

# Python In a Nutshell

## Why another programming language?

**All Programming Languages are created equal ;-)**

Bohm & Jacopini demonstrate that any programming language that allows you to write

- sequence
- conditional structure
- loops

can compute any computable function

[C. Bohm, e G. Jacopini,  
Flow Diagrams, Turing Machines and Languages with Only Two Formation Rules, in Communications of the ACM, 1966]

But... sometimes we need the 'easy' way

**"Economy of effort. Never stand up when you can sit down, and never sit down when you can lie down"**

[Churchill, 1946]

From this point of view, 'some programming languages are more equal than others' [Orwell, 1945]

Aim of this talk:

- give you a 'taste' of Python

show that

- some tasks are very easy in python (compared to other programming languages)
- python can be useful and handy to carry out Pattern Recognition tasks.

# Development environment:

IDE (**Pycharm**, Eclipse, **Spyder**, Wing Ide,...) (the usual way)

or

Jupyter notebook: an interactive environment

**Jupyter notebook** allows you to integrate in a single instrument

- Comments LIKE THIS
- Text with a *markup language*
- Code
- Output
- ... and latex formulas:

`a= \int_0^{+\infty} f(x) dx`

$$a = \int_0^{+\infty} f(x) dx$$

# Variables

There is no need to declare the variables.

The type is associate to the VALUE, not to the variable (dynamically typed).

It depends on the value that we assign.

```
a=1          # integer  
print (a, ': ', type(a))
```

Try this in a new cell:

```
b=1.0  
c='Newton'  
z=1+2*1j  
list_1=[10, 20, 30, 'a string!', 50]
```

Try this in a new cell:

```
list1.append(100)
print ('is 20 in the list? ', 20 in list_1)
print ('is 200 in the list? ', 200 in list_1)
```



# LOOPS

We can compare a FOR loop in C vs a FOR loop in python

Problem: print the numbers {10,15,20,7,14}

The C approach use an index to scan the array and print all the values.

BUT the index itself it is not necessary for the 'logic' of the problem.

- DEFINE THE ARRAY
- DEFINE A COUNTER
- SET THE COUNTER TO 0
- INCREMENT THE COUNTER
- CHECK THE COUNTER

## C CODE

```
int main(){
    int lunghezza=5;
    int int_array[]={10,15,20,7,14};
    int i;
    for (i=0; i<lunghezza; i++){
        printf ("%d\n", int_array[i]);
    }
    return 0;
}
```

Try this in python:

- create a list (i.e. `list_1`)
- iterate using `for el in list_1:`

# Numpy: Matlab without buying Matlab.

<http://docs.scipy.org/doc/numpy/>

```
# The package NUMPY allows us to use a Matlab-like environment without buying Matlab.
```

```
import numpy as np
```

```
# tells to the Interpreter to use the package NUMPY,  
# and to use the alias np for reason of brevity.
```

# ARRAY

Create an array:

- list of numbers -> a 'vector'
- list of lists -> an array

Try this!

---

Create an array using Python functions

```
```python
np.arange(n)
np.zeros(r,c)
np.random.randint(min_v,max_v,[r,c])
```

Try this!

Create an array like this:

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31

**hint:**

use `x=np.arange` to create values, and `x.reshape` to transform the 1-d array into a 2-d array.

Try this!

## Select rows (or columns)

```
y=np.array([2,3,1,3])
```

## Select the rows of x based on the values of y

```
x[condition, :]
```

## Try this!