Other notable systems

Andrea Sterbini – sterbini@di.uniroma1.it

Fancade: a 3D data-flow game building language

Data-flow computation WITH VARIABLES

Typed links (Number, Boolean, Position, Rotation, Object, Constraint, …)

NO text! (except for code comments and menus)

Executed 60 times for second (implicit forever loop)

Execution order: left-right and top-down (with respect to the isometric positions)

Explicit Variables (used also to reduce wires and increase fan-out)

Code can read/write game blocks properties (position/speed/bounce …)

Can add or change movement/rotation constraints for blocks/agents

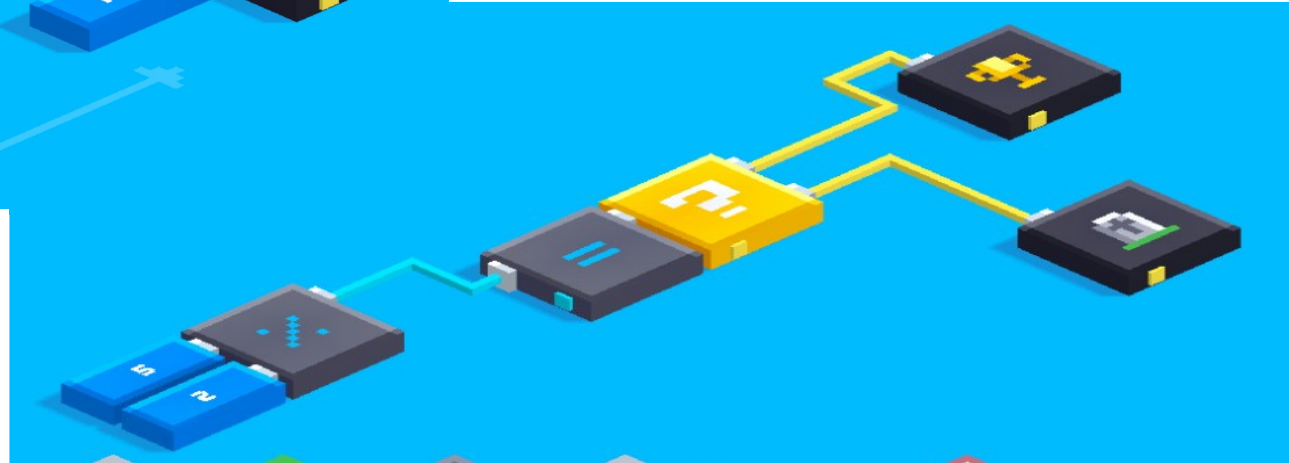# Explicit Control = before/after links (jellow wires)
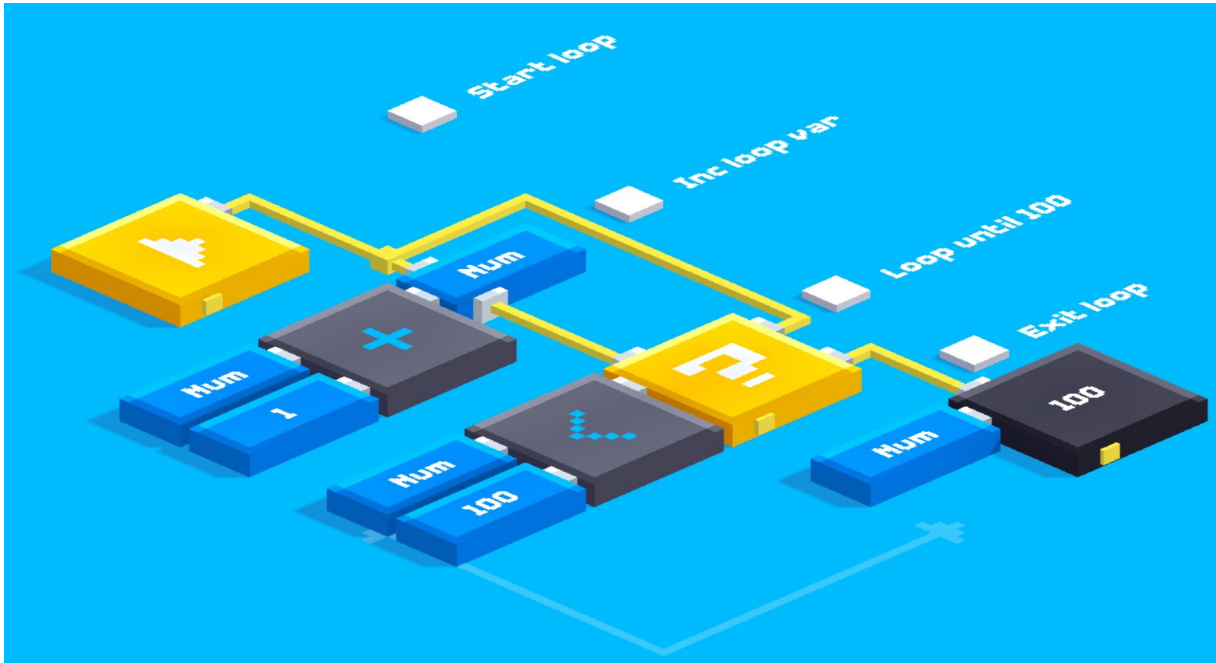
Loops

