Dietro il coding



"Je suis fracas quand la foule est tranquille" (Zebda, 1998)

Stefano Penge 2016 Creative Commons BY/SA

steve@lynxlab.com

http://steve.lynxlab.com

What is Coding?

Sta per "introduzione anticipata alla programmazione dei computer tramite un linguaggio visuale in un ambiente giocoso online".

Coding è più corto, come i media, i social, etc.

- Perché anticiparla? Perché per tutti? Perché visuale? Perché online?
 - Obiettivi a breve e lungo termine, interessi, pedagogie
- Con cosa?
 - Linguaggi, modelli
- Chi e con quali competenze?
 - Didattiche, tecniche, culturali
- In quale contesto?
 - Code.org, Minecraft, Apple Swift Playground, ...
- Quando?
- Che ne pensano i docenti, le famiglie, i bambini?

- E' facile
- E' libero: non c'è un curriculo ufficiale
- E' utile
- E' nuovo

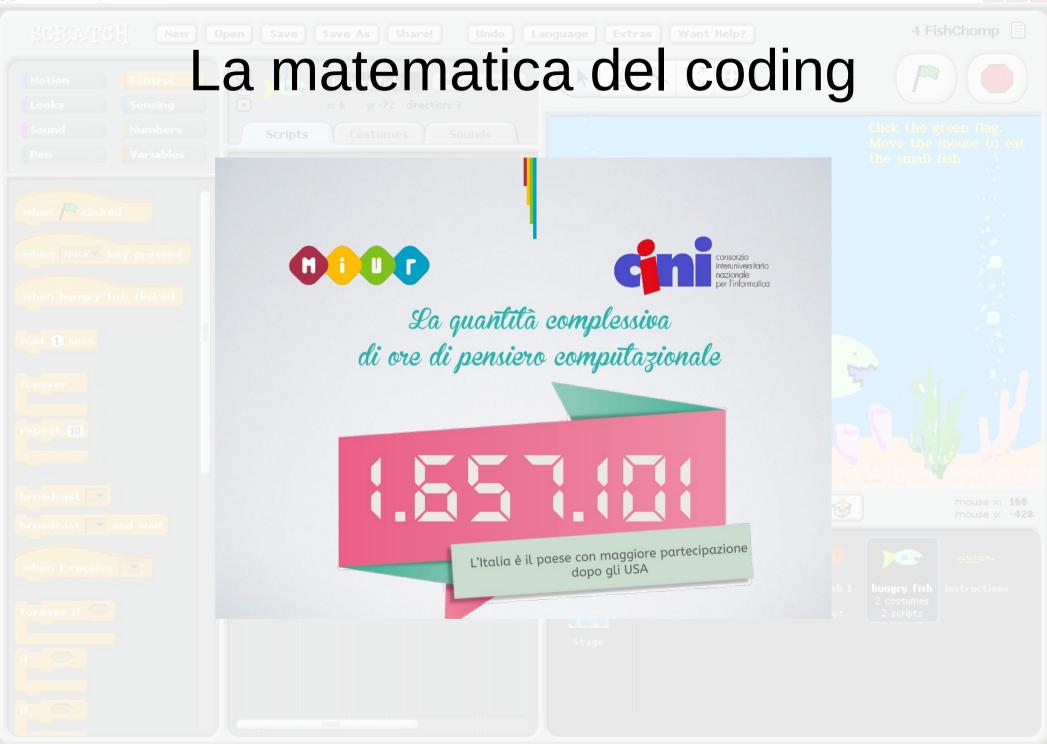
Il bello del coding?

- Ha a che fare con la cultura scientifica (STEM) ma anche con lo story telling (e cioè?)
- E' divertente (per chi? Fino a che età?)
- E' facile (fino a dove?)
- E' libero: non c'è un curriculo ufficiale (finora)
- E' utile (a cosa?)
- E' nuovo (insomma...)

Visioni diverse



- Ma io che c'entro, io insegno latino...
- E' tutta una manovra far entrare l'aziendalismo nella scuola e per sottomettere i docenti alle logiche del mercato.
- Che ti devo dire? Con Scratch ci divertiamo un sacco, l'ora passa in un attimo
- La "scuola digitale" non esiste e non sono gli strumenti che possono indurre cambiamenti significativi anche se conditi di (pseudo) metodologia.
- Si passa troppo tempo al computer tralasciando i rapporti umani!!!
- Mi raccomando ragazzi: fatevi tutti un account su Gmail e poi registratevi su www.lamiaclasseduepuntozero.it
- E' inutile studiare: tanto gli studenti ne sanno sempre più di noi, loro sono nativi
- Probabilmente anche tu hai già utilizzato strategie tipiche del metodo computazionale ma non ne sei consapevole
- Ma vuoi mettere il Pascal? Ai miei tempi si che si facevano cose serie coi computer
- A che serve? I computer sono stupidi, Wikipedia è piena di fatti ma bisogna insegnare la capacità critiche



