# mcpp_taller4_Daniela_Gaitan_C

August 28, 2016

## 1 Taller 4

Métodos Computacionales para Políticas Públicas - URosario
**Entrega: viernes 2-sep-2016 11:59 PM**
**[Daniela Gaitán Cotrino]** [daniela.gaitanc@urosario.edu.co]

## 2 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.)

---

## 3 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

## 3.1 True / False

**1. Programmers rarely define their own funtions**

- False

**2. A funtion may only be called at one place in a program**

- False

**3. Information can be passed into a funtion through parameters**

- True

**4. Every Python funtion returns some value**

- True

**5. In Python, some parameters are passed by reference**

- False

**6. In Python, a funtion can return only one value**

- False

**7. Python funtions can never modify a parameter**

- False

**8. One reason to use funtions is to reduce code duplication**

- True

**9. Variables defined in a funtion are local to that funtion**

- True

**10. It's a bad idea to define new funtions if it makes a program longer**

- False

## 3.2 Multiple choice

**2. A Python function definition begins with**

- (a) def

**3. A function can send output back to the program with a(n)**

- (a) return

**6. In Python, actual parameters are passed to functions**

- (a) by value

**7. Which of the following is not a reason to use functions?**

- (d) to demonstrate intellectual superiority

**10. A function can modify the value of an actual parameter only if it's**

- (a) mutable

## 3.3 Programming Exercises

**1. Write a program to print the lyrics of the song "Old MacDonald." Your program should print the lyrics for five different animals, similar to the example verse below.**

```
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!

And on that farm he had a cow, Ee-igh, Ee-igh, Oh!

With a moo, moo here and a moo, moo there.

Here a moo, there a moo, everywhere a moo, moo.

Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
```

```
In [1]: def animals(animal, sound):
            print ("Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!")
            print ("And on the farm he had a ", animal + ", Ee-igh, Ee-igh,Oh!")
            print ("With a", sound + ",", sound, "here and a" , sound + ",",sound +
            print ("Here a", sound + ", there a", sound + ", everyehere a",sound  +
            print ("Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!")
            print
            return animals

        def oldMac():
            animals("cow", "moo")
            animals("pig", "oink")
            animals("horse", "neigh")
            animals("dog", "woof")
            animals("cat", "meow")

        oldMac()
```

```
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
And on the farm he had a  cow, Ee-igh, Ee-igh,Oh!
With a moo, moo here and a moo, moo there.
Here a moo, there a moo, everyehere a moo, moo.
```

```
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
And on the farm he had a  pig, Ee-igh, Ee-igh,Oh!
With a oink, oink here and a oink, oink there.
Here a oink, there a oink, everyehere a oink, oink.
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
And on the farm he had a  horse, Ee-igh, Ee-igh,Oh!
With a neigh, neigh here and a neigh, neigh there.
Here a neigh, there a neigh, everyehere a neigh, neigh.
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
And on the farm he had a  dog, Ee-igh, Ee-igh,Oh!
With a woof, woof here and a woof, woof there.
Here a woof, there a woof, everyehere a woof, woof.
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
And on the farm he had a  cat, Ee-igh, Ee-igh,Oh!
With a meow, meow here and a meow, meow there.
Here a meow, there a meow, everyehere a meow, meow.
Old MacDonald had a farm, Ee-igh, Ee-igh, Oh!
```

**3. Write definitions for these functions:**

sphereArea(radius) Returns the surface area of a sphere having the given radius.

sphereVolume(radius) Returns the volume of a sphere having the given radius.

- Use your functions to solve Programming Exercise 1 from Chapter 3.

In [13]:
```
import math

def Area_esfera(radio):
    area = math.pi*4*(radio**2)
    return area

def Volumen_esfera(radio):
    volume = (4/3)*math.pi*(radio**3)
    return volume

def main():
    print('Programa para calcular área y volume de una esfera.')
    radiusEnter = eval(input('Por favor ingrese el radio de la esfera:'))
    area = Area_esfera(radiusEnter)
    vol = Volumen_esfera(radiusEnter)
    print('\nCon un radio de {}, el área de la esfera es {:.2f} y el volum

main()
```

```
Programa para calcular área y volume de una esfera.
Por favor ingrese el radio de la esfera:84

Con un radio de 84, el área de la esfera es 88668.31 y el volumen es 2482712.71.
```

**4. Write definitions for the following two functions:**

```
sumN(n) returns the sum of the first n natural numbers.

sumNCubes(n) returns the sum of the cubes of the first n natural numbers.

- Then use these functions in a program that prompts a user for n and prints out th
```

```
In [3]: def sumN(n):
            result = 0
            for i in range(n+1):
                result += i
            return result

        def sumNCubes(n):
            result = 0
            for i in range(n+1):
                result += i**2
            return result

        def prompt_user():
            n = int(input("por favor ingrese n:"))
            print("la suma de los primeros n números es:", sumN(n))
            print("la suma de los primeros n cubos es:", sumNCubes(n))

        prompt_user()
```

```
por favor ingrese n:7
la suma de los primeros n números es: 28
la suma de los primeros n cubos es: 140
```

**11. Write and test a function to meet this specification.** squareEach(nums) nums is a list of numbers. Modifies the list by squaring each entry.

```
In [4]: def squareEach(nums):
            entry = 0
            for i in nums:
                nums[entry] = i**2
                entry = entry+1

        def main():
```

```
        print('Programa para elevar números al cuadrado automaticamente.')
        nums = input('Por favor ingrese varios números separandolos por comas:

        nums = nums.split(',')

        entry = 0
        for i in nums:
            nums[entry] = int(i)
            entry = entry+1

        squareEach(nums)

        print('\nEl resultado de elevar al cuadrado es: ',nums)
    main()
```

```
Programa para elevar números al cuadrado automaticamente.
Por favor ingrese varios números separandolos por comas: 89,998

El resultado de elevar al cuadrado es:  [7921, 996004]
```

**12. Write and test a function to meet this specification.** sumList(nums) nums is a list of numbers. Returns the sum of the numbers in the list.

```
In [5]: def sumList(nums):
            total = 0
            for i in nums:
                total = total+i
            return total

        def main():
            print('Programa para sumar todo los números ingresados.')
            nums = input('Por favor ingrese varios números separandolos por comas:

            nums = nums.split(',')

            entry = 0
            for i in nums:
                nums[entry] = int(i)
                entry = entry+1

            sumTotal = sumList(nums)

            print('\nEl resultado de la suma de los números ingresados es {}.'.form
        main()
```

```
Programa para sumar todo los números ingresados.
Por favor ingrese varios números separandolos por comas: 3,77,99,888
```

```
El resultado de la suma de los números ingresados es 1067.
```

**13. Write and test a function to meet this specification.** toNumbers(strList) strList is a list of strings, each of which represents a number. Modi- fies each entry in the list by converting it to a number.

```
In [4]: def toNumbers(strList):
            return [int(num) for num in strList]

        a=['1','2','3']

        b=toNumbers(a)
        b

Out[4]: [1, 2, 3]
```

---