

Tecnologie didattiche

Tecnical Report

Marco Righi*

6 agosto 2022

keyword: robot, didattica, IoT, Arduino, ESP32, ESP8266, NetLogo, Scratch, elettronica, programmazione imperativa, programmazione visuale, tecnologie STEM

DOI:10.5281/zenodo.6968690

**marco.righi@isti.cnr.it*, Istituto di Scienza e Tecnologia, Consiglio Nazionale delle Ricerche, 56124 Pisa, Italia

Indice

1	Sommario	3
2	Introduzione	3
3	Problem Solving: dall'algoritmo alla realizzazione	3
4	Dal gioco alla realizzazione	3
5	Conclusioni	4

1 Sommario

Questo documento riassume le principali tecnologie didattiche hardware e software utilizzabili in base all'età per lo sviluppo dei concetti scientifici nelle nuove generazioni.

2 Introduzione

Le tecnologie orientate alla didattica si sono rapidamente evolute negli ultimi anni tanto da creare un mercato consumer accessibile: tra questi troviamo prodotti LEGO tra cui la serie Mindstorm[7], dispositivi elettronici facilmente programmabili come i Microbit.org [9] e altri prodotti reperibili in negozi specializzati per hardware [3, 13]. I prodotti specializzati sono più versatili ma al contempo richiedono una maggiore specializzazione per il loro utilizzo. I prodotti LEGO rappresentano uno strumento adattabile a diverse esigenze con percorsi che possono coinvolgere bambini dai 6 anni in su, questo è reso possibile sia dal sistema di programmazione visuale basato su Labview che astrae da una sintassi formale sia dagli insegnanti che possono guidare gli studenti. I dispositivi Micro:bit, diversamente dai LEGO, molto legati all'elettronica, sono consigliabili da bambini dagli 8 anni (fino ai 14 anni). I vantaggi e i limiti di questi strumenti hardware/software come vedremo coincidono: la loro combinazione fa sì che tra di due debba esserci un protocollo codificato affinché il sistema nel suo complesso funzioni. A esempio, se abbiamo un robot con due ruote parallele che deve muoversi in linea retta, una ruota deve girare in senso orario mentre l'altra in senso antiorario altrimenti il robot gira su se stesso. Didatticamente questo permette di avere tutti i dettagli realizzativi e ciò è un vantaggio, contestualmente, per una sperimentazione estesa il dover realizzare fisicamente molti modelli meccanici è un limite. Sottolineiamo l'esistenza di strumenti software che astraggono totalmente dall'hardware e che permettono di sfruttare le nuove tecnologie in modo diverso, tra questi strumenti troviamo Netlogo [10], Geotortue [5], Scratch [14] e ScratchJr [15] (solo per citarne alcuni). Tali strumenti software possono essere usati da bambini di 5 anni (es. ScratchJr) e quindi anche in età prescolare. Questi strumenti non presentano in generale particolari limiti di età per il loro utilizzo in quanto esso è funzionale a quello che si vuole realizzare.

3 Problem Solving: dall'algoritmo alla realizzazione

La curiosità scientifica dei ragazzi è molto stimolata dalla robotica dove la Lego si è decisamente affermata. Con questi strumenti gli studenti imparano la meccanica della robotica (per quanto semplificata i principali problemi dei "robot veri" sono presenti in quelli Lego) e istintivamente le leggi matematiche e fisiche a essa legata. Successivamente con l'ambiente di programmazione grafico [8] (1) anche i più piccoli possono scrivere dei veri e propri programmi per "dare vita" ai robot. I linguaggi grafici, per quanto siano un valido strumento per programmare e siano usati anche in ambito professionale come LabView [6] rappresentano un approccio limitato a quelle che sono le tecnologie per trasformare un'idea in un automatismo per la soluzione di un problema. Ci vengono incontro altri strumenti legati al paradigma di programmazione imperativa come il C [11, 2]. Esistono anche altri paradigmi di programmazione fino a qui non contemplati come la programmazione funzionale o la programmazione logica. Capire come funzionano queste tecnologie e confrontarsi con esse è un lavoro impegnativo e educativo che permette di intravedere quanto lavoro vi sia in quei contesti che implicano attività dove hardware e software devono avere una coerenza tale da permetterne il corretto funzionamento. Gli studenti imparano a manipolare certi tipi di strutture sia hardware che software entrando in prima persona nei problemi a essa legati (come i suddetti problemi di meccanica elementare) ma al contempo pone delle limitazioni che prodotti interamente software non hanno. Pensiamo a un ambiente Netlogo dove possono interagire centinaia di agenti [17]. Costruire centinaia di robot è certamente fattibile ma non con la semplicità con cui si istanziano centinaia di agenti! Avere centinaia di entità che debbono interagire è un passo importante per comprendere che cosa significa condividere delle risorse e contestualmente avere le nozioni base che permettono di far funzionare i moderni sistemi operativi dei computer. La mancanza della parte fisica permette inoltre di simulare dei comportamenti "sociali" tra agenti che possono concorrere alla vita in un ambiente (l'ambiente dove si muovono e in un certo senso vivono la loro vita digitale).

