

# Taller 1

Métodos Computacionales para Políticas Públicas - URosario

Entrega: viernes 12-feb-2021 11:59 PM

\*\*[Juan Diego Castro Rodriguez]\*\*

[juand.castro@urosario.edu.co]

## Instrucciones:

- Guarde una copia de este *Jupyter Notebook* en su computador, idealmente en una carpeta destinada al material del curso. Sugiero una estructura similar a la del repositorio del curso.
- Modifique el nombre del archivo del *notebook*, agregando al final un guión inferior y su nombre y apellido, separados estos últimos por otro guión inferior. Por ejemplo, mi *notebook* se llamaría: mcpp\_taller1\_santiago\_matalana
- Marque el *notebook* con su nombre y e-mail en el bloque verde arriba. Reemplace el texto "[Su nombre acá]" con su nombre y apellido. Similar para su e-mail.
- Desarrolle la totalidad del taller sobre este *notebook*, insertando las celdas que sea necesario debajo de cada pregunta. Haga buen uso de las celdas para código y de las celdas tipo *markdown* según el caso.
- Recuerde salvar periódicamente sus avances.
- Cuando termine el taller:
  1. Descárguelo en PDF. Esto puede implicar instalar LaTeX en su computador. Resuélvalo por su cuenta, por favor. Recuerde: Google es su amigo.
  2. Suba los dos archivos (.pdf y .ipynb) a su repositorio en GitHub antes de la fecha y hora límites. Asegúrese de que Daniel sea "colaborador" de su repositorio y de que los dos archivos queden en su repositorio, en la nube (no solo en su computador). No lo deje para última hora. Talleres subidos después de la fecha y hora límites no serán valorados, como tampoco lo serán si son remitidos vía e-mail.

(Todos los ejercicios tienen el mismo valor.)

## 1. Zelle, sección 1.10 (p. 17):

- "Multiple Choice", Ejercicios # 1-10. 1:b 2:d 3:d 4:a 5:a 6:b 7:b 8:c 9:a 10:c
- "Programming Exercises", Ejercicio # 1.

In [5]:

print ("Hello, world!")

Hello, world!

In [6]:

print ("Hello", "world!")

Hello world!

In [7]:

print(3)

3

In [8]:

print (3.0)

3.0

In [9]:

print (2+3)

5

In [10]:

print(2.0+3.0)

5.0

In [11]:

print ("2"+"3")

23

In [12]:

print("2+3=", 2+3)

2+3= 5

In [13]:

print(2\*3)

6

In [14]:

print(2\*\*3)

8

In [15]:

print(2/3)

0.6666666666666666

En *computer science* son comunes los ejercicios denominados "pensar como un computador". Con estos usted evalúa si está comprendiendo el material, siempre y cuando no utilice un computador para correr el código del enunciado. Siempre que vea un ejercicio marcado con la etiqueta "pensar como un computador", use papel y lápiz o incluso una calculadora si es necesario para descifrar la respuesta, pero nunca ejecute el código en computador.

## 2. [Pensar como un computador] ¿Cuál es el valor de w después de ejecutar el siguiente código?

x= 7 y= 5.0 z= 10.0 w= x % 2 + y / z + z + y / (z + z) W= 7 % 2 + 5.0/10.0+ 5.0/ (10.0+10.0) w=7 % 2 + 5.0/10.0+ 5.0/20 w= 0.5+0.5+0.5 w=1.5

## 3. [Pensar como un computador] ¿Cuál es el valor de c después de ejecutar el siguiente código?

c = True d = False c = c and d c = not c or d c= "not c or d"

## 4. Ejecute el siguiente código y responda: ¿Por qué es falsa la tercera línea, mientras que las primeras dos son verdaderas?

1 == 1 "1" == "1" 1 == "1" porque en no es lo mismo tener al 1 como número a tener al 1 como texto.

## 5. Escriba un programa que le pida al usuario ingresar su nombre y que arroje un texto saludando de vuelta al usuario, así: "Hola, <nombre>. ¡Veo que aprendes Python rápidamente! ¡Felicitaciones!".

In [4]:

x = input ("Escriba su nombre:")  
print ("Hola, "+x+". ¡Veo que aprendes Python rápidamente! ¡Felicitaciones!")

Escriba su nombre:Juan Diego  
Hola, Juan Diego. ¡Veo que aprendes Python rápidamente! ¡Felicitaciones!