

Actividad 2: Primeros programas en Python

Luisa Fernanda Orci Fernandez.

30 de Enero del 2016

Python

Python es un lenguaje de programación interpretado, esto quiere decir que puede analizar y ejecutar otros programas. La diferencia entre un programa intérprete y un compilador, es que los compiladores traducen un programa desde un lenguaje de programación a código máquina y los interpretes como Python solo realizan la traducción conforme sea necesario.

Python es administrado por Python Software Foundation, posee una licencia de código abierto y es compatible con GNU.

Actividad a realizar

En esta actividad se nos proporcionaron distintos programas en Python, los cuales hacían diferentes tipos de cálculos matemáticos, para que los modificáramos un poco y así nos familiarizáramos mas con lo que es programar en Python.

1. Problema 1: Caída libre

”Se deja caer una pelota desde el techo de una torre de altura h . Se desea saber la altura de la pelota respecto a la torre a un determinado tiempo después de haber sido dejada caer”.

El programa proporcionado para calcular los resultados fue el siguiente:

```
h = float(input("Proporciona la altura de la torre: "))
t = float(input("Ingresa el tiempo: "))
s = 0.5*9.81*t**2
print("La altura de la pelota es", h-s, "metros")
```

A partir de este programa teníamos que crear uno que calculara el tiempo en que tarda la pelota en recorrer una distancia h de la torre, despejamos el tiempo t de $h - s$ y nos quedó de la siguiente forma:

$$t = \sqrt{\frac{h}{(.5)(9.8)}}$$

Y el programa final fue el siguiente:

```
from math import sqrt
h = float(input("Proporciona la altura de la torre: "))
t = sqrt(h/(.5*9.8))
print ("El tiempo que tarda la pelota en llegar al suelo es", t, "segundos")
```

Ejemplo de los cálculos realizados por el programa:

```
Proporciona la altura de la torre: 20
('El tiempo que tarda la pelota en llegar al suelo es',
2.0203050891044216, 'segundos')
```

2. Problema 2: Un satélite orbitando alrededor de la Tierra

Un satélite orbita la Tierra a una altura h , con un periodo T en segundos. Demuestre que la altitud h del satélite sobre la superficie de la Tierra esta dado por la expresión:

$$(R + h^3) = \frac{(GMT^2)}{4\pi^2} \quad (1)$$

donde:

$G = 6.67 \times 10^{-11} \frac{m^3}{kg(s)}$ es la constante de Gravitación Universal de Newton, $M = 5.97 \times 10^{24} kg$ es la masa de la Tierra y $R = 6371000m$ es su radio".

A partir de este problema debíamos realizar un programa al cual le proporcionabamos el periodo T en segundos y este nos calcularía la altura h a la cual se encontraba el satélite. A partir de la ecuación (1), despejamos a la altura h y la ecuación resultante fue la siguiente:

$$h = \sqrt[3]{\frac{(GMT^2)}{4\pi^2}} - R \quad (2)$$

Utiliamos esta ecuación en nuestro programa para que calculara el valor de la altura, el programa quedó así:

```
from math import pi
T = float(input("Ingrse un valor para el periodo en segundos" ))
G = 6.67e-11
M = 5.97e24
```

```

R = 6371000
h = (((G*M*T*T)/(4*pi*pi))**(1./3.)) - R
print ("El valor de la altitud del satellite sobre la superficie terrestre es",
h, "metros")

```

Para probar el programa ingresamos un periodo de 90 minutos, que corresponde a 5400 segundos y el resultado fue el siguiente:

```

Ingrse un valor para el periodo en segundos5400
('El valor de la altitud del satellite sobre la superficie terrestre es',
279321.6253728606, 'metros')

```

3. Problema 3: Coordenadas esféricas

Estas coordenadas se basan en la misma idea que las polares, y son utilizadas frecuentemente para determinar la posición de un punto en el espacio, utilizando una distancia y dos ángulos. El punto P se representa por tres magnitudes: el radio r , el ángulo polar ϕ y θ .

El siguiente programa pide solicita un punto en coordenadas cartesianas y las convierte a coordenadas polares.

```

from math import sin, cos, pi, acos, atan, sqrt
x = float(input("Introduce un valor para x:"))
y = float(input("Introduce un valor para y:"))
z = float(input("Introduce un valor para z:"))

r = sqrt((x**2) + (y**2) + (z**2))
theta = acos((z/r))
phi = atan((y/x))

print("r = ", r, "theta = ", theta, "phi = ", phi)

```

Corriendo el programa:

```

Introduce un valor para x:5
Introduce un valor para y:10
Introduce un valor para z:5
('r = ', 12.24744871391589, 'theta = ', 1.1502619915109316,
'phi = ', 1.1071487177940904)

```

4. Problema 4: Números pares e impares

En este problema solo corrimos un programa que se nos proporcionó, el objetivo de esto es ver como funciona el comando *while*. El programa detecta los números pares e impares:

```
print("Enter two integers, one even, one odd.")
m = int(input("Enter the first integer: "))
n = int(input("Enter the second integer: "))
while (m+n)%2==0:
    print("One must be even and the other odd.")
    m = int(input("Enter the first integer: "))
    n = int(input("Enter the second integer: "))
print("The numbers you chose are",m,"and",n)
```

Con esto nos dimos cuenta que el comando *while* sirve para hacer iteraciones.

5. Problema 5: Números de Catalan

Este problema consistía en crear un programa que calculara los números de Catalan menores o igual a 1000000. El problema anterior fue de gran ayuda, ya que nos mostró como utilizar el *while*, también fue de gran ayuda el ejemplo de como crear un programa para calcular la sucesión de números de Fibonacci:

```
f1, f2 = 1, 1
while f2<1000:
    print(f2)
    f1, f2 = f2, f1+f2
```

también se nos proporcionó la fórmula de recurrencia de los números de Catalan:

$$C_0 = 1, C_{n+1} = \frac{2(2n+1)}{(n+2)}C_n$$

El programa quedó de la siguiente manera:

```
n, c = 0., 1.
while c<=1000000:
    print(c)
    n, c = (n+1), (2*(2*n+1)/(n+2))*c
```

Y el resultado fue:

1.0
1.0
2.0
5.0
14.0
42.0
132.0
429.0
1430.0
4862.0
16796.0
58786.0
208012.0
742900.0

Conclusiones

Me pareció que esta actividad será de gran ayuda en un futuro, ya que los programas que hicimos se ven sencillos, pero contienen comandos que van a ser de gran ayuda.

Referencias

[1] Wikipedia, <https://en.wikipedia.org>

[2] <http://computacional1.pbworks.com/w/page/104476954/Actividad>