Open Roberta (Blockly-based)

Andrea Sterbini – sterbini@di.uniroma1.it

Open Roberta
Simple visual robot/microcontroller programming

Built with Blockly lab.open-roberta.org

Transforms visual programs to Python/Java/C/C++ (depending on which type of robot)

Deploys the program on the robot

Runs the program on the robot (or a simulation on the PC)

Debug the program by stepping/tracing it

Visual interface to the robot configuration details

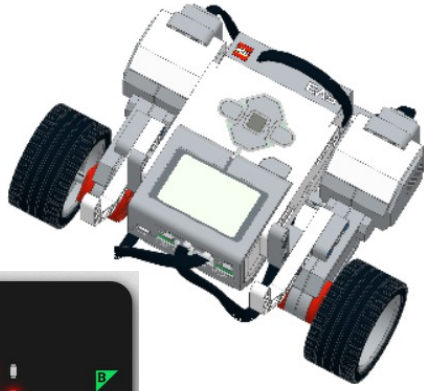
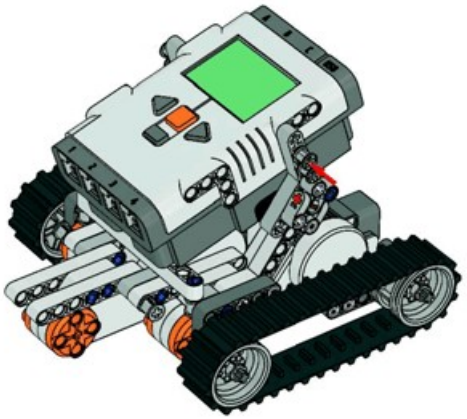
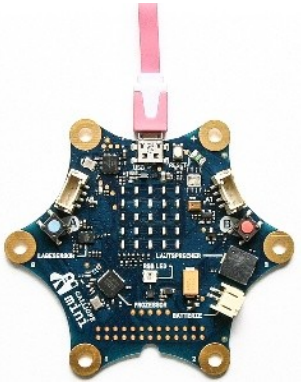
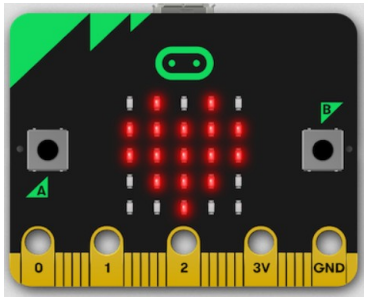
Motors, sensors, wheels geometry, LCD displays, LEDs, ports, shields

WIKI: https://jira.iais.fraunhofer.de/wiki/display/ORInfo

Open Roberta
Many robots and embedded systems supported

NAO, BOB3, Lego WeDo 2, Lego EV3, Lego NXT, Bot'n Roll, Calliope Mini, Micro:bit, Arduino, mBot, senseBox

