

Sinfonie di luce: dipingere il cielo con il codice

a cura di:

Maddalena Conte, Emilia Sera



Body Coding, Arte Generativa, Debugging Creativo

Introduzione

C'è un libro che non si legge: si guarda, si ascolta, se ne percepisce il ritmo. *Night Shells* di Jinta Hirayama è un albo illustrato in cui i fuochi d'artificio esplodono in una notte silenziosa, uno dopo l'altro, con la precisione rituale di un evento atteso. Ogni tavola è un istante catturato: la miccia che s'accende, il fuoco che sale, l'esplosione, il brillio, la scomparsa. Una sequenza. Un algoritmo.

È stato questo a suggerire il percorso: la struttura interna di quelle immagini, così narrativa e ordinata, ha offerto il sostegno ideale per introdurre il linguaggio del pensiero computazionale. Questa intuizione è nata e si è sviluppato all'interno della cornice del Progetto Marian Centro Zaffiria, una sperimentazione che ha esplorato l'intersezione tra linguaggi artistici, intelligenza artificiale e didattica inclusiva.

Il laboratorio online che presentiamo in questo articolo è stato condotto con classi di scuola dell'infanzia, prima e seconda primaria in una prima versione, e con classi di terza, quarta e quinta in una seconda, più articolata. Due percorsi distinti nella forma, ma accomunati da un'intuizione: quella in cui coding, matematica e arte si fondono in un unico ecosistema creativo in grado di generare linguaggi interconnessi con cui decodificare e ricreare il mondo.

Ogni fuoco è un programma

Il laboratorio si è svolto in formato online, in una sessione continua di tre ore. La scelta di coinvolgere contemporaneamente alunni di ordini e classi così diverse — dall'ultimo anno di scuola dell'infanzia alla quinta primaria — ha richiesto una progettazione attenta e duale: stesso tema, stesso protagonista artistico, ma materiali, linguaggi, obiettivi e complessità profondamente differenziati.

Non si è trattato di semplificare per i più piccoli e complicare per i grandi: si è trattato di trovare, per ciascuna fascia, il punto di ingresso più autentico. Per i bambini dell'infanzia e delle prime classi, quel punto è stato il corpo, il gioco, la narrazione gestuale. Per gli alunni di terza, quarta e quinta, è stata la programmazione come linguaggio formale con la sua grammatica, la sua coerenza interna, la sua capacità di produrre risultati visibili e modificabili.

Cominciare con Hirayama ha significato offrire agli alunni un ingresso estetico al pensiero computazionale. Prima ancora di nominare concetti come sequenza, loop o condizione, si è guardato insieme. Ci si è interrogati su cosa vedessero in quelle tavole, su come fossero fatti quei fuochi, su cosa succederebbe se si accendesse la miccia prima di tutto il resto, o se si saltasse un passo.

Il tutto accompagnato da una molteplicità di strumenti che ha incluso flashcard digitali, un Coding Lab a moduli sviluppato con Claude, la piattaforma di animazione Eatmy.art e l'ambiente di programmazione a blocchi Scratch.

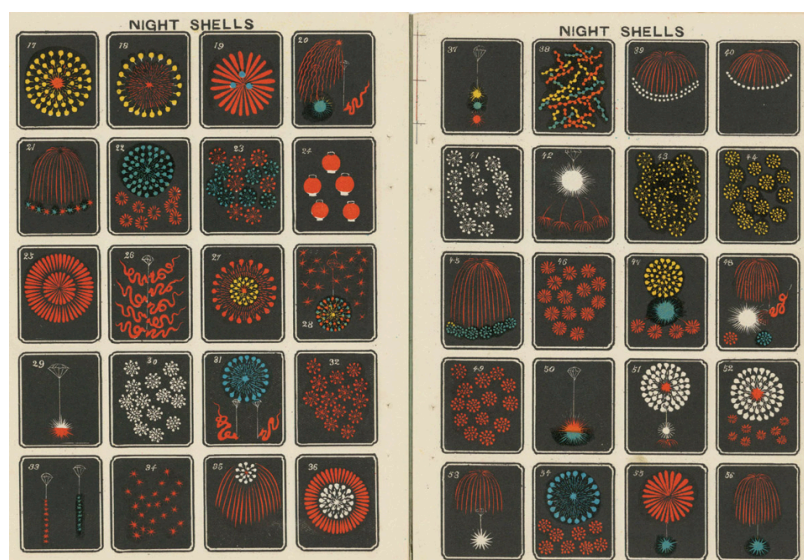


Figura 1 - Fuochi d'artificio Hirayama

Primo binario: infanzia e classi 1^a–2^a primaria

Il corpo come primo codice

Per i più piccoli, l'algoritmo non è stato un concetto da comprendere, ma un'esperienza da abitare con il corpo. Il punto di partenza è stato lo stesso per tutti: le tavole di Hirayama, osservate insieme, scomposte, interrogate. Ai bambini è stato chiesto di fare quello che i progettisti di fuochi d'artificio fanno da sempre, smontare l'evento visivo nei suoi elementi costitutivi.