"Programma il futuro"

- I partner: CINI (ente attuatore), Confindustria Digitale, Interactive Advertising Bureau
- Gli sponsor: TIM, Microsoft, Facebook, Engineering, CISCO, Samsung, ... (solo alcuni per il 2º anno)
- La comunicazione: il sito del progetto, la pagina FB, il concorso Codi-Amo, lo stream dalla Camera
- L'ambiente e i contenuti: Code.org
- Le modalità: L'ora del codice (anche unplugged)
- Il linguaggio: Scratch

Le novità di PIF

- Si parla di costruzione di artefatti digitali (e non di studio o di uso)
- Si parte dal livello più basso della piramide, cioè dalla programmazione (ma anche dall'hardware)
- Si parte dalla scuola primaria
- In collegamento a un movimento europeo e mondiale
- Non è un percorso affidato ad alcuni specialisti, ma a docenti di ogni materia, a volontari e a sponsor

Due piani diversi

Culturale: "[...] l'obiettivo non è quello di far diventare tutti dei programmatori informatici, ma di diffondere conoscenze scientifiche di base per la comprensione della società moderna".

Metacognitivo: "[...] la conoscenza dei concetti fondamentali dell'informatica aiuta a sviluppare la capacità di risoluzione di problemi e la creatività".



Carlo Batini, 1984

Su questo aspetto si innesca una importante questione, con cui vorremmo concludere il libro: quale cultura informatica serve all'uomo d'oggi? Forse mai come per altre discipline c'è il rischio che l'informatica venga culturalmente ridotta a un fatto consumistico, e venga insegnata, consapevolmente o no, per creare nuova domanda di nuovi consumi. In questo libro ho cercato di tracciare una via diversa, provando a mostrare alcuni concetti che l'informatica può fornire a tutti noi, concetti che possono già diventare, in varie forme, cultura di tutti.



Alfonso D'Ambrosio, 2016

"Allo stesso modo il coding non è solo pensiero computazionale, un algoritmo chiuso che ci aiuta a risolvere problemi, problemi a cui i ragazzi sono abituati leggendo anche la settimana enigmistica.

Il coding diventa pensiero divergente e creativo per fare storytelling, per creare giochi didattici disciplinari, per costruire modelli teorici, e, nel caso della scienza, leggi fisiche, dotandosi di sensori esterni."



Nathan Pickels, 2016

"Ritengo che l'insegnamento di [...] sia importante in quanto offre l'opportunità di pensare a un livello più profondo di quello della maggior parte delle altre materie scolastiche. Con ciò intendo che, mentre le altre materie sono solitamente capaci di sviluppare competenze nel loro specifico ambito, la [...] consente una visione più universale che ci permette di chiedere non solo "è giusto?", ma anche "che cosa significa?""

Il Coding tra i Curricoli Digitali

26 Settembre 2016: 4,3 M€ alle reti di scuole per produrre 25 curricoli su 10 tematiche:

- Fondamentali:
 - diritti in internet ve @ steps
 - educazione ai media (e ai social)
 - educazione all'informazione
- Caratterizzanti:
 - STEM (competenze digitali per robotica educativa, making e stampa 3D, internet delle cose)
 - big e open data
 - coding
 - arte e cultura digitale
 - educazione alla lettura e alla scrittura in ambienti digitali
 - economia digitale
 - imprenditorialità digitale

I contenuti del C.D. "Coding"

"Educazione e sviluppo del pensiero computazionale sia tramite attività unplugged (senza calcolatore) sia tramite linguaggi di programmazione visuali (scuola primaria) [...].

- coding by gaming, percorsi di apprendimento condivisi in classe; uso di strumenti di coding by gaming online;
- competenze computazionali di base; il codice binario; identificare e scrivere istruzioni sequenziali;
- esecuzione di sequenze di istruzioni elementari; programmazione visuale a blocchi;
- capire lo sviluppo e l'utilizzo di strumenti informatici per la risoluzione di problemi;
- calcolare espressioni logiche con gli operatori AND e OR; riconoscere nel procedimento di soluzione algoritmica di un problema gli elementi strutturali fondamentali: sequenza, scelta condizionata, iterazione;
- conoscere e saper applicare nella vita quotidiana metodologie di ricerca sequenziale, dicotomica e hash e comprenderne i limiti di applicazione e il grado di efficienza; conoscere le strategie per l'ordinamento di oggetti (selezione/inserimento, partizionamento) e comprendere, in modo intuitive, l'efficienza della strategia adottata;
- saper eseguire semplici algoritmi su grafi di ridotte dimensione, quali la ricerca di cammini
 e di matching;
- saper rappresentare i dati o i risultati di un problema mediante l'uso di tabelle, alberi o grafi; oggetti programmabili; verifica e correzione del codice;
- conoscere il concetto di ipertesto, il suo ruolo nel world wide web, e la struttura a rete di calcolatori su cui esso è basato"



Da un lato le "educazioni a...", dall'altro il coding.

Così si introducono già nei temi e nei curricoli le separazioni che poi è difficile recuperare:

- Tra riflessione teorica e attività tecnica
- Tra aspetti etici (diritti, doveri) e pratici
- Tra apprendimento cognitivo e affettività



Coding with love

- In generale, il rapporto affettivo con le macchine digitali (computer, tablet, robot) e con gli artefatti digitali (programmi) è davvero poco preso in considerazione.
- Ne può essere esagerata l'importanza (Papert) o può essere stigmatizzato come simulacro dei rapporti con le persone (Turkle)
- Ma è questo che fa funzionare il coding (e che fa amare il proprio lavoro).
- Non è detto che docenti e studenti (e famiglie) condividano questa tonalità affettiva particolare



Un errore pedagogico

- 1) L'apprendimento tramite un gioco è più divertente
- 2) Bisogna insegnare il CT
- 3) Quindi facciamolo con un ambiente giocoso ("coding by gaming online") così non sembra nemmeno scuola

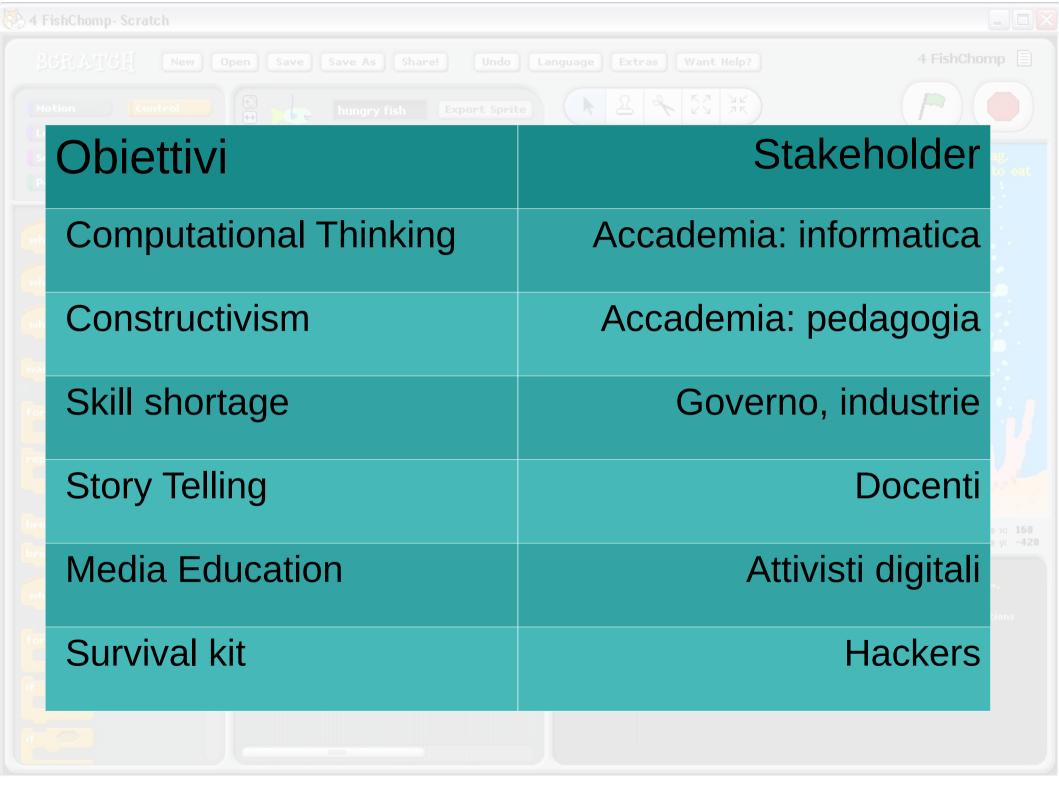
Apprendere, nel senso di procedere in un percorso di maggior comprensione del mondo, è gratificante di per sé.

Creare un (piccolo) mondo, assegnargli delle regole e modificarle, poi dargli vita e vederlo evolvere, è un modo molto efficace – e piacevole - di imparare.

Questa è la programmazione. Visuale o verbale, non fa molta differenza.

Dei ed eroi

- Paradossalmente: più si aggiungono "gadget", meno il bambino è concentrato sul processo di costruzione vera e propria del mondo artificiale
- L'identificazione con l'avatar (tartarughe o pupazzetti), tipica del videogioco, è solo uno dei possibili rapporti.
- Il programmatore, junior o meno, è il Demiurgo del suo programma, non semplicemente un eroe.



RetroCoding

- BASIC, 1964: un linguaggio semplice adatto a tutti
- Logo, 1967: un modello di interazione pensato per l'apprendimento
- SmallTalk, 1971: un ambiente grafico e una logica ad oggetti
- Pascal, 1973; il linguaggio a prova di errore
- Un buco di quasi trent'anni...
- ToonTalk, 1995: programmazione concorrente per bambini
- Squeak! /Etoys, 1996: smalltalk per bambini
- StarLogo, 2000: i blocchi verticali
- Scratch, 2003
- Snap, Kojo, PocketCode, Blockly, ...



Scratch è:

- Robot/avatar da dirigere e programmare
- Drag&drop di blocchi, scrittura
- Programmazione ad eventi
- Puzzle/Game
- Ambiente online, archivio e comunità
- Docente come tutor
- Flash/Air

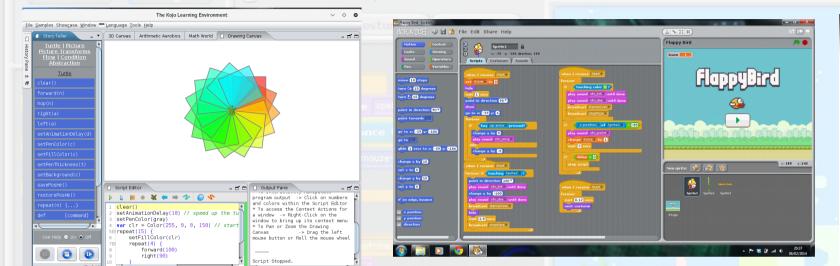


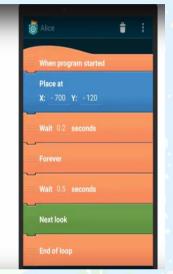
Ma ci sono alternative

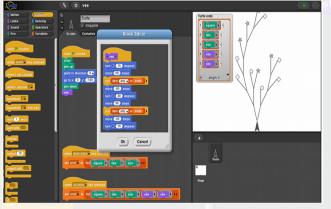
- Modelli di interazione diversi (insegnamento, dimostrazione, ...)
- Modelli didattici diversi
- Tipologie di linguaggio diverse (funzionale, a oggetti, a vincoli)
- Ambienti in cui si può scegliere l'interfaccia di interazione col codice
- Linguaggi diversi
- Licenze diverse

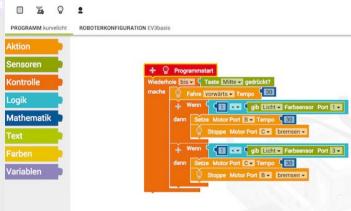


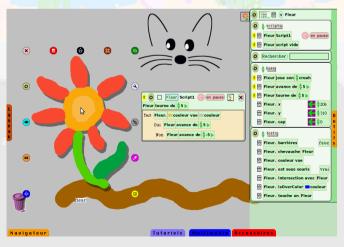












4 FishChomp- Scratch

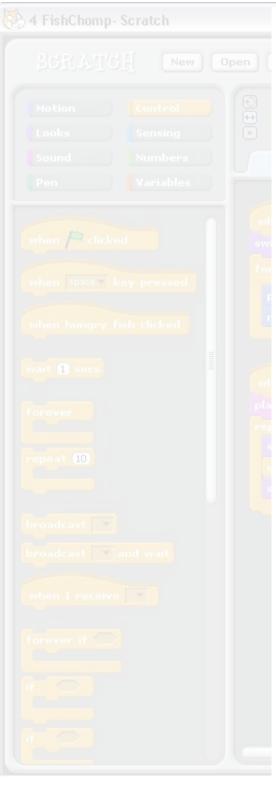
Competenze & conoscenze necessarie e sufficienti

Gestione pratica e tecnica
Incoraggiamento
Condivisione dei risultati
Buona volontà
Fiducia
Traduzioni dei tutorial

Cestione didattica
Monitoraggio e valutazione
Condivisione dei processi
Linguaggi, algoritmi
Privacy, sicurezza, licenze
Originali

Come potrebbe funzionare?

- Creando un piano didattico esteso, che comprenda diritti e doveri digitali e li colleghi al coding (i curricoli digitali?)
- Collegando il coding con le altre "materie", non alternandolo
- Adattando la proposta al contesto (culturale, didattico)
- Scegliendo il software/ambiente più adatto
- Rivalutando la componente linguistica
- Gestendo gli aspetti affettivi
- Inventando problemi e narrazioni
- + Ricerca, sperimentazione, apprendimento...



Dietro il coding



Significato, storia, obiettivi e alternative possibili

"Je suis fracas quand la foule est tranquille" (Zebda, 1998)

© Stefano Penge 2016 http://steve.lynxlab.com Creative Commons BY/SA