do:
    Num += 1
    while num<100

If-then-else

if 5%2==0:
    then: win else: lose

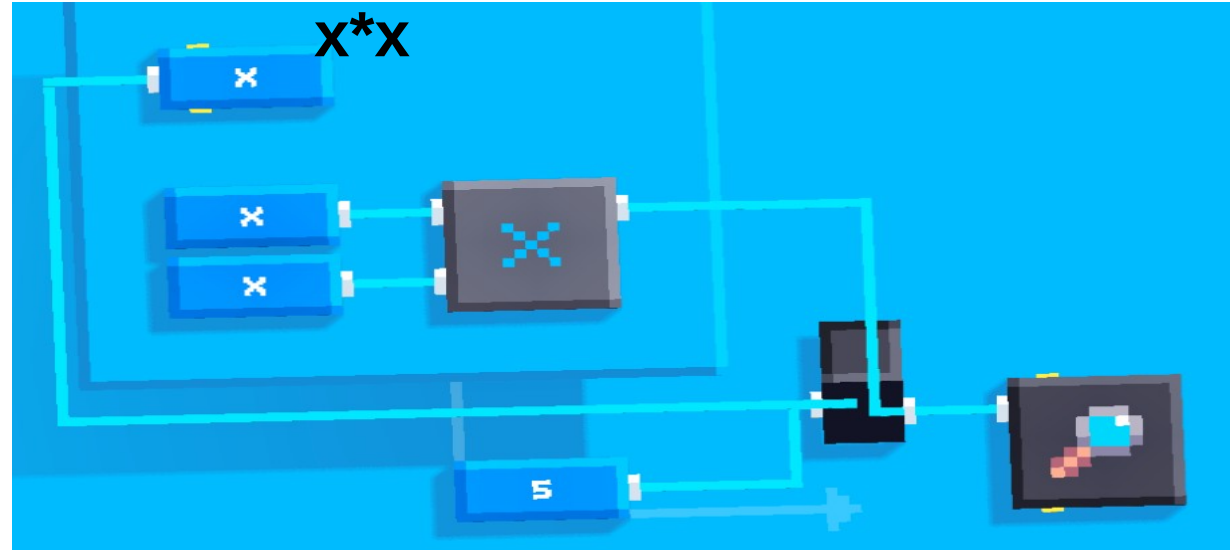# Functions: custom blocks

Functions?   YES

Just put "code" inside a block, with external connectors

def square(x): return x*x

Modularization:
- custom blocks can be placed inside other blocks
- BUT: <u>NO recursion is allowed</u> (direct or indirect)