Abbiamo quindi isolato i momenti chiave, trasformandoli in una sintassi gestuale:

- **L'evento di inizio (Input)-** *Accensione della miccia.* Nel codice diventerà "quando si clicca sulla bandierina verde". Per i bambini è stato il contatto fisico: il dito che tocca terra, attivando l'energia del sistema.
- **La scansione temporale (Wait)-** *L'attesa di 3 secondi.* Uno dei concetti più difficili da programmare è il silenzio o la pausa. Contando ad alta voce "1... 2... 3...", i bambini hanno interiorizzato che il computer, come il fuoco d'artificio, ha bisogno di tempi di esecuzione definiti.
- **Il vettore e la direzione (Motion)-** *La salita.* Alzarsi lentamente sulle punte ha permesso di percepire il movimento lungo l'asse verticale, preparando il terreno alla futura comprensione delle coordinate.
- **L'output visivo e sonoro-** *Esplosione, brillio e scomparsa.* Qui la narrazione si è fatta esplosiva: aprire le braccia, agitare le mani e infine rimpicciolirsi fino a sparire (abbassandosi).

Il corpo diventa il computer. La sequenza dei gesti è il programma. Eseguirlo significa incarnare, letteralmente, la logica di un algoritmo. Questo passaggio di *body coding* è stato fondamentale perché ha permesso ai bambini di essere, contemporaneamente, programmatori ed esecutori del proprio algoritmo: di capire dall'interno cosa significhi dare istruzioni precise a un sistema. La memorizzazione della sequenza fisica ha abbattuto il carico cognitivo necessario nella fase successiva, rendendo naturale il passaggio ai blocchi di codice.



Figura 2 - Flashcard digitali

Il disegno che prende vita

La successiva tappa del percorso per i più piccoli ha portato a **eatmy.art**, una piattaforma gratuita e straordinariamente accessibile che permette di animare disegni fatti a mano grazie all'intelligenza artificiale. Il nome originale del sito "*Mangiatore d'arte*" ha già in sé qualcosa di fiabesco, e l'esperienza di caricare un disegno e vederlo muoversi sullo schermo lo è altrettanto.

Per questa fascia d'età, il flusso di lavoro ha previsto uno storyboard semplice su tre fotogrammi: la miccia che brucia, il fuoco che sale, l'esplosione. Tre momenti chiave dell'algoritmo, tradotti in tre disegni a mano, caricati in sequenza sulla piattaforma. Il risultato è un'animazione breve, a volte imprecisa, sempre sorprendente: il proprio disegno che si muove, che *vive*.

Ciò che rende questo passaggio didatticamente prezioso va oltre la "magia" del risultato. Senza ricorrere a tecnicismi, abbiamo introdotto un pilastro della cultura digitale: **l'animazione come sequenza di immagini statiche**, il movimento come effetto emergente dalla giustapposizione di fotogrammi. È, a tutti gli effetti, un'altra forma di algoritmo: un programma visivo e pittorico che rende tangibile il legame tra l'idea creativa e la sua esecuzione automatizzata.

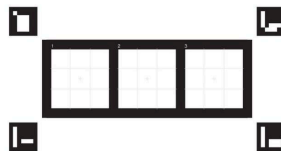


Figura 3- eatmy.art

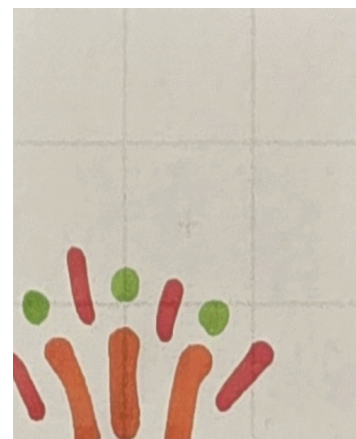


Figura 4- Fuochi d'artificio gif

Il Coding Lab

L'ultimo approdo del percorso per i più piccoli è stato il **"Fuochi d'Artificio — Coding Lab"** <https://bit.ly/49GGNNk>, un ambiente interattivo a moduli sviluppato ad hoc con Claude. Qui la narrazione ha lasciato il posto alla prima forma di astrazione tecnica. L'interfaccia si presenta chiara e simbolica, divisa in quattro aree tematiche: *Variabili*, *Loop*, *Condizione* e *Sequenza*.

