PHAS0102: Techniques of High-Performance Computing

Moodle

- You should now all have access to Moodle.
  - If not, email me

Tutorials

- (Group 1) Mondays 10-11, Euston Road (222) G01
  - Very full
- (Group 2) Mondays 11-12, Chadwick Building 2.18
  - 9 spaces

Virtual drop-in hour

- Wednesdays 11:30-12:30, link on Moodle
- **No drop-in on 12 October**

PHAS0102 Part 0: What is High-Performance Computing?

HPC Programming languages

Compiled languages

C / C++

Rust

Fortran

Go, Java (usually), COBOL, Haskell, Pascal...

Interpreted languages

Python

Matlab

Ruby

Julia

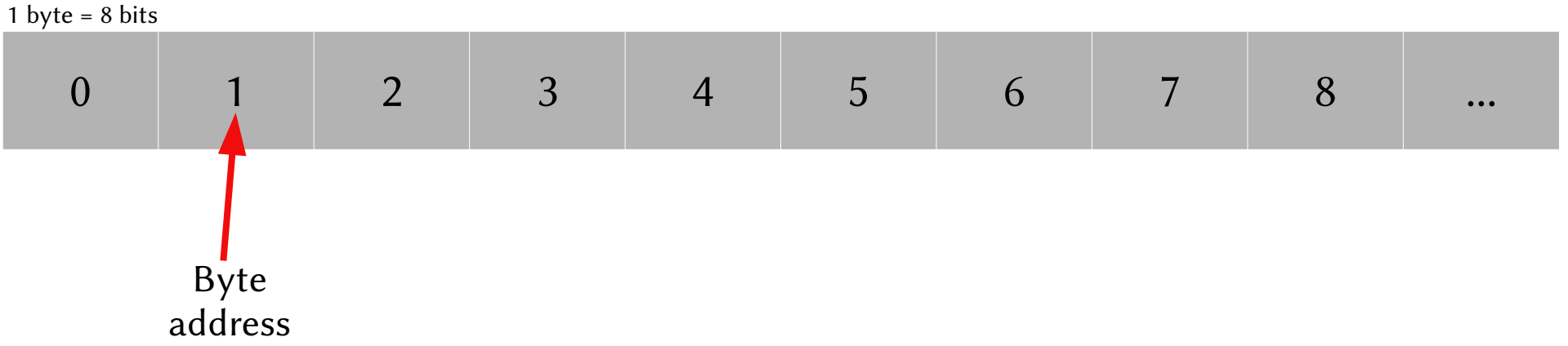
Javascript, Lua, Maple, Mathematica...

PHAS0102 Part 1: High-Performance Computing with Python

Numpy and Scipy

- Numpy
  - Fast data types for vectors and matrices
  - Linear algebra operations
- Scipy
  - Matrix algorithms, optimisation algorithms, etc

Memory layout



- It is faster to access a set of nearby bytes than a set of bytes that are spread out.
- Lower level languages like C and Fortran allow you to control exactly how memory is used, so can be much faster.

