

PyQB

Monga

Flow of control

# Programming in Python<sup>1</sup>

Mattia Monga

Dip. di Informatica Università degli Studi di Milano, Italia mattia.monga@unimi.it

Academic year 2023/24, I semester

1 ⊕⊕@ 2023 M. Monga. Creative Commons Attribuzione — Condividi allo stesso modo 4.0 Internazionale. http://creativecommons.org/licenses/by-tsa/4.0/deed.it > 4 € > € → Q (~



PyQB

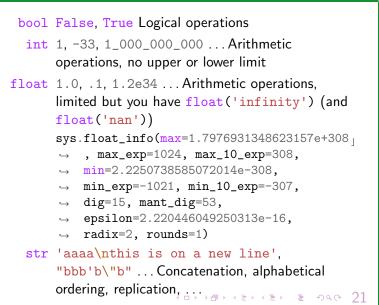
Monga

Flow of control

### Lecture III: Control flow

< 中 ト 4 周 ト 4 差 ト 4 差 ト 差 の 4 (や 20)

#### Basic types



Manage

Flow of control

### Sequence of operations

```
x = 1 + 2 * 3
>
```

```
x = x + 1
```

The 2 lines of code translate to at least 5 "logical" instructions (maybe more, for example adding two big numbers require multiple instructions):

(日) (個) (目) (目) (目) (22)







Flow of control

It is normally not very useful to write programs that do just one single computation. You wouldn't teach a kid how to multiply  $32 \times 43$ , but the general algorithm of multiplication (the level of generality can vary).

To write programs that address a family of problems we need to be able to select instructions to execute according to conditions.

	11 x == -1:
if $x < 0$ :	$\mathbf{x} = \mathbf{x} + 1$
$\mathbf{X} = -\mathbf{X}$	else:
	$\mathbf{x} = 3 + \mathbf{x}$
v = 2 * x	

y = 2 \* xIn Python the indentation is part of the syntax and it is mandatory.



Manag

Flow of control Selections

# Input (special command needed)

- A special command to ask to the operating system (same as print)
- input() or input("Prompt the user:")
- The operating system (or the operating environment as in cscircle) collect the input data (from keyboard/console or the network in cscircles) and returns them to Python as a str.
  - s = input() ## read a string
  - i = int(input()) ## read a string, convert to int
- Input on cscircles seems strange, but when one understands the need of the mediation, the machinery is rather straighforward



Monga

Flow of control Selections

<ロト 4 @ ト 4 E ト 4 E ト E 9 Q @ 24