

Universidad de Cundinamarca

Pregrado en Ingeniería de Sistemas

Taller Python - Curso: Métodos Numéricos

17 de Agosto del 2015

Nota:

- Investigué como se realizan tablas y tabulan datos en Python.
- Para consultar el módulo de matemáticas (math) en spyder, digite en la terminal `help(math)`. En el encontrará toda la información referente a las funciones matemáticas con que cuenta Python.
- Consulte el siguiente enlace: <https://docs.python.org/2/library/math.html>
- **Uso de funciones y constantes matemáticas:** Si desea utilizar una función o constante del módulo math en su código, en el encabezado del mismo debe ir la siguiente instrucción: ***from math import nombre función(s) o contante(s)***, donde las cantidades se separan con comas.
 - Ejemplo: si se desea calcular la función seno y también utilizar el valor de π , al inicio del código la instrucción debe ser: ***from math import sin, pi***

1. Implementé una función en Python que calculé las siguientes cantidades. Para cada caso determine el parámetro(s) mas conveniente (***nota:*** tenga en cuenta que son programas independientes).

- El volumen de una esfera, en este caso el parámetro de entrada debe *ser el radio*.
- El área superficial de una esfera.
- El área de un triángulo.
- El volumen de un paralelepípedo.
- El volumen de un cono.
- El área de un trapecio.
- El área de un hexágono.
- El perímetro de un hexágono.
- El área superficial de un cilindro.
- El área de un nonágono.

2. Dada una ecuación cuadrática $ax^2 + bx + c = 0$. Escriba una función en Python, con nombre *raices(a, b, c)* que retorne las dos raíces de la ecuación cuadrática, los valores de las raíces deben ser punto flotante. Utilice la función *raices* para distintos valores de a, b, c .

3. Implementé una función con parametro n en Python que calcule la sumatoria:

$$S = \sum_{i=1}^n i$$

Realice este programa utilizando una instrucción *for* y también un *while*.

4. Implementé una función con parametro n en Python que calcule la sumatoria:

$$S = \sum_{i=1}^n \frac{1}{i}$$

Realice este programa utilizando una instrucción *for* y también un *while*.

5. Implementé una función con parametro n en Python que calcule la sumatoria:

$$S = \sum_{i=1}^n \frac{1}{i^2 + 2}$$

Realice este programa utilizando una instrucción *for* y también un *while*.

6. Para convertir de grados centígrados a grados Fahrenheit se utiliza la siguiente relación:

$$F = \frac{9}{2}C + 32$$

Escriba una función que calcule los grados Fahrenheit, después obtenga una tabla en la cual la primera columna muestre el resultado en grados centígrados y la segunda su equivalente en grados Fahrenheit, la tabla debe iniciar en $C = -30^\circ\text{C}$ y finalizar en $C = 40^\circ\text{C}$.

7. Escriba un programa que imprima una tabla con el tiempo t en la primera columna y $y(t) = v_0t - 0.5gt^2$ en la segunda, utilice n valores de tiempo, uniformemente espaciados en el intervalo $[0, 2v_0/g]$. Utilice los valores $v_0 = 1$, $g = 9.81$ y $n = 11$.

8. Se quiere generar x coordenadas entre 1 y 2 con un espaciamiento de 0.01, las coordenadas están dadas por la formula $x_i = 1 + ih$ donde $h = 0.01$ y $i \in [0, 100]$. Implemente una función que calcule los valores de x_i y guárdelos en una lista, la cual esta vacía inicialmente. Llame su programa *coor1.py*, **nota:** Realice este programa utilizando una instrucción *for* y también un *while*.

9. Dadas $n + 1$ raíces r_0, r_1, \dots, r_n de un polinomio $p(x)$ de grado n , se puede calcular como:

$$p(x) = \prod_{i=0}^n (x - r_i) = (x - r_0)(x - r_1) \dots (x - r_{n-1})(x - r_n)$$

Implementé una función $p(x, n)$. Después guarde las raíces r_0, r_1, \dots, r_n en una lista, y realice un *loop* que calcule el producto, evalúe su programa para las raíces -1, 1, 2 y distintos valores de x .

10. Un triangulo arbitrario se puede describir por las coordenadas de sus tres vértices; (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , el area del triángulo esta dada por la ecuación (en función del valor absoluto $|\cdot|$):

$$A = \frac{1}{2} |x_2y_3 - x_3y_2 - x_1y_3 + x_3y_1 + x_1y_2 - x_2y_1|$$

Escriba una función *area(vertices)* que retorne el área de un triángulo para un conjunto de vertices.

11. El factorial de n se puede escribir como:

$$n! = n(n-1)(n-2)\dots 2.1$$

con los casos especiales $1! = 1$ y $0! = 1$, implemente una función *fact(n)* que retorne $n!$. Realice este programa utilizando una instrucción *for* y también un *while*.

12. Dado un número x y un entero no negativo n , realice una función que calcule x^n , utilice un *for* dentro del cuerpo de la función, verifique su resultado comparando con el resultado producido por $x * n$. Realice este programa también utilizando un *while*.

13. Implementé una función que convierta de coordenadas cartesianas a polares.

14. Implementé una función que convierta de coordenadas cartesianas a esféricas.

15. La función $\cos(x)$ se puede aproximar por la suma:

$$C(x; n) = \sum_{j=0}^n c_j$$

donde $c_j = -c_{j-1} \frac{x^2}{2j(2j-1)}$, $j = 1, 2, \dots, n$ y $c_0 = 1$. Implementé una función en Python para $C(x; n)$, también realice una tabla donde se calculé $C(x; n)$ y $\cos(x)$ para algunos valores de x y n .

16. Implementé una función que calculé el volumen de un tetraedro en función de las coordenadas de los vertices (x, y, z) , suponga que uno de los vertices coincide con el origen $(0, 0, 0)$.

Nota: Datos interesantes sobre el tetraedro y sus propiedades geométricas en: <https://en.wikipedia.org/wiki/Tetrahedron>