



Stefano Rini
Equipe Formativa Territoriale Emilia-Romagna
Servizio Marconi TSI - USR Emilia-Romagna

Oltre la sintassi: coding come linguaggio

*Riflessioni sul coding a scuola, in generale ed in particolare in questo momento di emergenza Covid-19, a margine di un articolo di Mitchel Resnick e Natalie Rusk, **Coding at a Crossroads**, uscito il mese scorso qui*

<https://cacm.acm.org/magazines/2020/11/248219-coding-at-a-crossroads/fulltext>.

L'articolo è disponibile nella traduzione italiana di Stefano Rini (con l'apporto di Carmelo Presicce) <https://stefanorini.medium.com/il-coding-a-un-bivio-8f3ce9b6c6fe>

Mi piace cominciare a ragionare sul coding nella scuola e sulla programmazione e l'informatica (*computer science*) sostenendo che il coding sia un linguaggio nel senso più ampio del termine: non solo una sintassi e un sistema di regole codificate per programmare degli agenti automatici (un programma del computer) ma anche una rete di significati che danno senso al mondo, che aiutano il bambino a comprendere la realtà attorno a lui, oltre che a esprimere se stesso [1].

Di fronte a temi così vasti trovo una guida rassicurante al pensiero il vedere il proprio ruolo in modo 'metaforico'. Immaginarsi al di fuori del proprio campo può aiutare a considerare aspetti e punti di vista che di solito non valutiamo, aiuta a pensare e vedere i problemi secondo la chiave della leggerezza. Una visione 'educativa' sul coding non è sufficiente, come parziale risulta a volte il punto di vista dei professionisti dell'informatica o degli accademici, che sono forse troppo abituati a

ragionare partendo dalle definizioni e poi confrontando il reale a quelle definizioni che siano state avallate dalla comunità scientifica¹.

Quell'immagine che a volte ci seduce per la sua chiarezza e trasparenza, del coding come di un insieme cristallizzato di regole, di una grammatica e di una sintassi, si scontra con il mondo reale dell'informatica che è fluido sotto vari punti di vista. Prima di tutto da quello dell'hardware e dei linguaggi di programmazione. Gli standard sono in continuo aggiornamento; i linguaggi di programmazione cambiano e si avvicendano seguendo le tendenze del momento. L'hardware rende disponibili strumenti ed esperienze che erano impensabili anche solo pochi mesi prima. In un panorama in cui queste coordinate tecnologiche sono in continua evoluzione non cambia il fatto che questo complesso sistema definisca alcune competenze che devono appartenere al cittadino del XXI secolo, perché sia in grado di contribuire attivamente e in modo creativo alla comunità cui appartiene. Per farne un autore e un creatore tramite gli strumenti informatici piuttosto che un fruitore passivo (consumatore di tecnologia e device)².

Di recente l'interesse attorno al coding si è riacceso, anche per effetto dello stimolo dell'illuminante articolo di Mitchel Resnick e Natalie Rusk, che vuole fare il punto sulle pratiche e sulla riflessione attorno al coding e alla programmazione nella scuola. Sono passati dieci anni da quando Scratch è stato lanciato e tante cose sono cambiate, in Scratch [2] ma anche nel modo in cui Scratch facilita la collaborazione e la creatività di milioni di utenti, e nell'educazione per la quantità di esperienze riconducibili al coding che vengono proposte nelle scuole.

Alle radici delle idee di Resnick c'è il pensiero di Seymour Papert [3] e il costruzionismo. Secondo Papert l'apprendimento è più profondo, duraturo, significativo quando chi impara sta costruendo qualcosa di reale, un artefatto cognitivo, sia questo reale - come nelle esperienze del tinkering - o virtuale come un programma del computer. È interessante notare come il funzionamento di un'attività costruzionista, sia essa un laboratorio di scratch o un'attività di tinkering, rispecchi il funzionamento della comunità scientifica: i ricercatori - che lavorano al confine tra la scienza nota e codificata, e un campo di indagine invece nuovo - devono naturalmente lavorare in modo collaborativo, costruiscono in modo