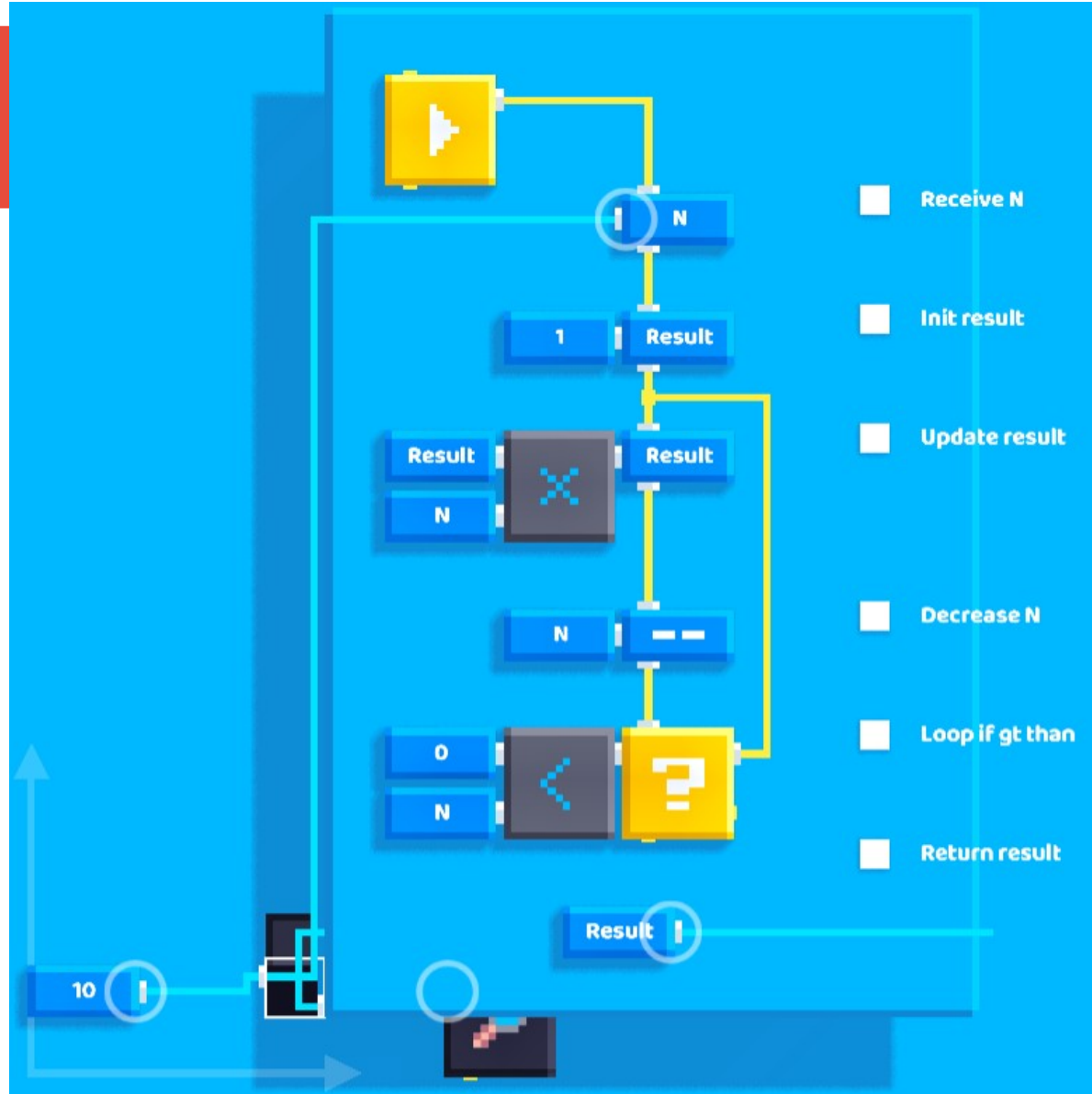# No recursion (even implicit)

Iterative factorial(N)

VALUES (different types)

VARIABLES (getter/setters

MATH (operators)

CONTROL (if-then, loops ...

