

Exame de xaneiro de 2017 de programación en Python

Escrebe un programa en Python que realice as seguintes operacións:

1. Define unha variable enteira $n = 10$ e dúas variables reais $a = -2$ e $b = 2$. Calcule dous vectores \mathbf{x} e \mathbf{y} de dimensión n cos elementos dados polas expresións:

$$x_i = \frac{(b-a)i + an - b}{n-1} \quad i = 0, \dots, n-1 \quad (1)$$

$$y_i = x_i^2 e^{-x_i^2} - x_{i-1} \quad i = 1, \dots, n-1 \quad y_0 = 0 \quad (2)$$

2. Visualiza na pantalla os elementos de \mathbf{x} que son maiores ós elementos de \mathbf{y} , xunto cos índices nos que se cumpre a condición.
3. Define unha función `transforma(...)`, cos argumentos axeitados, que calcule un vector \mathbf{z} de dimensión n , onde

$$z_i = \sum_{j=0}^m y_j \sin(x_i x_j) \quad (3)$$

sendo m o primeiro número que cumpra

$$\sum_{j=0}^m y_j > \frac{i+1}{7} \quad (4)$$

4. Chama á función `transforma()` para calcular o vector \mathbf{z} , e axusta os puntos $\{x_i, z_i\}$, $i = 0, \dots, n-1$ a un polinomio de orde 3. Representa nun gráfico 2D os puntos $\{x_i, z_i\}$ con asteriscos verdes e o polinomio axustado como unha liña discontinua azul. Ponlle ó gráfico título, enreixado, etiquetas nos eixos e lendas. Gárdao no arquivo `grafico.png`.
5. Pide ó usuario o nome dun arquivo e garda nel os vectores \mathbf{x} , \mathbf{y} e \mathbf{z} cun ancho de 8 cifras e 3 decimais.

```
#!/usr/bin/python
#-*- coding: utf-8 -*-
from numpy import *
from matplotlib.pyplot import *
a, b, n=-2.0, 2.0, 10
t=arange(n)
x=((b-a)*t+a*n-b)/(n-1)
y=zeros(n)
for i in range(1, n):
    y[i]=x[i]*x[i]*exp(-x[i]*x[i])-x[i-1]
p=where(x>y)[0]
print 'Indices x>y: ', p
print 'Valores x>y: ', x[p]
def transforma(x, y):
    n=x.size
    umbral=float(i+1)/7
    z=zeros(n)
    for i in range(n):
        sz=0; sy=0
        for j in range(n):
            sz=sz + y[j]*sin(x[i]*x[j])
            sy=sy + y[j]
            if sy > umbral:
```

```

        break
    z[i]=sz
    # outra alternativa seria
    # sz=0; sy=0; j=0
    # while sy < umbral and j<n:
    #     sz=sz + y[j]*sin(x[i]*x[j])
    #     sy=sy + y[j]
    #     j=j+1
    # z[i]=sz
    return z
z=transforma(x,y)
p=polyfit(x,z, 3)
xx=linspace(min(x), max(x), 50)
clf(); plot(x,z, 'g*', label='Puntos')
plot(xx, polyval(p,xx), 'b--', label='Curva axustada')
title('Axuste a polinomio grao 3')
xlabel('x'); ylabel('z'); legend(loc='upper right')
grid(True); show(False); savefig('grafico.png')
nome=raw_input('Nome arquivo: ')
out=vstack([x, y, z])
try:
    savetxt(nome, out, fmt='%8.3f')
except IOError:
    print 'Erro escribindo en ', nome

```