¹ La comunità - lo si vedrà più avanti - è una parte fondamentale dell'idea di del coding come 'ambiente educativo'. Le comunità però hanno modi diversi di esprimersi e di definirsi: quando vivono il margine di ciò che è certo e indagano un sapere che viene messo alla prova o quando sono intente a consacrare la validità del proprio paradigma scientifico

² Viene detto spesso, impariamo a leggere e scrivere non per essere per forza giornalisti, scrittori o letterati; non impariamo la matematica per diventare per forza scienziati, matematici o fisici. Impariamo questi linguaggi perché sono fondamentali per poterci muovere nella nostra realtà culturale. La controparte di questa idea, dai risvolti comunque pesanti è il fatto che altri linguaggi, ritenuti meno fondamentali, vengono tenuti in secondo piano, lasciati nell'ambito dell'"espressività" come se questa fosse un campo 'minore'.

comunitario un sapere che si basa su una continua revisione delle conoscenze, su un continuo meccanismo di prova ed errore³.

Ma l'articolo ci chiede proprio di interrogarci sulla natura di queste esperienze. Come vengono proposte e quale messaggio stiamo realmente comunicando? Quali pratiche e atteggiamenti nei confronti del coding stiamo promuovendo? Stiamo favorendo l'autonomia dei bambini e dei ragazzi? Stiamo fornendo loro le occasioni necessarie per sviluppare la loro voce, renderli capaci di affermare il proprio ruolo di autori e non solo fruitori delle tecnologie che il mondo e la modernità ci mettono a disposizione? Queste domande sono fondamentali per non rischiare di sprecare il potenziale educativo di scratch e del coding in generale.

Il modello pedagogico (e epistemologico) del costruzionismo ci spiega le ragioni di come lo strumento Scratch sia stato concepito. Un luogo dove sperimentare liberamente, procedendo per prove ed errori ma sostenendo i nostri sforzi anche con l'aiuto di una comunità con la quale possiamo collaborare e che ci aiuta a pensare in modo critico e anche pratico/funzionale a quello che stiamo cercando di sviluppare.

Perso nella traduzione (Lost in traslation)

Molti termini che appartengono a queste ricerche e a questi ambiti di interesse non hanno una diretta traduzione italiana, per lo meno non ne hanno una che risulti, a chi conosce il significato dell'originale, come una traduzione veritiera capace di trasportare la complessità dei concetti e delle idee.

Spesso si dice che chi gioca con Scratch stia '**messing around**' che significherebbe letteralmente qualcosa come 'pasticciando'. In realtà per questo termine (o anche per la formula simile 'tinkering with') risulta più funzionale la traduzione con *sperimentare liberamente*.

Altro termine importante è la parola **play**. Per l'italiano *gioco* gli inglesi hanno play e game. Il termine play in questo contesto è fondamentale ed ha a che fare con quella speciale atmosfera di giocosità che si crea quando chi partecipa ad un laboratorio si sente libero di sperimentare, prendersi dei rischi, sbagliare per poi ricominciare.

Nel sopra citato articolo di Resnick e Rusk si usa il termine **computational fluency** [4]. La formula significa padronanza computazionale, fluenza (come nella linguistica) nel linguaggio del coding. La traduzione letterale, per quanto vicina

³ Anche noi stiamo provando a valorizzare questo legame tra il funzionamento della comunità di ricerca e le dinamiche di apprendimento di una comunità scolastica che sperimenta il pensiero creativo (Ricciardi, Villa, Rini, Ferrante, Anzivino *FabLearn Italy Conference Proceedings*, 2020)

all'originale non ha la stessa immediatezza e la stessa potenza, rischiando di risultare più astratta e accademica. Si è per questo preferito il termine di 'espressività computazionale' per sottolineare il valore del coding come linguaggio a tutto tondo, strumento di espressione e affermazione di sé.

Un altro termine tanto fondamentale quanto difficile da tradurre è quello di **learner**, letteralmente "apprenditore", che definisce chiunque (bambino, adulto, ricercatore) si trovi nella condizione di apprendere, imparare e sperimentare qualcosa di nuovo. Le parole che siamo costretti ad usare ci limitano ad un ambito: quello scolastico (alunni, studenti), quello informale (bambini), quando il termine originale definisce semplicemente lo stato di chi sta imparando a prescindere dal resto.

