
bucles_listas

Informática 2015-2016

November 21, 2015

Part I

Bucles y Listas

Un poco de motivación

Queremos hacer la media de unos números reales, cada uno en una variable. ¿qué ocurre si no se sabe cuántos números hay?

```
numberlst = [4.0, 6.0, 7.0, 3.0, 2.0]
```

In [1]:

```
s = 0.0
```

In [2]:

```
s = s + numberlst[0]
s = s + numberlst[1]
s = s + numberlst[2]
s = s + numberlst[3]
s = s + numberlst[4]
mean = s / len(numberlst)
mean
4.4
```

Out [2]:

Eso no es generalizable. Necesitamos un bucle

```
In [3]: def mean(numberlst):
        """This function returns the mean of a list of numbers

        Parameters
        -----
        numberlst : [float]
            List of float numbers

        Returns
        -----
        float
            Mean of the list

        Example
```

The logo for Cartagena99 features the text 'Cartagena99' in a stylized, blue, serif font. The '99' is significantly larger and more prominent than the 'Cartagena' part. The text is set against a light blue background with a subtle gradient and a soft shadow effect.

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70

```
return s / len(numberlst)
mean(numberlst)
```

In [4]: 4.4

Out [4]:
Queremos hacer funciones con polinomios

poly —> 3x³ - 6x + 2

Como representamos este polinomio, ¿con 4 reales? ¿y si el polinomio tiene grado n? Necesitamos algo que nos permita representar n reales: 3, 0, -6, 2

```
poly = [2.0, -6.0, 0, 3.0]
poly[0], poly[1], poly[2], poly[3]
(2.0, -6.0, 0, 3.0)
```

Out [5]:

Trabajar con listas

La longitud de una lista

```
len(poly)
```

In [6]: 4

Out [6]:
Las posiciones de la lista poly van desde 0 hasta len(poly). Ojo a los accesos ilegales

```
poly[4]
```

In [7]:

```
-----
IndexError                                Traceback (most recent
call last)

<ipython-input-7-a085d43b506f> in <module>()
----> 1 poly[4]
```

IndexError: list index out of range

Se puede cambiar el contenido de una posición de una lista

```
poly[3] = -7.0
poly
[2.0, -6.0, 0, -7.0]
```

```
Out [8]: poly[4] = 9.0
```



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70

```
<ipython-input-9-d742d21044b4> in <module>()
----> 1 poly[4] = 9.0
```

IndexError: list assignment index out of range

Los elementos de una lista se pueden usar en cualquier contexto

```
In [10]: a = poly[2] + 1 # expresión
abs(poly[3]) #llamada a función
7.0
```

Out [10]:
Se pueden añadir más elementos a una lista

```
In [11]: poly.append(9.0)
poly
[2.0, -6.0, 0, -7.0, 9.0]
```

Out [11]:

Ejemplos de bucles con listas

Vamos a hacer una función que evalúe el polinomio en el punto x

```
In [12]: x = 2
result = 0.0
power = 1
result = result + power * poly[0]
power = power * x
result = result + power * poly[1]
power = power * x
result = result + power * poly[2]
power = power * x
result = result + power * poly[3]
result
-66.0
```

Out [12]:
¿y si el polinomio tiene grado n?

```
In [18]: x = 2
result = 0.0
power = 1
degree = 4
i = 0
while i <= degree:
    result = result + power * poly[i]
    power = power * x
    i = i + 1
result
78.0
```

Out [18]: `def eval_poly(poly, x):`

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70

The logo for Cartagena99 features the text 'Cartagena99' in a stylized, blue, serif font. The '99' is significantly larger and more prominent than the 'Cartagena' part. The text is set against a light blue background with a subtle gradient and a soft shadow effect.

```

    Point

Returns
-----
float
    Value of the polynomial at point x

Example
-----
>>> eval_poly( [1.0, 1.0], 2)
3.0
"""
result = 0.0
power = 1
degree = len(poly) - 1
i = 0
while i <= degree:
    result = result + poly[i] * power
    power = power * x
    i = i + 1
return result
eval_poly(poly, 2.0), eval_poly( [1.0, 1.0], 2)

```

In [14]: (78.0, 3.0)

Out [14]:

Ejercicio propuesto: Descomposición en factores primos de un número.

Ejemplos:

60 = 2, 2, 3, 5

15 = 3, 5

Seguimos el algoritmo 'clásico' de descomposición, se comienza dividiendo el número original entre el divisor más pequeño posible, se acumula el dividendo y se continúa con el divisor.

60|2 30|2 15|3 5|5 1

```

In [15]: def factors(n):
    """
    This function computes the list of factors of n

    Parameters
    -----
    n : int
        Integer number to decompose, n > 1

    Returns
    -----
    [int]
        Factors of n

    Example
    -----
    >>> factors(256)
    [2, 2, 2, 2, 2, 2, 2, 2]
    """
    fct = 2 # 2 is the first prime number

```



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70

```
factors(392000), factors(256)
```

```
In [16]: ([2, 2, 2, 2, 2, 2, 5, 5, 5, 7, 7], [2, 2, 2, 2, 2, 2, 2, 2])
```

Out [16]:

Si queremos calcular la multiplicidad podemos hacer que devuelva una lista de tuplas

```
In [17]: def factors_exp(n):
    """
    This function returns the list of factors of n,
    together with their exponents. The function returns
    a list of tuples, the first element of the tuple is the factor and the
    second one is the exponent.

    Parameters
    -----
    n : int
        Integer number to decompose, n > 1

    Returns
    -----
    [(int, int)]
        Factors (factor, exponene) of n

    Example
    -----
    >>> factors_exp(256)
    [(2, 8)]
    """
    fct = 2
    factors = []
    while n > 1:
        if n % fct == 0: #if it is a divisor, we compute its multiplicity.
            exp = 1
            n = n // fct
            while (n % fct) == 0:
                exp += 1
                n = n // fct
            factors.append((fct,exp))
            # at this point n % fct != 0, so we increase fct in any case.
            fct += 1
    return factors
```

```
factors_exp(60), factors_exp(256)
```

```
In [18]: ((2, 2), (3, 1), (5, 1)), [(2, 8)]
```

Out [18]:

In []:

The logo for Cartagena99 features the text 'Cartagena99' in a stylized, blue, serif font. The '99' is significantly larger and more prominent than the word 'Cartagena'. The text is set against a light blue background with a subtle gradient and a soft shadow effect.

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70