

# mcpp\_taller1\_juan\_salgado

August 7, 2019

## 1 Taller 1

Métodos Computacionales para Políticas Públicas - URosario

Entrega: viernes 9-ago-2019 11:59 PM

- Juan Camilo Salgado Ramirez
- juanca.salgado@urosario.edu.co

### 1.1 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\_mataallana
- 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:
  - Descárguelo en PDF. Esto puede implicar instalar LaTeX en su computador. Resuélvalo por su cuenta, por favor. Recuerde: Google es su amigo.
  - 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.)

---

### 1.2 1. Zelle, sección 1.10 (p. 17):

- “Multiple Choice”, Ejercicios # 1-10.
- “Programming Exercises”, Ejercicio # 1.

### 1.2.1 Respuestas “Multiple Choice”

1 - b  
2 - d  
3 - d  
4 - a  
5 - b  
6 - b  
7 - b  
8 - b  
9 - a  
10 - d

### 1.2.2 Respuestas “Programming Exercises”

```
In [1]: print("Hello, world!")
```

Hello, world!

```
In [2]: print("Hello", "world!")
```

Hello world!

```
In [3]: print(3)
```

3

```
In [4]: print(3.0)
```

3.0

```
In [5]: print(2 + 3)
```

5

```
In [6]: print(2.0 + 3.0)
```

5.0

```
In [7]: print("2" + "3")
```

23

```
In [8]: print("2 + 3 =", 2 + 3)
```

$2 + 3 = 5$

```
In [9]: print(2 * 3)
```

6

```
In [10]: print(2 ** 3)
```

8

```
In [11]: 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.

### 1.3 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)
```

#### 1.3.1 Respuesta

```
w = 1 + 0,5 + 10 + 5/20
w = 11,5 + 0.25
w = 11,75
w valdrá 11,75
```

### 1.4 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
```

### 1.4.1 Respuesta

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

### 1.5 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"
```

### 1.5.1 Respuesta

- La primera línea es **verdadera** porque se está comparando el valor de dos *int* iguales: 1 y 1
  - La segunda línea es **verdadera** porque se está comparando el valor de dos *strings* iguales: "1" y "1"
  - La tercera línea es **falsa** porque se está comparando el valor del *número* 1 con el *\*string* "1": 1 != "1", porque son dos tipos de datos distintos.
- 

### 1.6 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 [1]: nombre = input("¿Cuál es su nombre? \n")
        print(f"Hola {nombre}. ¡Veo que aprendes Python rápidamente! ¡Felicitaciones!")

¿Cuál es su nombre?
jc
Hola jc. ¡Veo que aprendes Python rápidamente! ¡Felicitaciones!
```

---