

Taller 4

Métodos Computacionales para Políticas Públicas - URosario

Entrega: viernes 28-feb-2020 11:59 PM

****[Andrés Ramírez Vela]****

[andrese.ramirez@urosario.edu.co]

Instrucciones:

- Guarde una copia de este *Jupyter Notebook* en su computador, idealmente en una carpeta destinada al material 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_taller4_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.
 2. Suba los dos archivos (.pdf y .ipynb) a su repositorio en GitHub antes de la fecha y hora límites.

(Todos los ejercicios tienen el mismo valor.)

Zelle, Exercises 6.8 (p. 159):

- True/False: 1-10
- Multiple choice: 2, 3, 6, 7, 10
- Programming Exercises: 1, 3, 4, 11, 12, 13

True/False 1. Falso 2. Falso 3. Verdadero 4. Verdadero 5. Falso 6. Falso 7. Falso 8. Verdadero 9. Verdadero 10. Falso
Selección Múltiple 2. A 3. A 6. A 7. D 10. A
Ejercicios de Programación

In [5]: *#Ejercicio 1*

```
def cancion():
    print("Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!")

def animal(animal, sonido):
    print("And on this farm he had a", animal + ", Ee-igh, Ee-igh, Oh!")
    print("With a", sonido + ", ", sonido, "here, and a", sonido + ", ", sonido, "there.")
    print("Here a", sonido + ", ", "there a", sonido + ", ", "everywhere a", sonido + ", ", sonido + ".")

def main():
    oldMac()
    animal('dog', 'guau')
    oldMac()
    print()
    oldMac()
    animal('wolf', 'auu')
    oldMac()
    print()
    oldMac()
    animal('tiger', 'groaarr')
    oldMac()
    print()
    oldMac()
    animal('sheep', 'baah')
    oldMac()
    print()
    oldMac()
    animal('cat', 'miau')
    oldMac()

main()
```

Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!
 And on this farm he had a dog, Ee-igh, Ee-igh, Oh!
 With a guau, guau here, and a guau, guau there.
 Here a guau, there a guau, everywhere a guau, guau.
 Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!

Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!
 And on this farm he had a wolf, Ee-igh, Ee-igh, Oh!
 With a auu, auu here, and a auu, auu there.
 Here a auu, there a auu, everywhere a auu, auu.
 Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!

Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!
 And on this farm he had a tiger, Ee-igh, Ee-igh, Oh!
 With a groaarr, groaarr here, and a groaarr, groaarr there.
 Here a groaarr, there a groaarr, everywhere a groaarr, groaarr.
 Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!

Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!
 And on this farm he had a sheep, Ee-igh, Ee-igh, Oh!
 With a baah, baah here, and a baah, baah there.
 Here a baah, there a baah, everywhere a baah, baah.
 Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!

Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!
 And on this farm he had a cat, Ee-igh, Ee-igh, Oh!
 With a miau, miau here, and a miau, miau there.
 Here a miau, there a miau, everywhere a miau, miau.
 Old MacDonald had a farm, Ee-igh, Ei-igh, Oh!

In [9]: *# Ejercicio 3*

```
import math as m

def sphereArea(radius):
    area = 4 * m.pi * (radius*radius)
    return area
def sphereVolume(radius):
    volume = 4/3 * m.pi * (radius ** 3)
    return volume

def main():
    r = (eval(input("Introduzca el valor del radio: ")))
    a = sphereArea(r)
    v = sphereVolume(r)
    print("Una esfera con radio de", r, "tiene un volumen de", round(v,
3), "y una superficie de área de", round(a, 3),".")

main()
```

Introduzca el valor del radio: 20
 Una esfera con radio de 20 tiene un volumen de 33510.322 y una superficie de área de 5026.548 .

In [20]: *#Ejercicio 4*

```
def sumN(n):
    suma = 1
    for i in range(n,1,-1):
        suma = suma + i
    return suma
def sumNCubes(n):
    cubo = 1
    for x in range(n,1,-1):
        cubo = cubo + x ** 3
    return cubo
def main():
    n = eval(input("Ingrese un número natural: "))
    N = sumN(n)
    NCubes = sumNCubes(n)
    print("La suma de los primeros {0} números naturales es {1}, y la suma
    "
    "de los primeros {0} cubos es {2}".format(n, N, NCubes))

main()
```

Ingrese un número natural: 10

La suma de los primeros 10 números naturales es 55, y la suma de los primeros 10 cubos es 3025

In [26]: *# Ejercicio 11*

```
def squareEach(nums):
    for x in range(len(nums)):
        nums[x] = nums[x] ** 2
    return(nums)
def test():
    list = [1,2,3,4,5,6,7,8,9]
    y = squareEach(list)
    print(y)
test()
```

[1, 4, 9, 16, 25, 36, 49, 64, 81]

In [27]: *# Ejercicio 12*

```
def sumList(nums):
    y = 0
    for x in range(len(nums)):
        y = nums[x] + y
    return(y)
def test():
    list = [1,2,3,4,5,7,8,9,10,11,12,13,14,15]
    z = sumList(list)
    print(z)
test()
```

114

```
In [30]: # Ejercicio 13

def toNumbers(strList):
    for x in range (len(strList)):
        strList[x] = int(strList[x])
    return(strList)
def test():
    list = ["1", "2", "3", "4", "5", "6", "7", "8", "9", "10"]
    z = toNumbers(list)
    print(z)
test()

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```