Metodologie attive

Per gli insegnanti mettere al centro il bambino, il processo educativo in generale, ridiscutere (essere disponibili a farlo) il proprio approccio didattico, sono tutti segnali di una rinnovata e forte attenzione al proprio ruolo, un ruolo forte di facilitatore dell'apprendimento, di mediatore: capace di gestire i tempi e le dinamiche dell'attività didattica senza perdere di vista il senso profondo di quello che sta avvenendo. Saper leggere e intervenire nei momenti di frustrazione e di difficoltà che sempre avvengono nei laboratori senza però dare ricette e soluzioni rapide e immediate.

La *playfulness* che deve essere parte di queste attività deve essere un ingrediente anche per gli educatori che sperimentano e partecipano dello stesso ambiente di apprendimento che contribuiscono a creare. Si pongono nella relazione educativa certo come docenti e facilitatori ma sono loro stessi 'apprenditori' (Learner). Anche l'insegnante è costantemente (e contestualmente al ruolo di facilitatore) in una condizione di apprendimento (giocoso) continuo [5].

Pandemia

Stiamo vivendo evidentemente un periodo di grandissima difficoltà che ha richiesto un'azione istantanea a tutti i livelli come quando - durante lo scorso anno scolastico - ci siamo trovati a dover insegnare a distanza. Anche in questo secondo anno sono evidenti molte difficoltà: assenze prolungate, quarantene, difficoltà nella condivisione e nella gestione dei materiali, difficoltà nel lavorare in gruppo.

Anche in questi mesi gli insegnanti si sono trovati di nuovo a fare i conti con piattaforme online, insegnamento sincrono e preparazione di materiali per il lavoro da casa.

Scratch per questo è uno strumento che ci offre opportunità eccezionali. E' un ambiente di apprendimento, una comunità online, che già - senza adattamenti - funziona in modo *virtuale*, dandoci delle possibilità immediate ad esempio attraverso l'uso delle classi virtuali e delle gallerie condivise.

Ogni bambino può sviluppare in modo assolutamente libero la propria idea, realizzando quell'ideale di "pareti ampie" di cui spesso ci parlano gli autori riguardo le potenzialità di scratch che sta proprio a rappresentare la grandissima varietà di

progetti che possiamo realizzare con questo strumento.⁴ Come educatori dobbiamo garantire che questo sia chiaro da subito. E lo sarà se permettiamo a ognuno di esprimere la propria creatività.

Nell'ambiente della classe virtuale che è possibile predisporre i ragazzi saranno naturalmente portati a condividere le proprie creazioni, potranno indagare e sperimentare i progetti degli altri 'scratcher', guardando il loro codice e *remixando* i loro progetti.

I bambini scelgono di dedicare il loro tempo, di impegnare tutta la loro attenzione, quando capiscono che quel 'lavoro' che stanno facendo è uno strumento che li aiuta a realizzarsi: inserendo i loro disegni, i suoni che hanno registrato, programmando in modo assolutamente personale i loro progetti.

L'articolo che ha mosso questi ragionamenti tocca tematiche importanti e soprattutto urgenti. E' il momento di scegliere una strada, di seguirla con convinzione e caparbietà. Convinti (noi insegnanti) del nostro ruolo di educatori e facilitatori.

Riferimenti

1. Resnick, M. *Mother's Day, Warrior Cats, and Digital Fluency: Stories from the Scratch Online Community*, Constructionism 2012, Athens, Greece.
2. <http://scratch.mit.edu/>
3. Papert, S (1980) *Mindstorms: Children, Computers, And Powerful Ideas* Basic Books. In italiano *Mindstorms. Bambini computers e creatività*, Emme edizioni (1984).
4. Resnick, M. (2001) *Closing the Fluency Gap*, Communications of the ACM, vol. 44, no. 3 <https://web.media.mit.edu/~mres/papers/cacm-3-01.htm>
5. Resnick, M. (2017) *Lifelong Kindergarten: Cultivating Creativity through Projects, Passion, Peers, and Play*. MIT Press. In italiano *Come i Bambini*, Erickson (2018).

4

https://en.scratch-wiki.info/wiki/Project_Types#:~:text=On%20the%20Scratch%20website%2C%20the%20music%2C%20art%2C%20and%20stories.