# Distributed Computing and Introduction to High Performance Computing

Ahmed Ratnani[1]

[1]Mohammed VI Polytechnic University, Benguerir, Morocco

MOHAMMED VI
POLYTECHNIC
UNIVERSITY

# Outline of this lecture

- About Python
- Python is slow
- Some Benchmarks
- Outline of this lecture

# About Python

- Python was created by Guido van Rossum in 1991 (last version 3.9 - 05/10/2020)
- Python is **simple**
- Python is **fully featured**
- Python is **readable**
- Python is **extensible**
- Python is **ubiquitous, portable, and free**
- Python has **many third party libraries, tools, and a large community**

# About Python

- Python was created by Guido van Rossum in 1991 (last version 3.9 - 05/10/2020)
- Python is **simple**
- Python is **fully featured**
- Python is **readable**
- Python is **extensible**
- Python is **ubiquitous, portable, and free**
- Python has **many third party libraries, tools, and a large community**
➠ **But Python is slow!!**

# About Python

- Python was created by Guido van Rossum in 1991 (last version 3.9 - 05/10/2020)
- Python is **simple**
- Python is **fully featured**
- Python is **readable**
- Python is **extensible**
- Python is **ubiquitous, portable, and free**
- Python has **many third party libraries, tools, and a large community**
➡ **Does is really matters?**

# Python is slow
When does it matter?

- Is my code fast enough to produce the results I need in the time I have?
- How many CPUh is this code going to waste over its lifetime?
  - How inefficient is it?
  - How long does it run?
  - How often will it run?
- Does it cause problems on the system it's running on?
- How much effort is it to make it run faster?
➡ For those who are interested, you can follow this MOOC

## Some Benchmarks

**Rosen-Der**

| Tool | Python | Cython | Numba | Pythran | Pyccel-gcc | Pyccel-intel |
|------|--------|--------|-------|---------|------------|--------------|
| Timing ($\mu s$) | 229.85 | 2.06 | 4.73 | 2.07 | **0.98** | **0.64** |
| Speedup | — | $\times$ 111.43 | $\times$ 48.57 | $\times$ 110.98 | $\times$ **232.94** | $\times$ **353.94** |

**Black-Scholes**

| Tool | Python | Cython | Numba | Pythran | Pyccel-gcc | Pyccel-intel |
|------|--------|--------|-------|---------|------------|--------------|
| Timing ($\mu s$) | 180.44 | 309.67 | 3.0 | 1.1 | **1.04** | $6.56 \ 10^{-2}$ |
| Speedup | — | $\times$ 0.58 | $\times$ 60.06 | $\times$ 163.8 | $\times$ **172.35** | $\times$ **2748.71** |

**Laplace**

| Tool | Python | Cython | Numba | Pythran | Pyccel-gcc | Pyccel-intel |
|------|--------|--------|-------|---------|------------|--------------|
| Timing ($\mu s$) | 57.71 | 7.98 | $6.46 \ 10^{-2}$ | $6.28 \ 10^{-2}$ | $8.02 \ 10^{-2}$ | $2.81 \ 10^{-2}$ |
| Speedup | — | $\times$ 7.22 | $\times$ 892.02 | $\times$ 918.56 | $\times$ **719.32** | $\times$ **2048.65** |

**Growcut**

| Tool | Python | Cython | Numba | Pythran | Pyccel-gcc | Pyccel-intel |
|------|--------|--------|-------|---------|------------|--------------|
| Timing ($s$) | 54.39 | $1.02 \ 10^{-1}$ | $4.67 \ 10^{-1}$ | $8.57 \ 10^{-2}$ | $6.27 \ 10^{-2}$ | $6.54 \ 10^{-2}$ |
| Speedup | — | $\times$ 532.37 | $\times$ 116.45 | $\times$ 634.32 | $\times$ **866.49** | $\times$ **831.7** |

# Outline of this lecture

- Profiling a Python code

- Accelerate a Python code
    - Using Numpy
    - Using Cython
    - Using Numba
    - Using Pyccel