Per le classi prime e seconde, l'esplorazione si è concentrata sul modulo **Sequenza**. La sfida? Programmare una serie di tre fuochi d'artificio, definendo per ognuno parametri precisi: forma, colore e dimensione. Mentre i bambini compongono la propria sequenza, il sistema restituisce un feedback prezioso: nella parte inferiore dello schermo, il codice generato appare in tempo reale in un riquadro che mostra la sintassi nel "linguaggio della macchina".

Questo rappresenta il primo vero incontro con il codice testuale. La forza didattica risiede nell'**immediatezza del risultato**: ogni modifica ai parametri produce un effetto visivo istantaneo sullo sfondo scuro dello schermo. Non è una programmazione astratta; è un dialogo in cui l'alunno interroga la macchina e riceve una risposta visiva coerente. Questo trasforma il coding da esercizio logico a strumento di potere creativo, rendendo l'esperienza concreta e profondamente motivante.

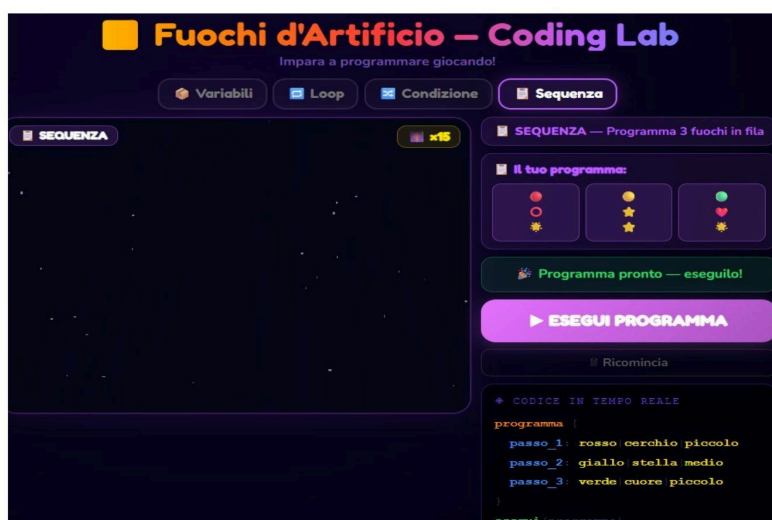


Figura 5- Coding Lab

Il secondo binario: classi 3^a, 4^a e 5^a primaria

Uno sguardo più profondo

Se per i più piccoli il fulcro è stata l'azione, per gli alunni più grandi la sfida si è spostata sulla **formalizzazione**. Anche per le classi superiori il laboratorio ha preso avvio dalle tavole di Hirayama, ma la domanda che ha accompagnato quella visione è stata diversa.

Non ci siamo chiesti solo come funzioni un fuoco, ma: *come possiamo descriverlo con un linguaggio preciso?* Come cambiano forma, dimensione e colore da un fuoco all'altro? Quali sono le variabili in gioco? Cosa succederebbe se volessimo riprodurre lo stesso fuoco cento volte, o se volessimo cambiare solo il colore lasciando tutto il resto invariato?

Questi interrogativi appartengono contemporaneamente alla matematica e alla programmazione e hanno trovato nella struttura del laboratorio una risposta concreta, sperimentabile e modificabile in tempo reale.

Il Coding Lab completo: variabili, loop, condizioni

Per le classi 3^a, 4^a e 5^a, il Coding Lab <https://bit.ly/49GGNNk> è stato esplorato nella sua interezza. Ogni modulo ha aperto una conversazione diversa.

Il modulo **Variabili** ha introdotto la possibilità di assegnare un nome simbolico a un valore — colore, forma, dimensione — e di riutilizzarlo più volte nel programma senza doverlo riscrivere ogni volta. È un concetto apparentemente tecnico che rivela, a uno sguardo matematico, qualcosa di familiare: la variabile come generalizzazione, come segnaposto per una classe di valori possibili.

Il modulo **Loop** ha affrontato la ripetizione come struttura: non riscrivere la stessa istruzione dieci volte, ma dire *ripeti dieci volte* e lasciare che la macchina lo faccia. È un'economia del pensiero che assomiglia, nei suoi fondamenti logici, alla moltiplicazione: invece di sommare lo stesso numero più volte, si dà un nome a quella ripetizione. Il parallelismo non è forzato, è reale, e gli alunni di quarta e quinta lo colgono con facilità una volta che viene loro mostrato..