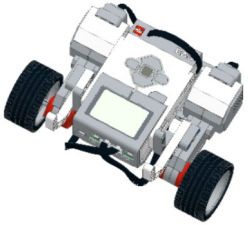


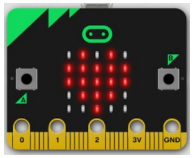







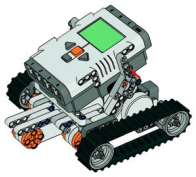

Many generated languages

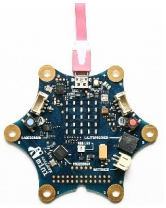
Python: Lego EV3 micro:bit NAO



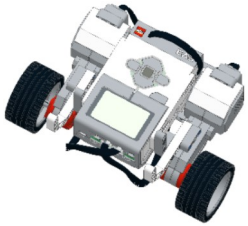


C/C++: Arduino, Bot'n roll, Lego NXT and EV3, BOB3, SenseBox, mBot, Calliope



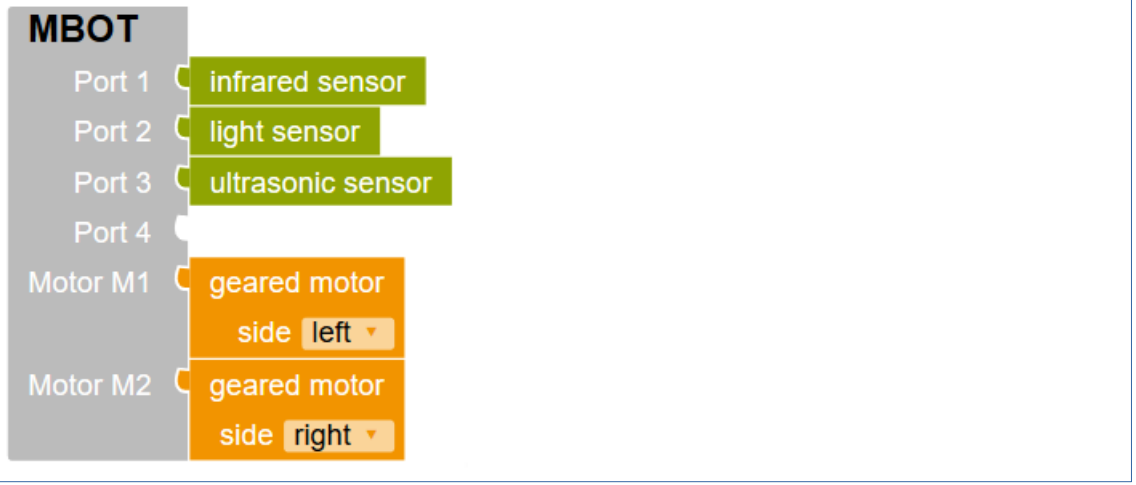


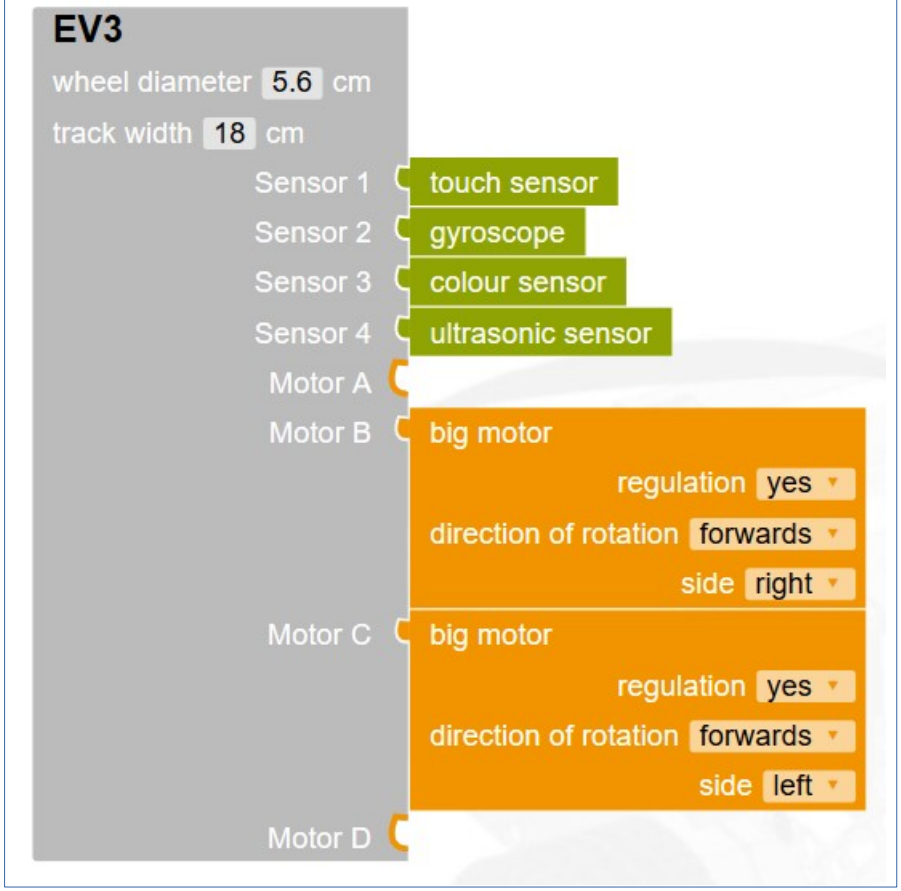
Java: Lego EV3

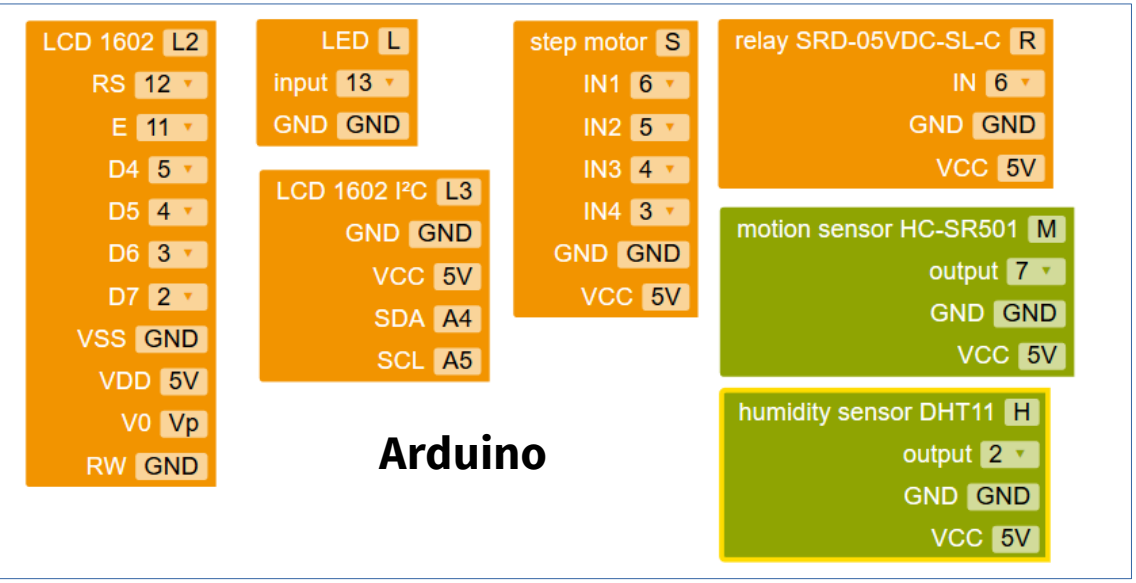


Json: Lego WeDo (runs on PC)

Visual Robot/Microcontroller configuration of the sensort/actuators connected (and where)

