

Computación

Aula Virtual: <https://famaf.aulavirtual.unc.edu.ar/course/view.php?id=747>

Resguardo tutoriales: <https://www.famaf.unc.edu.ar/~moreschi/docencia/Computacion/>

Tutorial Problema 8 de la Guía N° 3

Problema 8:

Utilice su función `long_de_camino` para calcular π . Para ello recuerde que π es igual al perímetro de un círculo de radio $1/2$. Suponga que aproxima a la circunferencia por un polígono de $N+1$ vértices sobre el círculo. La longitud de éste polígono puede encontrarse usando la función `long_de_camino`. Note que puede generar $N + 1$ puntos sobre el círculo con las fórmulas:

$$x_i = \frac{1}{2} \cos\left(\frac{2\pi i}{N}\right), \quad y_i = \frac{1}{2} \sin\left(\frac{2\pi i}{N}\right), \quad i = 0, 2, 3, \dots, N$$

Grafique el camino de este problema para $N = 20$; mostrando el mismo en pantalla y guardándolo en archivo ‘.png’ en subdirectorio apropiado (ej.: "gráficos/").

Tutorial:

- Guarde en el archivo `p8-g3.py` las siguientes instrucciones:

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # primer cálculo -----
5 N = 200
6 print('N = ',N)
7
8 x=[]
9 y=[]
10
11 for i in range(N+1):
12     x.append(0.5*np.cos(2.*np.pi*i/N))
13     y.append(0.5*np.sin(2.*np.pi*i/N))
14
15
16 def long_de_camino(x,y):
17     long = 0
18     for i in range(len(x)-1):
19         long = long + pow((x[i] - x[i+1])**2 + (y[i] - y[i+1])**2 , 0.5)
20     return long
21
22
23 print('\nEl largo del camino es = %27.20e' %(long_de_camino(x,y)) )
24 print( '          El valor de pi es = %27.20e \n' %(np.pi) )
25
26 # segundo cálculo -----
27 N = 20
28 print('N = ',N)

```

```

29
30 x=[]
31 y=[]
32
33 for i in range(N+1):
34     x.append(0.5*np.cos(2.*np.pi*i/N))
35     y.append(0.5*np.sin(2.*np.pi*i/N))
36
37
38 def long_de_camino(x,y):
39     long = 0
40     for i in range(len(x)-1):
41         long = long + pow( (x[i] - x[i+1])**2 + (y[i] - y[i+1])**2 , 0.5)
42     return long
43
44
45 print('\nEl largo del camino es = %27.20e' %(long_de_camino(x,y)) )
46 print( '          El valor de pi es = %27.20e \n' %(np.pi) )
47
48 # gráfico -----
49 plt.figure(figsize=(7.5, 7.5))
50 plt.title('Cálculo de $\pi$')
51 plt.xlabel('x')
52 plt.ylabel('y')
53 plt.grid()
54 #plt.plot(v1[0],v1[1], 'r^',v2[0],v2[1], 'r^',v3[0],v3[1], 'r^')
55 plt.plot(x,y, 'ro')
56 plt.plot(x,y, 'b')
57 plt.savefig('graficos/p8-g3.png', dpi=150)
58 plt.show(block = True)

```

- Desde la terminal ejecute:

```
python3 p8-g3.py
```

e interprete el resultado.

Modifique el programa y pruebe otras cosas.