Il modulo **Condizione** ha introdotto la logica decisionale: *se* una certa condizione è vera, *allora* esegui questa istruzione; *altrimenti*, esegui un'altra. Il fuoco cambia comportamento a seconda di ciò che trova, un operatore binario che sta alla base tanto della logica formale quanto di molte situazioni della vita quotidiana. Lavorare su questo modulo ha aperto conversazioni inaspettate sulla previsione, sul caso, sull'idea che un programma non sia necessariamente deterministico.

Il catalogo dei fuochi d'artificio con Scratch

Il percorso è culminato nella programmazione su Scratch, dove la libertà creativa si è scontrata e fusa con il rigore del codice. Qui, la teoria delle variabili si è tradotta in blocchi colorati, e le tavole di Hirayama sono state ricostruite non come copie, ma come sistemi dinamici e interattivi.

Lo sprite ruota, si sposta, lascia una traccia con la penna e se il loop è abbastanza lungo, se l'angolo di rotazione è scelto con cura, sulla tela digitale compare una figura radiale e simmetrica.

Il cuore del programma si è articolato in due script principali. Il primo ha governato lo sprite originale:

- quando si clicca sulla bandiera verde
 - o cancella tutto cancella lo sfondo.
 - o per sempre
 - ruota di 15 gradi: lo sprite ruota di 15° ad ogni ciclo.
 - stampa: disegna l'immagine corrente sullo sfondo.
 - crea un clone di me stesso: crea un clone che utilizzerà lo script successivo.

Effetto: lo sprite ruota e "stampa" copie di se stesso, creando una figura radiale; ogni clone poi segue la propria strada.



Figura 6- Scratch prima parte

Il secondo ha governato il comportamento di ogni singolo clone generato:

- quando inizio come clone
 - o ripeti 150
 - muovi di 10 passi: il clone si allontana dal centro.
 - modifica l'effetto fantasma di 2: diventa gradualmente più trasparente.
 - ruota di 3 gradi: curva leggermente.
 - modifica l'effetto colore di 4: cambia colore mentre si muove.
 - o elimina questo clone: il clone scompare.

Effetto: ogni clone si comporta come una scintilla di fuochi d'artificio che parte dal centro, si sposta verso l'esterno, cambia colore e scompare.

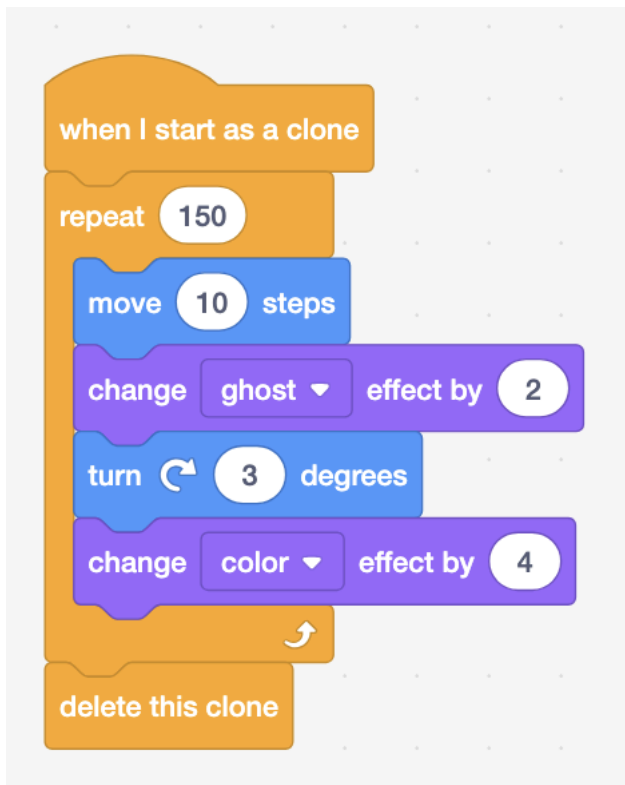


Figura 7- Scratch seconda parte

Ogni istruzione ha rivelato qualcosa. La **rotazione** come trasformazione geometrica nello spazio. L'**effetto fantasma** come dissolvenza progressiva, un valore che cresce di 1 a ogni passo, fino a rendere il clone completamente trasparente e quindi invisibile, *sparito*, proprio come nell'algoritmo originale. Il **loop con contatore** come struttura iterativa controllata, diversa dal "per sempre" perché ha un limite, una fine. La **clonazione** come meccanismo che permette a più istanze dello stesso sprite di esistere contemporaneamente e di comportarsi in modo indipendente, una forma elementare di parallelismo, concetto che nell'informatica teorica appartiene a livelli molto avanzati ma che Scratch rende visibile, gestibile e persino giocoso.