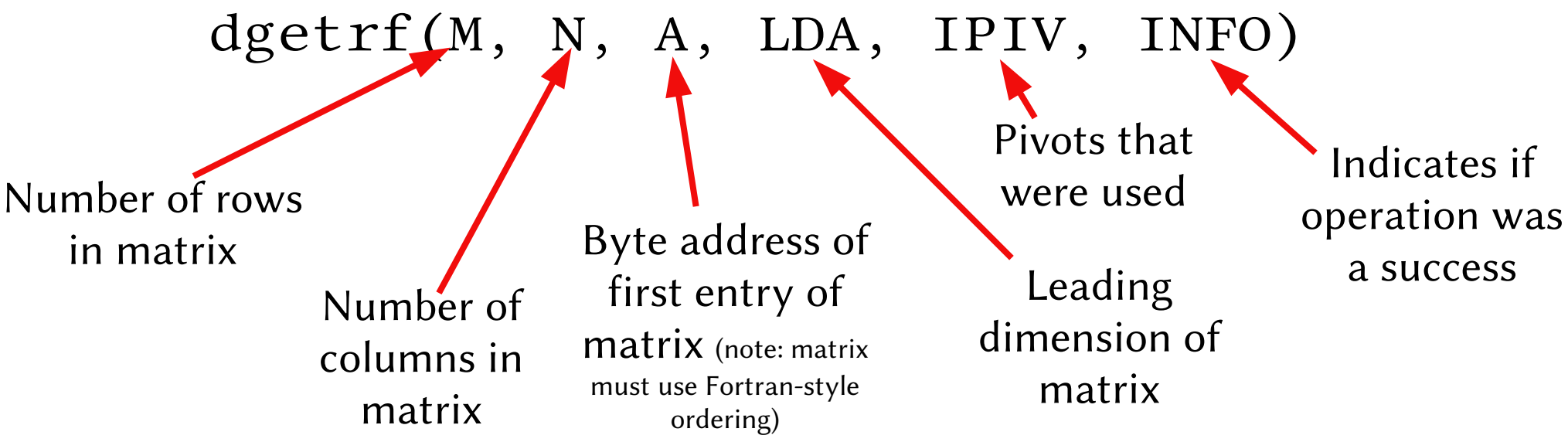
Storing a matrix

0	1	2	3	4	5	6	7	8	0
	$\begin{pmatrix} 5 & -3 \\ 2 & 1 \end{pmatrix}$		5	-3	2	1	← C-style ordering		
			5	2	-3	1	← Fortran-style ordering		

- By default, Numpy uses C-style ordering, but you can tell it to use Fortran-style instead.

BLAS and LAPACK

- BLAS = Basic Linear Algebra Subroutines
- LAPACK = Linear Algebra Package

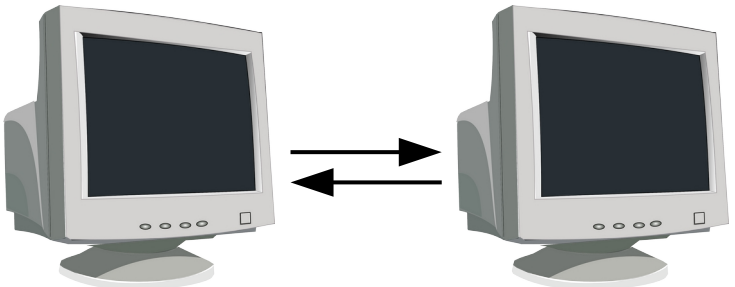


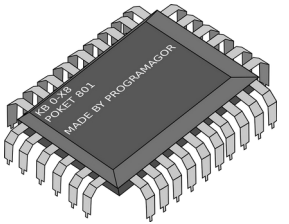
Numpy

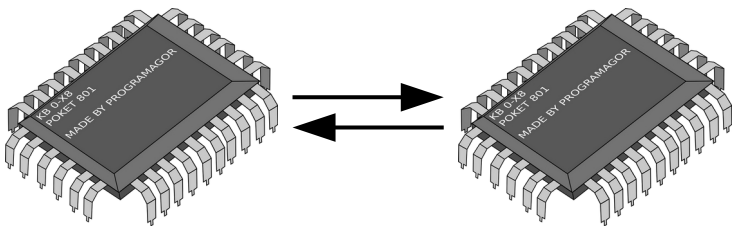
- Internally, Numpy uses BLAS and LAPACK routines.
- Numpy handles the memory layout, data types, etc for you.

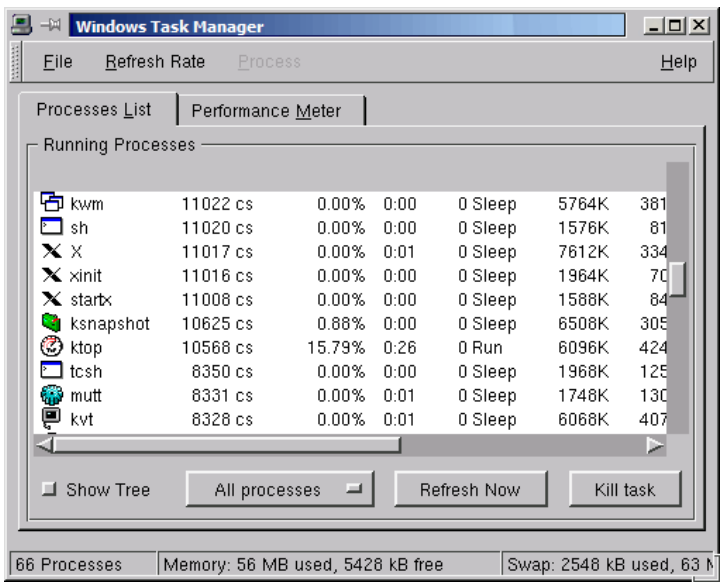
[live Numpy demo]

