

2021-I_mcpp_taller_1_Valentina_Cuenca

February 11, 2021

1 Taller 1

Métodos Computacionales para Políticas Públicas - URosario

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

Valentina Cuenca norma.cuenca@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_matallana
- 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.)

1.2 1. Zelle, sección 1.10 (p. 17):

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

- Multiple choice:

1. B
2. D
3. D
4. C
5. A
6. B
7. B
8. B
9. A
10. D

- Programming Exercises:

```
[22]: print("Hello, world!")
print("Hello", "world!")
print(3)
print(3.0)
print(2 + 3)
print(2.0 + 3.0)
print("2" + "3")
print("2 + 3 =", 2 + 3)
print(2 * 3)
print(2 ** 3)
print(2 / 3)
```

```
Hello, world!
Hello world!
3
3.0
5
5.0
23
2 + 3 = 5
6
8
0.6666666666666666
```

- a) Hello, wordl!
- b) Hello world!
- c) 3
- d) 3.0
- e) 5
- f) 5.0
- g) 23
- h) $2 + 3 = 5$

- i) 6
- j) 8
- k) 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

Respuesta: 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

Respuesta: Verdadero

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"

Porque la ultima linea esta igualando un string con un numeric, lo que no es posible, mientras que antes estaba igualando semejantes.

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!”.

```
[23]: Nombre= input ("¿Cómo es tu nombre? ")
print("Hola," , Nombre,"¡Veo que aprendes Python rápidamente! ¡Felicitaciones!" )
```

¿Cómo es tu nombre? Valentina
Hola, Valentina ¡Veo que aprendes Python rápidamente! ¡Felicitaciones!

[]: