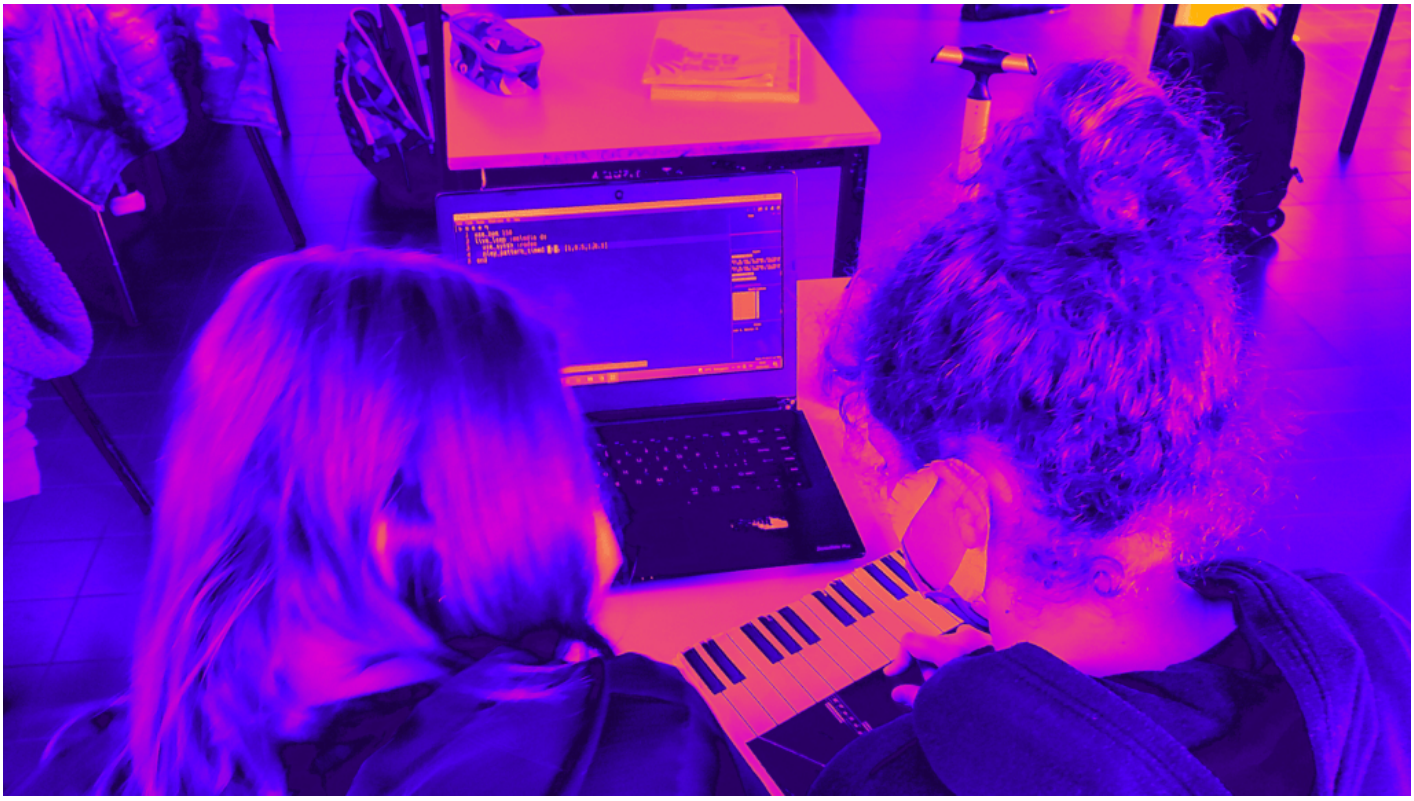
musicheria
suoni relazioni educazione

https://www.musicheria.net/2023/05/07/coded-by-me-pensiero-computazionale-ed-educazione-musicale/

#Coded by me. Pensiero computazionale ed educazione musicale

7 Maggio 2023



Premessa

In questo intervento (tenuto per la tavola rotonda “Le tecnologie digitali nella didattica musicale” organizzato dalla SIEM Bologna il 15/4/2023 per Eufonica 2023) intendo anzitutto discutere le ragioni per le quali ritengo che il pensiero computazionale sia un tema di cui l’educazione musicale debba occuparsi. Partirò da tre discorsi diffusi esplicitamente o implicitamente nelle pubblicazioni e, soprattutto, negli scambi di idee tra colleghi a cui ho avuto occasione di partecipare. Tali discorsi rappresentano secondo me tre equivoci il cui scioglimento può aiutare far chiarezza sulle questioni in gioco. Nella seconda parte dell’intervento porterò alcuni esempi di attività didattiche da me sviluppate in classe nelle scuole secondarie di primo grado.

