

# Función Python para la aproximación del número $\pi$

Técnicas Experimentales  
Práctica de Laboratorio #6

12 de marzo de 2014

## Resumen

El objetivo de esta práctica es definir una *función* en Python que permita la aproximación del valor de  $\pi$  con una precisión dada.

## 1. Motivación y Objetivos

El número  $\pi$  se puede aproximar numéricamente con la siguiente fórmula:

$$\pi \approx \frac{1}{n} \sum_{i=1}^n f(x_i), \text{ con } f(x) = \frac{4}{(1+x^2)}, x_i = \frac{i-\frac{1}{2}}{n}, \text{ para } i = 1, \dots, n$$

Un mayor número de intervalos hace mayor la precisión en los decimales de la aproximación de  $\pi$ . Por lo tanto, se concluye que  $n$  ha de constituir un *parámetro* de la función Python que aproxime  $\pi$ . En la Figura 1 se muestra gráficamente utilizando  $n = 4$ .

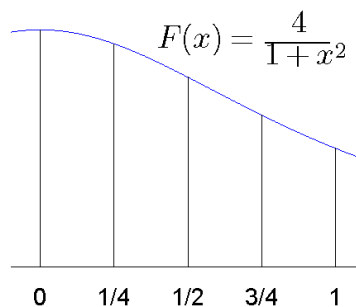


Figura 1: Aproximación de  $\pi$  numéricamente

## 2. Ejercicios propuestos

Escriba una función que reciba como parámetro el número de intervalos con los que se desea abordar la aproximación de  $\pi$  y que devuelva el valor aproximado.

Compruebe la exactitud de la aproximación, invocando a la función tantas veces como indique el usuario y almacenando los resultados en una lista.

Por ejemplo, si se desean 10 aproximaciones utilizando intervalos múltiplos de 10, la salida debería ser:

```
TE/prct6/src$ ./pi.py 10 10
El modo de uso es: ./pi.py [factor] [veces]
[3.1424259850010987, 3.1418009868930943, 3.1416852461797484,
 3.1416447369226574, 3.1416259869230028, 3.141615801737901,
 3.1416096603924974, 3.1416056744231198, 3.141602941655633,
 3.1416009869231254]
```

### 3. Entregable

En la tarea habilitada para esta práctica en el Aula Virtual, se subirá la dirección del repositorio *github* donde se ha almacenado la práctica.

### 4. Para saber más...

Amplíe el programa Python que ha desarrollado para que muestre por la consola los valores de la lista y su diferencia con una aproximación de 35 decimales. ¿Cómo y dónde se encuentra la primera aproximación de 3 decimales exactos?

```
TE/prct6/src$ ./pi.py 10 10
```

```
El modo de uso es: ./pi.py [factor] [veces]
```

i	PI35DT	lista i	PI35DT - lista i
1	3.1415926536	3.1424259850	0.0008333314
2	3.1415926536	3.1418009869	0.0002083333
3	3.1415926536	3.1416852462	0.0000925926
4	3.1415926536	3.1416447369	0.0000520833
5	3.1415926536	3.1416259869	0.0000333333
6	3.1415926536	3.1416158017	0.0000231481
7	3.1415926536	3.1416096604	0.0000170068
8	3.1415926536	3.1416056744	0.0000130208
9	3.1415926536	3.1416029417	0.0000102881

### Referencias

[1] Tutorial de Python. <http://docs.python.org/2/tutorial/>