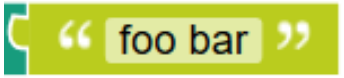




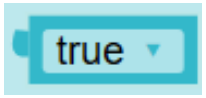


Data types

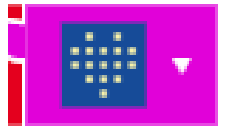
Number 

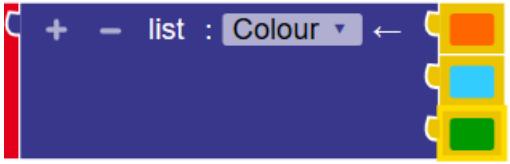
String 

Colour 

Boolean 

Connection

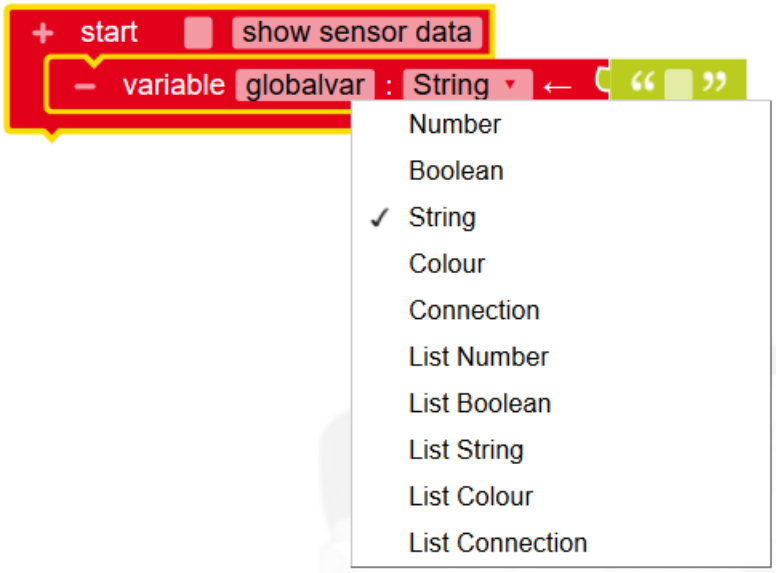
Image 

List of <T> 

(same type for all elements)

Variables and arguments are <u>typed</u>
(the connector is coloured)

Data types are <u>visually enforced</u>
(cannot join if the type is wrong)



Execution model: single thread

Single thread of execution (main program/main loop)

New Functions?	YES	
Global variables?	YES	(defined only at main level)
Local variables?	YES?	(defined as function's arguments)
Messages?	NO?	(but some robots can communicate)
Events?	NO	

Events simulated by polling the sensors + "when"

Lego EV3 robots can connect via BT and exchange text messages

Other robots can communicate over serial wires

“Advanced-enough” programming

Counted Loops, Foreach,
Repeat until, Repeat while

Continue, break

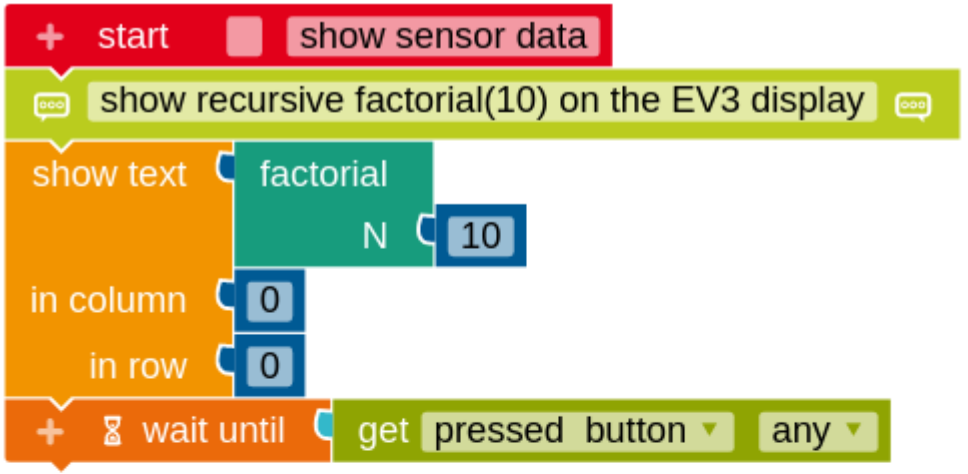
Wait N ms,
Wait until condition …
or other condition … or else