*(NdR) In allegato (**riservato agli abbonati – visibile facendo login**) le schede della presentazione usate nella tavola rotonda.*

La normativa, l'informatica e la programmazione

Normativa

Chi è attento al lavoro di ricerca e legislativo delle istituzioni europee e nazionali sa bene che l'invito a sviluppare percorsi didattici sulle competenze digitali è sempre più pressante per la scuola (cfr. documenti con il *DigComp Edu*). In particolare, l'espressione "pensiero computazionale" è diventata una sorta di prezzemolo buono per condire qualsiasi decreto o circolare e la prospettiva è ormai quella di introdurre percorsi didattici sul pensiero computazionale fin dalla scuola d'infanzia. Da ciò è scaturito un dibattito simile a quello sull'Educazione Civica che colpisce soprattutto le scuole secondarie, dove l'insegnamento è strutturato per discipline: in quali ore si deve insegnare il pensiero computazionale? Chi deve insegnarlo?
Questo modo per porre la questione parte però da una premessa fuorviante nella misura in cui il pensiero computazionale non è una disciplina scolastica, ma un modo di risolvere problemi e di ragionare che è trasversale alle discipline.
Ma cos'è esattamente il pensiero computazionale[1]? Il pensiero computazionale è un modo di ragionare, un approccio alla soluzione dei problemi che può essere rilevato in molti campi del sapere fin dall'antichità e che oggi, con l'avvento dei computer, trova applicazioni sempre più sofisticate. Si tratta di un modo di ragionare che smonta e scompone i problemi nelle loro unità minime per poi elaborare strategie formalizzate e strutturate per risolverli.

Informatica

Come ho detto, tra pensiero computazionale e computer c'è un evidente legame, e questo ci porta al secondo equivoco che è alla base di un sentire comune tra gli educatori di area umanistica e artistica: il pensiero computazionale riguarda l'informatica e quindi attiene alle materie di area STEM (Science, Technology, Engineering, Mathematics) . Quello di considerare un tema non pertinente alla propria area disciplinare è un atteggiamento comune quando un insegnante si trova davanti a tecnologie e attività dove il legame con lo specifico dei contenuti della propria materia non è evidente o, perlomeno, non è parte della tradizione didattica consolidata.
Associare in modo univoco pensiero computazionale ed informatica restituisce però una visione limitata di un quadro molto più complesso. Il pensiero computazionale è – appunto – un *pensiero*, un modo di ragionare, e come tale è utile per affrontare la realtà, qualsiasi realtà, non solo quella di ambito scientifico e matematico. Inoltre, è un modo di pensare che precede l'avvento dei computer.

Programmazione

Recentemente le scuole hanno cominciato a inserire nelle proprie programmazioni (didattiche) attività di programmazione (informatica), spesso in prospettiva trasversale alle discipline. Tali attività sono abitualmente denominate *coding*, termine con il quale si indica la scrittura del codice destinato ad essere eseguito da un computer, vale a dire – appunto – l'attività di programmazione. Anche il concetto di programmazione si presta però ad equivoci. Noi siamo abituati a pensare alla programmazione come un'attività volta a rendere più veloci ed efficienti procedure che hanno uno scopo pratico e immediato, utile nella vita quotidiana (localizzare un indirizzo, inviare una email, gestire sistemi di domotica) o nelle attività professionali (implementare processi di produzione, di

automazione e di robotica industriale). In questo senso ci si potrebbe chiedere cosa c'entri tutto questo con il mondo dell'arte.
Il quadro cambia se usciamo dai luoghi comuni e assumiamo quella prospettiva che da Seymour Papert a Mitchel Resnick è molto diffusa anche in pedagogia[2]. Qui la programmazione non è semplicemente un modo per rendere veloci ed efficienti alcune pratiche di produzione industriale e alcune attività quotidiane, ma è una modalità espressiva, una lingua, e, come tale, può essere usata anche per esprimersi in qualsiasi settore, anche in quello artistico. Anzi, di fatto la programmazione è usata ormai da vari anni nelle pratiche artistiche, anche musicali. In questo senso, dunque, la programmazione ha direttamente a che fare con i contenuti specifici dell'educazione musicale tanto quanto la notazione musicale o l'esecuzione strumentale.

Coding espressivo

Intendiamoci, ci sono tanti modi di introdurre attività di programmazione in educazione musicale. Il *coding* con i suoni è infatti una tecnica che l'insegnante può utilizzare nelle sue lezioni per molteplici finalità didattiche, anche, ad esempio, per studiare la teoria musicale usando un mezzo diverso dal solito.
Per quanto mi riguarda, la scoperta del *coding* sonoro mi ha fornito l'occasione per integrare e potenziare i progetti didattici in un ambito dell'educazione musicale che ritengo prioritario: quello delle attività creative e di invenzione musicale. Per questo mi piace parlare di "coding espressivo", che vuole essere un modo per indicare quei progetti didattici dove le potenzialità educative del pensiero computazionale e quelle delle attività di espressione artistica convergono per avviare percorsi interdisciplinari quanto mai attuali, visto le premesse da cui siamo partiti.
L'espressione "coding espressivo" così intesa è un'idea nata da una serie di riflessioni che abbiamo sviluppato io e il collega Leo Izzo in un recente articolo pubblicato da *Musica Domani*, a cui rimando[3]. Volendo accennare ad un orientamento teorico, dirò solo che nell'espressione "coding espressivo" vorrei far convergere da un lato la prospettiva costruzionista già menzionata di Papert e Resnick e, dall'altro, una tradizione di studi di pedagogia musicale che vedo ben incarnata nella recente proposta del Centro Studi Maurizio di Benedetto di un "Laboratorio di Pedagogia dell'Invenzione Musicale"[4].
Ma non è questo il momento per approfondire questi aspetti teorici. Come annunciato, vorrei ora passare ad illustrare alcune attività che sto sviluppando nelle mie classi.

Risorse e attività

Scratch: imparare a programmare con i blocchi

Un ambiente di programmazione molto famoso a scuola, utilizzato da docenti di varie discipline, è Scratch, creato nel 2006 da Mitchel Resnick presso il MIT di Boston[5]. Il linguaggio di programmazione di Scratch è grafico, a blocchi di comando. Questo lo rende particolarmente intuitivo e immediato, dunque adatto anche ai bambini delle primarie.
Non essendo una risorsa pensata specificamente per la musica, le possibili attività di produzione musicale sono però limitate. Per questo, uso Scratch soprattutto per introdurre gli allievi ai concetti di base della programmazione musicale per poi passare a risorse più ricche dal punto di vista musicale, ma anche più complesse.
Per quanto riguarda l'approccio didattico adottato, lo definirei "capovolto": in una prima fase, in classe, si imparano i comandi principali componendo musica in modo condiviso, in plenaria, e ci si esercita individualmente o in coppia; in seguito, gli allievi visionano a casa dei tutorial prodotti

dall'insegnante per poi a scuola svolgere principalmente attività laboratoriale di composizione, naturalmente sotto la guida dell'insegnante.
Nella presentazione ho inserito il link alla pagina web che uso per l'attività didattica, dove è possibile consultare alcuni dei materiali messi a disposizione degli allievi[6]. Inoltre, nella presentazione ho anche inserito un link ad un lavoro di programmazione sviluppato nel febbraio del presente anno scolastico da un bambino di prima, quindi davvero alle prime armi non solo con il *coding*, ma anche con l'alfabetizzazione musicale[7]. Si tratta di un breve evento musicale che il ragazzo ha composto pensando al campo di basket. A me piace pensare a questi brevi eventi musicali come degli *Ètude*, brevi frammenti musicali pensati per sperimentare ed esercitare tecnica e inventiva.
Per la CodeWeek[8] dell'anno scolastico 2022-2023 ho invece sviluppato un breve progetto interdisciplinare musica-tecnologia con la prof.ssa Simona Catalani con una classe terza che non aveva mai fatto attività di coding: abbiamo proposto un percorso di circa 5 lezioni che è sfociato in un incontro dove la classe, suddivisa in coppie, ha realizzato progetti grafici-sonori sulla base di una serie di istruzioni. Tali istruzioni sono consultabili da un link in presentazione[9], dove è anche possibile trovare anche un secondo link che porta ad un video che documenta uno dei lavori realizzati dagli studenti[10].

Sonic Pi: software professionale per piccoli live coder

Come dicevo, sto usando Scratch solo per brevi attività circoscritte volte ad introdurre i principi della programmazione musicale per poi passare al software Sonic Pi.
Sonic Pi è un software sviluppato con obiettivi didattici presso l'Università di Cambridge in collaborazione con la Raspberry Pi Foundation[11]. Attualmente è un software diffuso sia tra i musicisti professionisti, sia nelle scuole: il suo creatore, Sam Aaron, continua infatti a sviluppare Sonic Pi senza dimenticare l'originale vocazione didattica della sua creazione. Proprio per questo Sonic Pi è uno strumento ideale per la scuola: permette ai ragazzi di usare gratuitamente una risorsa autentica specificamente progettata per la musica che allo stesso tempo presenta caratteristiche di accessibilità.
Sonic Pi rimane comunque un software relativamente complesso. Anzitutto è pensato per il *live coding*, vale a dire non tanto per la composizione, quanto per la performance musicale. Inoltre, il linguaggio di programmazione non è a blocchi, ma a righe di testo. Sonic Pi prevede dunque una maggiore capacità di astrazione rispetto a Scratch, cosa che rende il suo uso adatto soprattutto per le scuole secondarie e per gli studi superiori.
Personalmente sto sperimentando Sonic Pi nelle classi delle scuole secondarie di primo grado con soddisfazione. L'approccio didattico non è diverso da quello che utilizzo per Scratch, come credo risulti evidente dalla pagina web dove ho raccolto alcuni dei materiali che uso a scuola (link nella presentazione[12]). Da questa pagina web è possibile anche raggiungere l'unità didattica *Sound Code Club. Coding espressivo con i suoni*, da me preparata per il sito INDIRE dedicata alle attività di educazione artistica. Questa unità raccoglie le prime attività da me svolte quando avvio un percorso didattico sul coding musicale espressivo usando Sonic Pi.