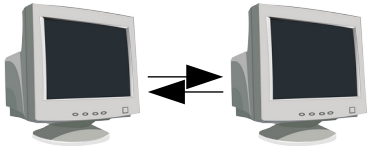
Parallelisation





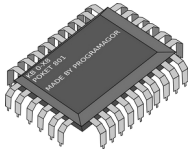






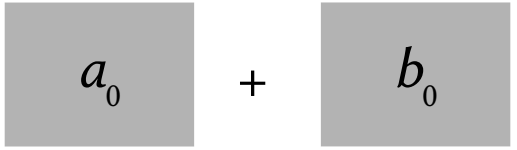
MPI

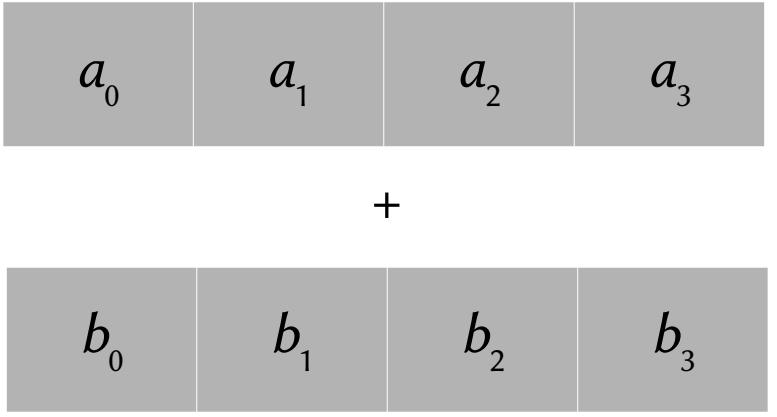
- MPI = Message Passing Interface
- Used to send commands and data between processors / computers
- `import mpi4py`



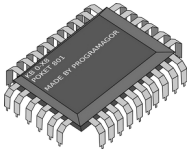
SIMD

- SIMD = Single Instruction Multiple Data





- AVX2 – 256 bits can be operators on in a single CPU cycle.

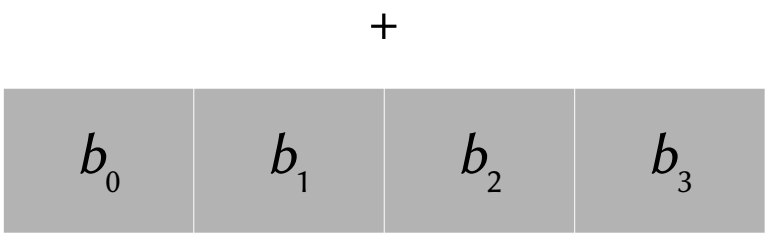


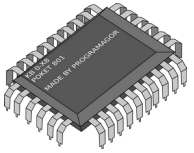
SIMD

256 bits = 32 bytes

Single precision floating point number is 4 bytes ⟶ 8 singles at once
Double precision floating point number is 8 bytes ⟶ 4 doubles at once

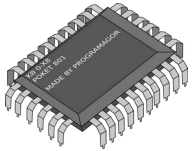
- AVX2 – 256 bits can be operators on in a single CPU cycle.





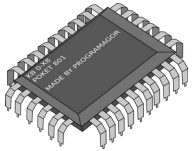
SIMD

- Some CPUs support AVX-512 (512 bits in one operation) for some operations



Multithreading

- Multithreading = running multiple threads / processes at once on the same CPU
- `import multithreading`
- Due to how Python's memory management works, only one thread will be active at a time.



Multithreading

- Next Friday: Using Numba to do proper multithreading with Python

- Monday: Hands-on Numpy practice (tasks to work through in lecture notes or Moodle or mscroggs.co.uk/PHAS0102)