PHYSICS (gravity, vectors,

GAME (win/lose/score)

SCENERY

SCRIPTS

INSPECTORS

CREATURES

Values

Variables

Control

Physics

Game

Objects
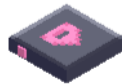
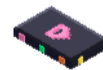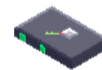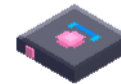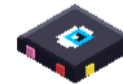|					
Get Position	Set Position	Raycast	Get Size	Set Visible	Create Object

Destroy Object

DEMO

PlayOsmo: <u>tangible</u> interaction for small kids (with iPad or Fire)

Your <u>tablet scans the table in front of it</u> (on a stand, with a mirror on the camera)

You play with tiles (instructions / tangram pieces / letters / drawings)

Recognize your "code" / words / drawing depending on the game/app selected

Loops and simple repetitions

Blocks represent action and direction

- modifier: number of repetitions

Loops

The instructions' <u>meaning depends on the game</u>

- Music playing game

- Monster in a labyrinth game

But also many game apps

Drawing (virtual reality helped drawing)

Spell/word games (char recognition)

Tangram (shape recognition)

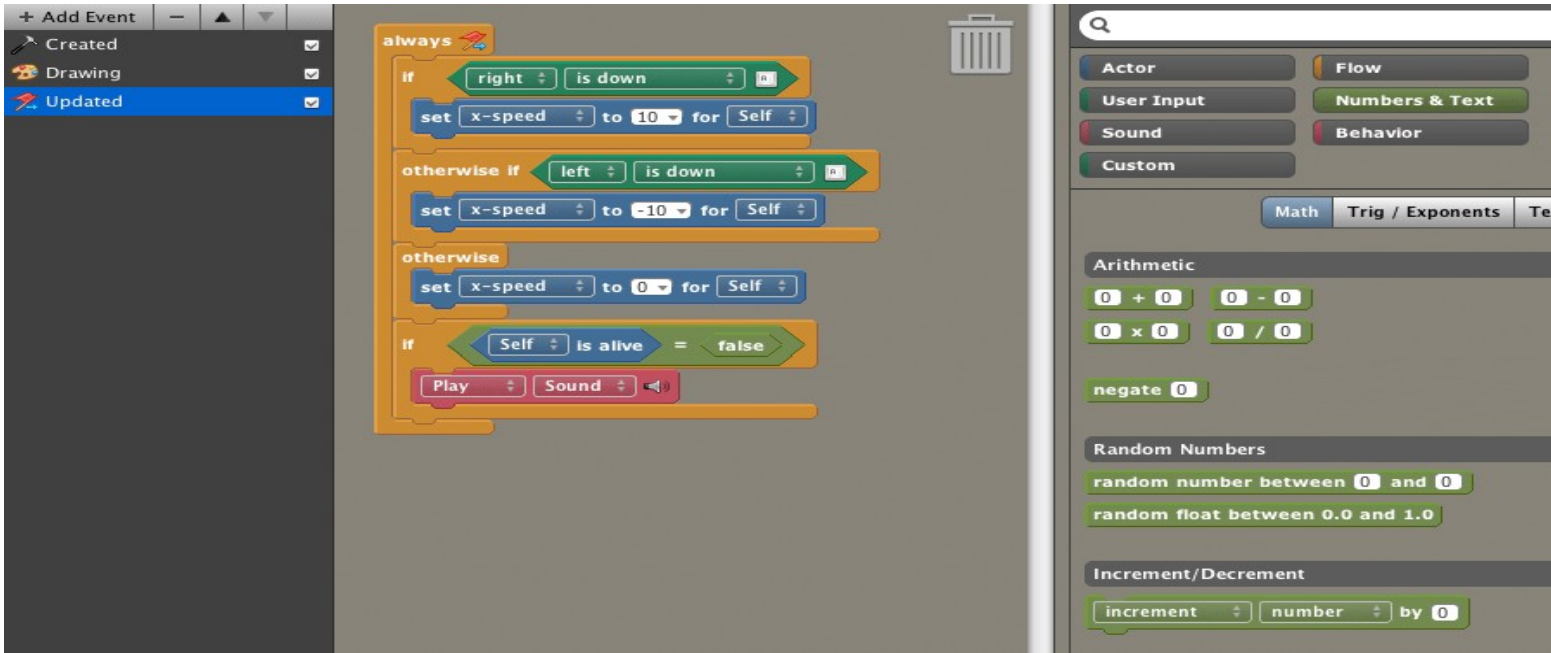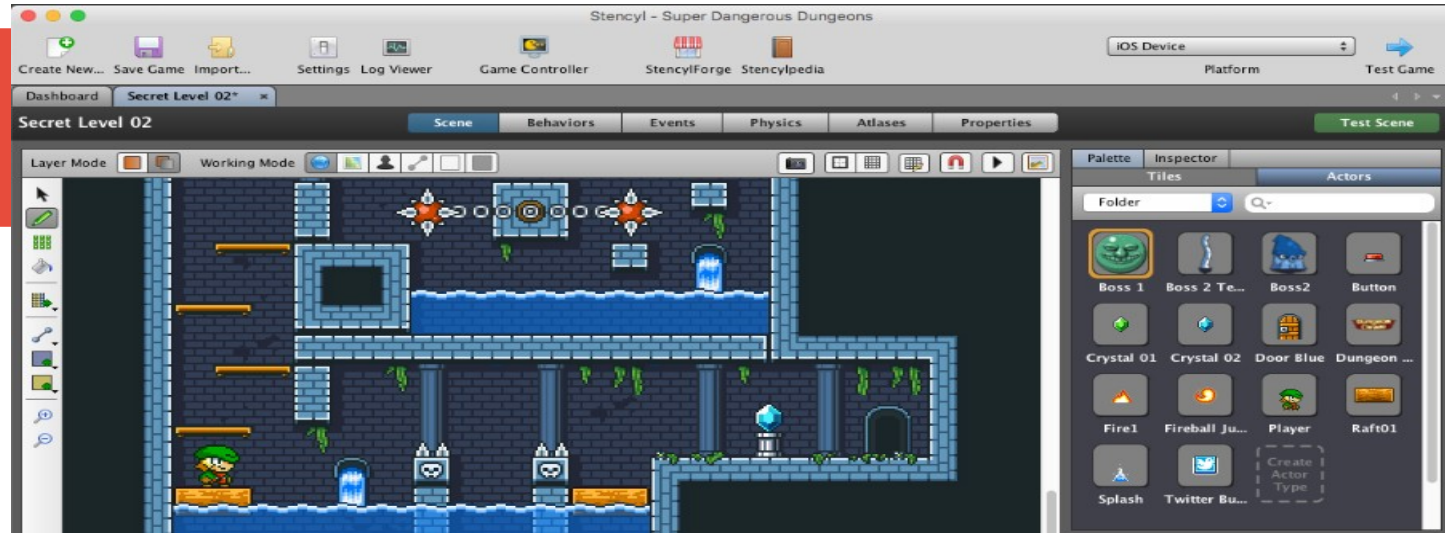Pizza shop (shapes + money/math)
Wizards (math or words)
...