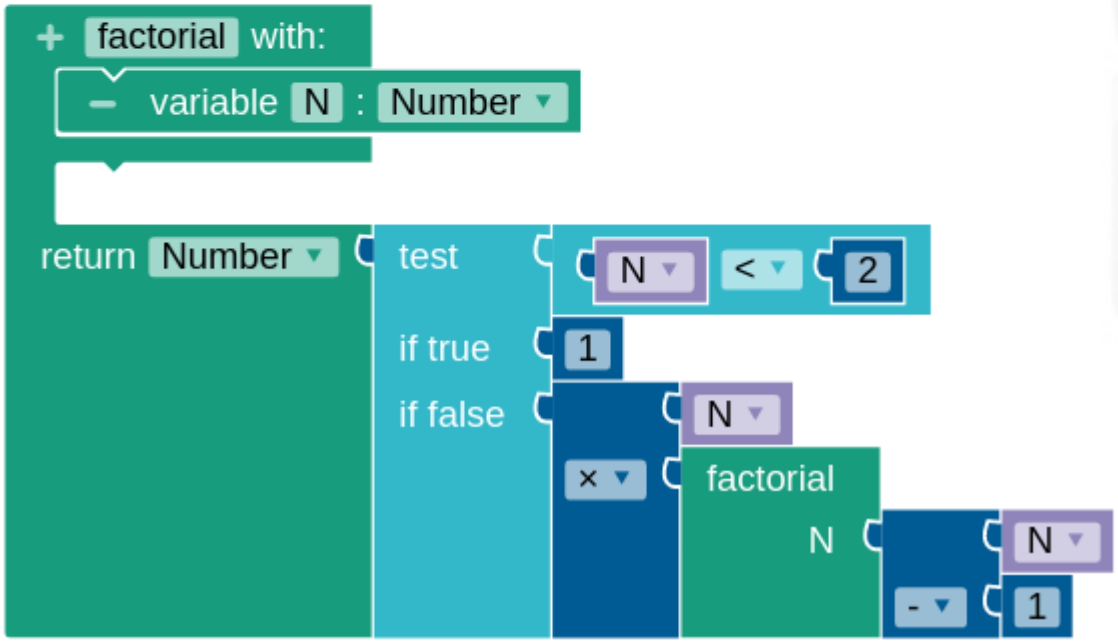
If, if-else, if-elif-…-else

Constrain value between

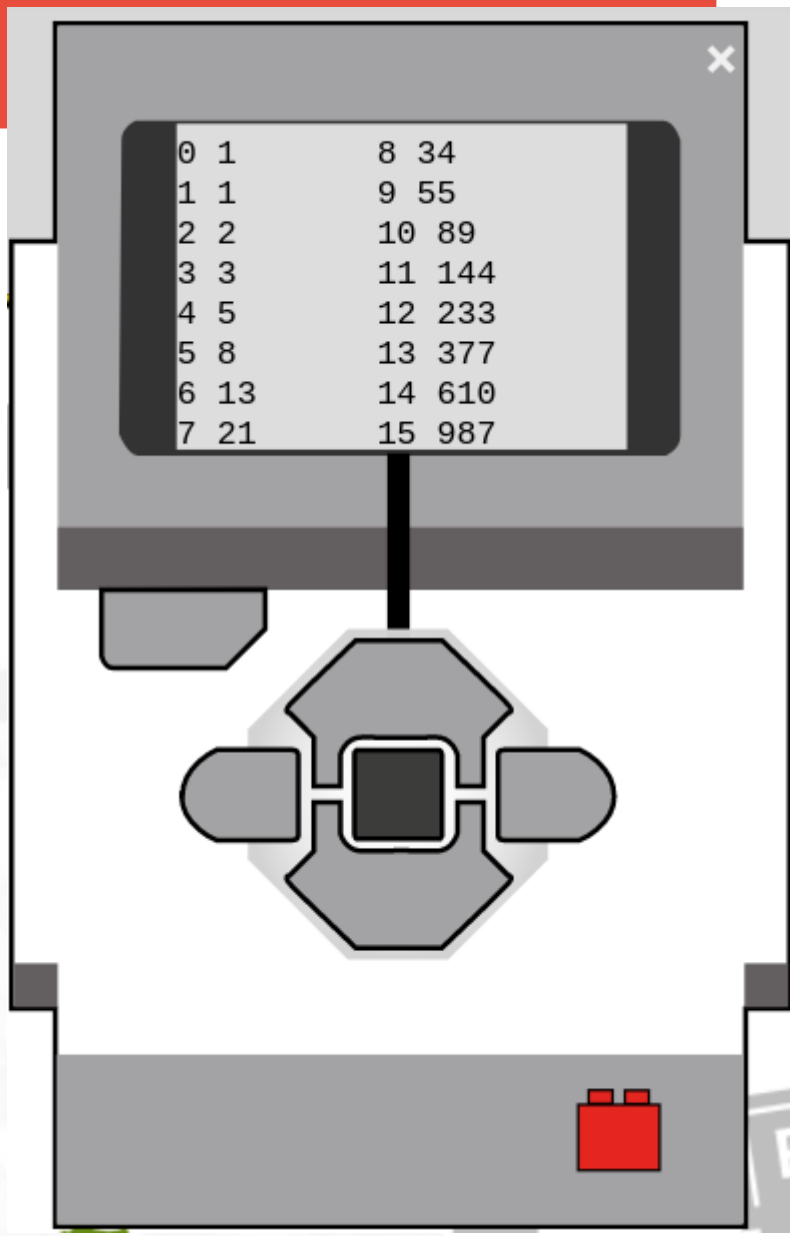
Recursion? YES

Local variables as arguments(!)





