz( C, '\nMatriz C:', '%8d' )
```

SOLUÇÃO:

- Nas linhas 17, 19, 26 e 28, as variáveis *i* e *j* precisam ser inicializadas como 0 em vez de 1;

- Na linha 44, deve-se usar o operador or no lugar do and:
if *a* < *media-desvpad* or *a* > *media+desvpad*: