



PyQB

Monga

NumPy

ndarray

Creation

Programming in Python¹

Mattia Monga

Dip. di Informatica
Università degli Studi di Milano, Italia

`mattia.monga@unimi.it`

Academic year 2025/26, I semester



PyQB

Monga

NumPy

ndarray

Creation

Lecture XIII: NumPy arrays



NumPy is a third-party library very popular for scientific/numerical programming (<https://numpy.org/>).

- Features familiar to `matlab`, R, Julia programmers
- The key data structure is the `array`
 - 1-dimension arrays: `vectors`
 - 2-dimension arrays: `matrices`
 - n-dimension arrays

In some languages array is more or less synonym of list: Python distinguishes: `lists` (mutable, arbitrary elements), `arrays` (mutable, all elements have the same type), `tuples` (immutable, fixed length, arbitrary elements).



NumPy arrays

The most important data structure in NumPy is `ndarray`: a (usually fixed-size) sequence of same type elements, organized in one or more dimensions.

<https://numpy.org/doc/stable/reference/arrays.ndarray.html>

Implementation is based on byte arrays: accessing an element (all of the same byte-size) is virtually just the computation of an 'address'.

PyQB

Monga

NumPy

`ndarray`

Creation

Why?



PyQB

Monga

NumPy

ndarray

Creation

- using NumPy arrays is often more compact, especially when there's more than one dimension
- faster than lists when the operation can be **vectorized**
- (slower than lists when you append elements to the end)
- can be used with element of different types but this is less efficient