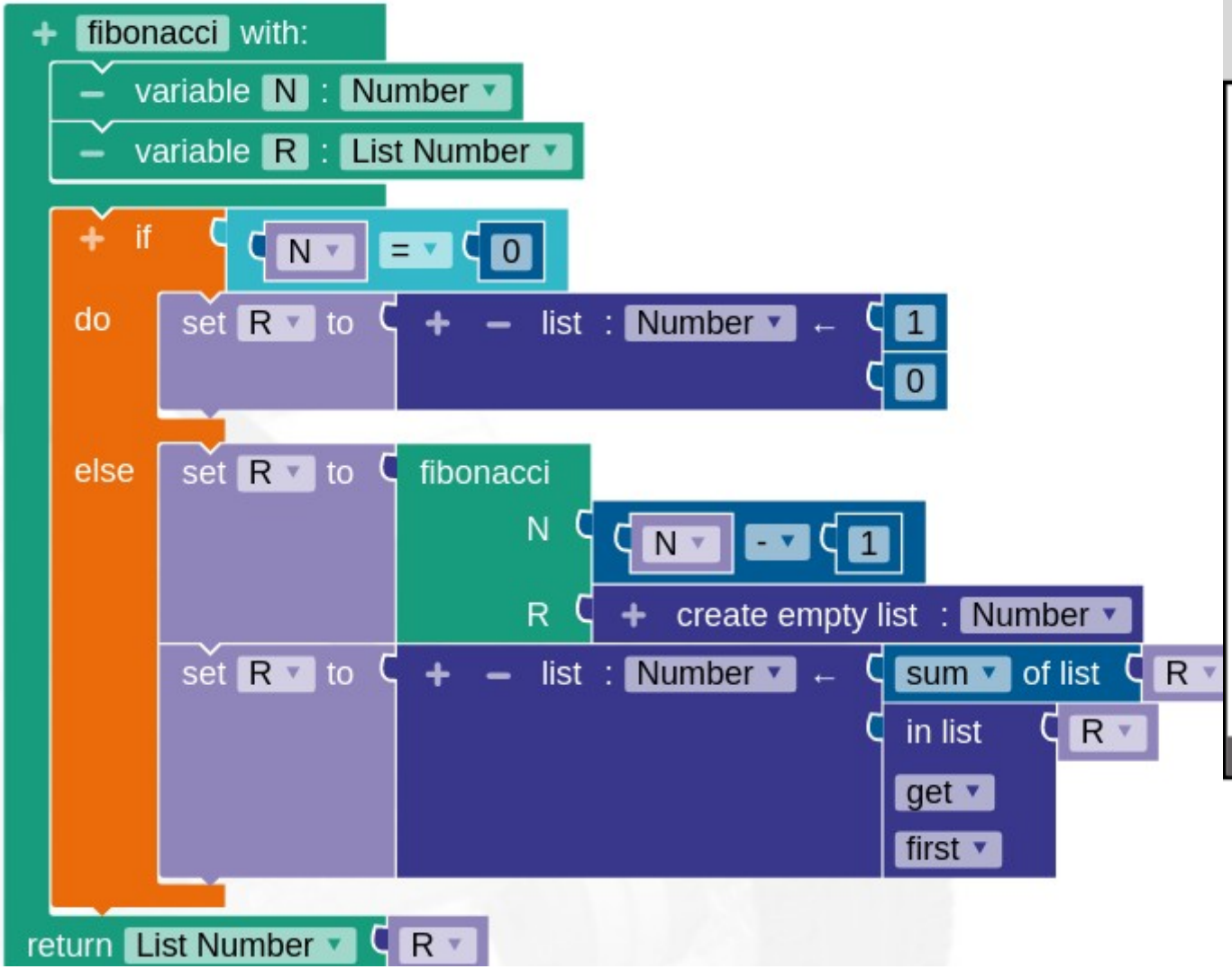
Example: efficient recursive Fibonacci (forward loop simulation)