La conversazione matematica che è nata intorno a questi blocchi ha sorpreso anche noi: l'idea di una funzione (un'operazione che prende un input e produce un output prevedibile), la composizione di trasformazioni geometriche, il concetto di stato come descrizione di una situazione in un determinato momento del programma. Concetti che normalmente incontriamo in traiettorie disciplinari separate come geometria, algebra, informatica, qui si presentavano insieme, al servizio dello stesso obiettivo: far esplodere un fuoco d'artificio sullo schermo.

RIFLESSIONI E CONCLUSIONI

Al termine del laboratorio, ciò che resta non è solo una collezione di animazioni digitali o di sequenze di codice, ma la consapevolezza di aver partecipato a un processo creativo unitario.

La forza di questo percorso risiede nella sua **verticalità autentica**. Abbiamo dimostrato che è possibile parlare degli stessi concetti (sequenza, variabile, attesa) a bambini di cinque anni e a ragazzi di undici, a

patto di scegliere la "porta di ingresso" corretta. Per i più piccoli, la porta è stata il corpo e il gioco; per i più grandi, è stata la sfida logica e la padronanza della sintassi.

L'integrazione di un "arsenale" digitale così variegato (Claude, Scratch, Eatmy.art) ha permesso di:

- **abbattere l'astrazione:** il coding è diventato lo strumento per dare vita a un'emozione visiva, non un fine in sé.
- **personalizzare l'apprendimento:** ogni studente, dal più timido al più esperto, ha trovato un modo per esprimersi, che fosse attraverso un disegno animato o uno script complesso.
- **valorizzare l'errore:** nel "Coding Lab" come in Scratch, il feedback immediato ha trasformato l'intoppo in un'occasione di *debugging* creativo.

In conclusione, l'opera di Jinta Hirayama ci ha insegnato che un algoritmo può essere silenzioso e luminoso come un fuoco d'artificio. E che la scuola, quando accetta di mescolare i linguaggi dell'arte, della matematica e della tecnologia, può trasformare una lezione online in un'esperienza di scoperta che, proprio come quei fuochi, lascia una scia duratura nella mente degli alunni.



Emilia Sera

Indirizzo eMail: emilia.sera@scuola.istruzione.it

Docente di Scienze Umane e componente dell'Équipe Formativa Territoriale Lazio del Ministero dell'Istruzione e del Merito. Nell'ambito della formazione del personale scolastico ha maturato una solida esperienza nella progettazione e conduzione di percorsi di sviluppo professionale, con particolare attenzione alle metodologie attive e alla didattica laboratoriale. I suoi principali ambiti di interesse riguardano l'innovazione metodologica, le tecnologie per l'apprendimento, l'intelligenza artificiale applicata ai contesti educativi, la ricerca pedagogica e i processi di inclusione.



Maddalena Conte

Indirizzo eMail: maddalena.conte10@scuola.istruzione.it

Docente di scuola primaria e componente dell'Equipe Formativa Territoriale del Ministero dell'Istruzione per la regione Lazio. È laureata in Scienze della Formazione Primaria e in Linguistica Moderna, con abilitazione sia per posto comune che per sostegno. I suoi principali interessi di ricerca includono le metodologie didattiche innovative, le tecnologie, l'intelligenza artificiale, la ricerca educativa e l'inclusione.

Abstract

Il contributo presenta una sperimentazione didattica nata all'interno del Progetto Marian (Centro Zaffiria), che esplora l'intersezione tra linguaggi artistici e pensiero computazionale. Partendo dall'albo illustrato Night Shells di Jinta Hirayama, il laboratorio trasforma la sequenza rituale dei fuochi d'artificio in un ecosistema creativo dove arte, matematica e tecnologia convergono. Il percorso si articola su due binari paralleli: per i bambini di 5-7 anni, l'algoritmo viene esperito attraverso il corpo, la narrazione gestuale e l'animazione analogico-digitale (eatmy.art); per gli alunni di 8-11 anni, il focus si sposta sulla formalizzazione logica di variabili, loop e condizioni attraverso un Coding Lab dedicato e Scratch. L'integrazione di strumenti diversificati - dalle flashcard all'intelligenza artificiale - abbatte l'astrazione del codice, rendendolo strumento di espressione estetica. Il percorso dimostra come un approccio verticale e incarnato al pensiero computazionale favorisca un apprendimento significativo, in cui il debugging diventa riflessione creativa e l'algoritmo un linguaggio per interpretare e ricostruire il mondo.