Coded by 1A

Rimaniamo su Sonic Pi. *Coded by 1A* è un video che raccoglie gli esiti di un percorso didattico durato circa un quadrimestre che ho sviluppato in una classe prima durante l'anno scolastico 2022-23[13]. Si tratta di una serie di *Ètude* sviluppati individualmente dagli alunni. Questo lavoro rappresenta il mio contributo al gruppo di ricerca sul coding espressivo promosso dall'INDIRE e

coordinato da Giovanni Nulli a cui ho partecipato lo scorso anno scolastico.

Non c'è problema

L'ultimo esempio è quello a cui tengo maggiormente: si tratta della documentazione video dello spettacolo di teatro *Non c'è problema* (che purtroppo per vari motivi abbiamo potuto rappresentare solo in classe di fronte a colleghi e altre classi, come l'ambientazione mostra chiaramente)[14]. Considero questo lavoro significativo perché permette di osservare non solo l'attività di programmazione nella dimensione performativa, ma anche l'integrazione tra live coding e strumenti tradizionali, a dimostrazione del fatto che l'uno non esclude gli altri. Più in generale, il video evidenzia le potenzialità didattiche del lavoro interdisciplinare in chiave espressiva. Lo spettacolo è infatti frutto della collaborazione tra il sottoscritto e la docente di lettere esperta in teatro educativo Diana Ligue Denis: due (lunghi) laboratori (teatro e musica) si sono intrecciati nelle ore curricolari per dare voce, parole, suoni e movimenti all'universo affettivo di una classe seconda, usando vari mezzi, digitali e non, in modo del tutto integrato e trasparente.
A conclusione del mio intervento, spero di essere riuscito a dare esempi concreti di come le riflessioni della prima parte del mio intervento possano concretizzarsi in classe.

[1] Cfr. Peter J. Denning & Matti Tedrey, *Computational Thinking*, Cambridge (MA), The MIT Press, 2019.

[2] Cfr. Seymour Papert, *Mindstorms. Children, Computers, and Powerful Ideas*, New York, Basic Books, 1993 (pr. ed. Brighton, Harvester, 1980); Mitchell Resnik, *Lifelong Kindergarten: Cultivating Creativity through Projects, Passion, Peers, and Play*, Cambridge (MA), MIT Press, 2011; tr. it. *Come i bambini*, Trento, Erickson, 2018.

[3] Roberto Agostini e Leo Izzo, "Coding the beats. Coding espressivo con algoritmi sonori", *Musica Domani*, 185, 2021, pp. pp. 40-52.

[4] "Laboratorio di Pedagogia dell'Invenzione Musicale (LaPIM)", Musicheria, 17 Marzo 2022, https://www.musicheria.net/2022/03/17/laboratorio-di-pedagogia-dell-invenzione-musicale-lapim/.

[5] https://scratch.mit.edu/.

[6] https://sites.google.com/ic7bo.istruzioneer.it/agostini/coding-con-i-suoni/scratch.

[7] https://youtu.be/733_n90iPoQ.

[8] La CodeWeek è la Settimana Europea del Codice, cfr. https://codeweek.eu/.

[9] https://docs.google.com/document/d/1IcjDaHMKy5z4heATnQIouzhKWh-eHrbxYwNiTKV4ImE/edit?usp=sharing.

[10] https://youtu.be/HWrbuSxT9XI.

[11] Sonic Pi è scaricabile gratuitamente al sito ufficiale https://sonic-pi.net/. Su *Musicheria* abbiamo già parlato di Sonic Pi: cfr. Roberto Agostini (2020), "Rock around Sonic Pi. Sam Aaron, il live coding e l'educazione musicale", Musicheria, 26 aprile 2020,

https://www.musicheria.net/rubriche/studi-e-ricerche/5243-rock-around-sonic-pi.

[12] https://sites.google.com/ic7bo.istruzioneer.it/agostini/coding-con-i-suoni/sonic-pi.

[13] https://youtu.be/BMSAeqyM124.

[14] Il video non può essere visionato online. Può essere solo scaricato dalla pagina web seguente: https://www.ic9bo.edu.it/pagine/teatro-in-classe—-non-c-problema.

Roberto Agostini