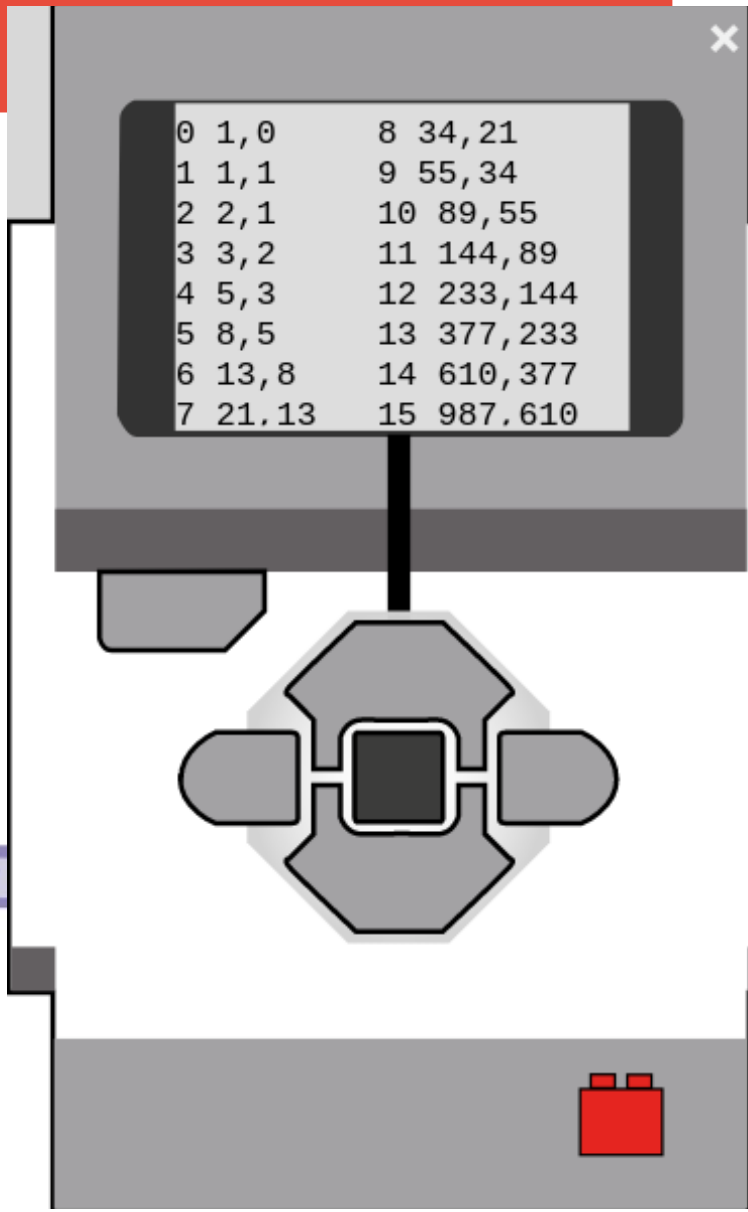



```
+ fibonacci with:
    – variable N2 : Number

  return Number   fibonacci_  N  N2   F  1   Fold  0
```

```
+ fibonacci_ with:
    – variable N : Number
    – variable F : Number
    – variable Fold : Number

  return Number   test  N  =  0
                  if true   F
                  if false  fibonacci_  N  N  -  1   F  F  +  Fold   Fold  F
```

Example: efficient recursive Fibonacci (backward loop simulation)





Local install (for a better network access)

Open source

Available on https://github.com/OpenRoberta/openroberta-lab

Java based, built with Maven

You can enable/disable separately each module

You can run the server on your laptop in class and share your wifi

Robots and PC browsers in the class connect by wifi to your laptop

Available also for Android

Demo

Demo