4 Dal gioco alla realizzazione

Gli ambienti fin qui presi in esame a livello didattico sono certamente un'ottima palestra per imparare e forniscono gli strumenti per accedere con facilità a ambiti lavorativi e sociali in forte espansione come, a esempio, lo sviluppo di applicazioni e servizi nell'ambito dello IoT (Internet of Things). Infatti grazie alle varie piattaforme hardware a

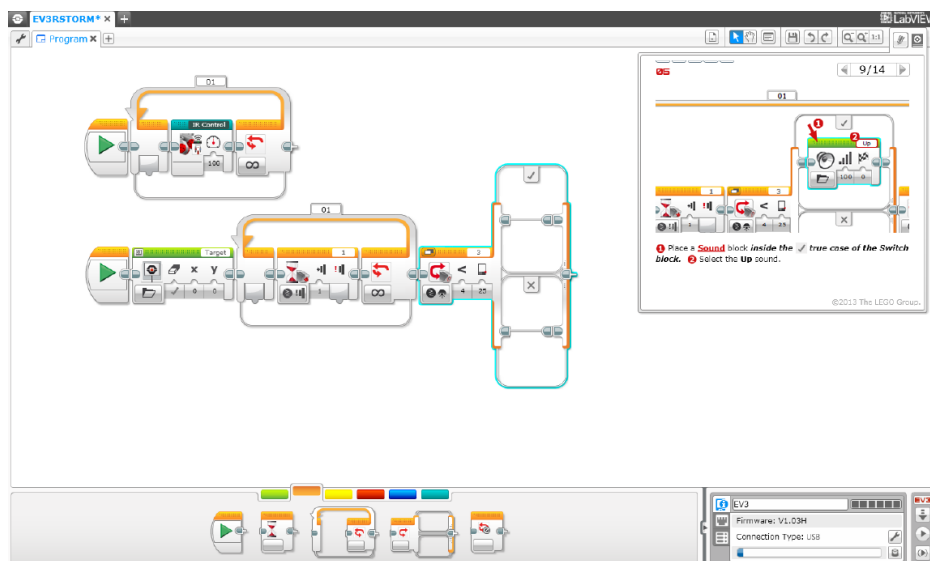


Figura 1: linguaggio grafico per bambini

basso costo, come Raspberry, Arduino, ESP8266 e loro lavorazioni [4, 16, 12, 1], è possibile sviluppare idee innovative partendo da investimenti minimi. Dominare queste tecnologie necessita di un background ben più ampio di quello fin qui illustrato ma le conoscenze acquisite forniranno comunque dei riferimenti importanti per molte attività. A complemento di questi studi per lo sviluppo completo di sistemi di IoT vi sono l'elettronica e l'elettrotecnica. Per quanto riguarda i problemi di elettronica, alcuni aspetti possono essere insegnati con lo studio dei dispositivi quali Raspberry, Arduino e ESP8266.

5 Conclusioni

Le varie tecnologie qui esposte non hanno la pretesa di dare una descrizione esaustiva dell'offerta del mercato ma di fornire delle idee con riferimenti a realtà che già esistono. Questo permette di iniziare a esplorare e a creare delle realtà basate su strumenti consolidati e al contempo innovativi per l'insegnamento delle scienze teoriche e applicate per mezzo delle nuove tecnologie

Riferimenti bibliografici

- [1] Arduino. <https://www.arduino.cc/>. [Online; ultimo accesso 27-luglio-2022].
- [2] Bricx command center 3.3. <http://bricxcc.sourceforge.net/>. [Online; ultimo accesso 27-luglio-2022].
- [3] Campustore. <https://www.campustore.it/>. [Online; ultimo accesso 27-luglio-2022].
- [4] Esp8266. <https://it.wikipedia.org/wiki/ESP8266>. [Online; ultimo accesso 27-luglio-2022].
- [5] Geotortue. <http://geotortue.free.fr/>. [Online; ultimo accesso 27-luglio-2022].
- [6] Labview. <https://www.ni.com/it-it/shop/labview.html>. [Online; ultimo accesso 27-luglio-2022].
- [7] Lego. <https://www.lego.com/it-it/kids/search?q=mindstorm>. [Online; ultimo accesso 27-luglio-2022].
- [8] Lego: Impara a programmare. <https://www.lego.com/it-it/themes/mindstorms/learntoprogram>. [Online; ultimo accesso 27-luglio-2022].
- [9] Microbit. <https://microbit.org/>. [Online; ultimo accesso 27-luglio-2022].
- [10] Netlogo. <https://ccl.northwestern.edu/netlogo/>. [Online; ultimo accesso 27-luglio-2022].

- [11] Programming ev3 in ansi c. <http://www.thenxtstep.com/2014/08/programming-ev3-in-ansi-c.html>. [Online; ultimo accesso 27-luglio-2022].
- [12] Raspberry pi. https://it.wikipedia.org/wiki/Raspberry_Pi. [Online; ultimo accesso 27-luglio-2022].
- [13] Robot-italy. <https://www.robot-italy.com/>. [Online; ultimo accesso 27-luglio-2022].
- [14] Scratch. <https://scratch.mit.edu/>. [Online; ultimo accesso 27-luglio-2022].
- [15] Scratchjr. <https://www.scratchjr.org/>. [Online; ultimo accesso 27-luglio-2022].
- [16] Wemos. <https://www.wemos.cc/en/latest/>. [Online; ultimo accesso 27-luglio-2022].
- [17] Stuart Jonathan Russell and Peter Norvig. Artificial intelligence: A modern approach, 2016.