

# Grao en Física: Informática para científicos@s

## Exame de programación en Python de xuño de 2026

---

Escribe no editor varias liñas con números, como o exemplo seguinte, e gárdao co nome `datos.txt`:

```
1 2 3 4 5
7 8
9 8 7 6 5 4 3 2 1
2 3 4 5
6 7 8 9 8 7
```

Escribe un programa en Python chamado `exame.py` que realice o seguinte:

1. Lea o arquivo anterior liña a liña, almacenando nos vectores  $\mathbf{v}$  e  $\mathbf{w}$  os valores situados en posicións pares e impares da liña, respectivamente. Mostra  $\mathbf{v}$  e  $\mathbf{w}$ . Le por teclado e almacena nunha cadea de caracteres  $s$  a expresión analítica dunha función  $f(x)$ , podes usar a expresión `x**2*log(1+x**2)-4`. Crea o vector  $\mathbf{z}$  concatenando  $\mathbf{v}$  e  $\mathbf{w}$ , e define  $a$ ,  $b$  e  $n$  como o mínimo, máximo e lonxitude de  $\mathbf{z}$ .
2. Chama á función `func(·)`, cos argumentos axeitados, que calcule simbólicamente o valor de  $t = \int_a^b f(x)dx$ , e a matriz  $\mathbf{a}$  de orde  $n$ . Para calcular  $\mathbf{a}$ , crea primeiro o vector  $\mathbf{y}$  de lonxitude  $n$  cos elementos do vector  $\mathbf{z}$  barallados. Entón, cada elemento  $a_{ij}$ , con  $i, j = 0, 1, \dots, n - 1$  calcúlase percorrendo os elementos  $z_k$  de  $\mathbf{z}$ , comezando por  $k = \lfloor (i + 2j + 2)/3 \rfloor$  e facendo  $k = 0$  cando chegas ó final do vector  $\mathbf{z}$ , ate atopar un elemento  $z_k$  no vector  $\mathbf{z}$  maior ou igual que o elemento  $y_k$  no vector  $\mathbf{y}$ . O elemento  $a_{ij}$  será o número de elementos  $z_k$  percorridos.
3. De volta no programa principal, mostra  $t$  por pantalla con formato real de ancho 10 e 5 decimais. Representa nun gráfico en 3D a matriz  $\mathbf{a}$  cos eixos X e Y variando entre 1 e  $n$ , que na gráfica se deben chamar Temperatura ( $^{\circ}\text{K}$ ) e Presión (Pa), respectivamente. Pon a cadea de caracteres  $s$  do apartado 1 como título do eixo Z da gráfica.

```
from numpy import *
from numpy.random import *
from sys import *
# apartado 1-----
nf='datos.txt';v=array([]);w=array([])
try:
    f=open(nf)
except IOError:
    printf('erro open %s'%nf);exit()
for l in f:
    p=l.split();i=0
    for q in p:
        x=int(q)
        if i%2==0:
            v=append(v,x)
        else:
            w=append(w,x)
        i+=1
f.close()
print('v=',v,'\nw=',w)
s=input('f(x)? ')
z=concatenate((v,w))
a=min(z);b=max(z);n=len(z)
print('a=%d b=%d n=%d'%(a,b,n))
# apartado 2-----
```

```

import sympy as sp
def func(f,a,b,z,n):
    x=sp.symbols('x')
    t=sp.integrate(f,(x,a,b)).evalf()
    a=zeros([n,n]);y=permutation(z)
    for i in range(n):
        for j in range(n):
            k=int(floor((i+2*j+2)/3));l=0
            while z[k]<y[k]:
                k+=1;l+=1
                if k==n:
                    k=0
            a[i,j]=1
    return [t,a]
[t,a]=func(s,a,b,z,n)
# apartado 3-----
print('t=%10.5f'%t)
from pylab import *
from mpl_toolkits.mplot3d import Axes3D
x=arange(1,n+1);X,Y=meshgrid(x,x)
fig=figure(1);clf()
ax=Axes3D(fig,auto_add_to_figure=False)
fig.add_axes(ax)
ax.plot_surface(X,Y,a)
ax.set_xlabel('Temperatura (°K)');
ax.set_ylabel('Presión (Pa)')
ax.set_zlabel(s)
grid();show()

```