Stencyl

Game designer (platform games)

Alice 3 : 3D world programming (Java)

Ren'Py

Dialogue-based adventures (Visual Novels)

Python-inspired simplified syntax (with indentation)

Runs in Python

Others

Kodu: Design 3D games on Xbox (and PC)

Kodular: game editor

Construct: HTML5 game editor

Swift Playground: by Apple

MakeBlock: robots/microcontrollers + Scratch

Tynker: Scratch-inspired

Roblox: 3D game programming

… and many other game-editors

No Code development?

Goal: allow non-programmers to jot-down easy applications

Template-based development environments where you:

- define the data tables and their relations

- define the (web) interface/pages showing and managing the data
    - with a lot of widgets already available

- define the operations applied on the data when you interact with the pages
    - in general very simple operations / computations

- define the workflow steps

Many ready-made templates, apps and modules are available

Systems: **Softr**, **Bubble**, **Glide**, **Draftbit**, **Zapier**, **Bildr**, **Backendless**, **FlutterFlow**, **...**

Low Code development platforms

Simple development with <u>minimal</u> coding

E.g.:

- Microsoft PowerApps:
Excel-like formulas + database/spreadsheet + GUI editor + CoPilot (AI)

- Google AppSheet:
Actions/Workflows + database/spreadsheet + GUI editor

- Appian:
Low-code + Workflows + Process Analysis + AI

- ...