

CodyColor KIT

Teacher trainer resource

Unplugged

Multidisciplinare

Versatile

**Per sviluppare competenze
di pensiero computazionale**



| 1. Scopri il metodo

CodyColor è un metodo di **coding unplugged** (senza dispositivi elettronici) basato su regole semplicissime, che si presta ad essere introdotto fin dalla scuola dell'infanzia, ma consente di svolgere attività sfidanti anche per la scuola secondaria e per gli adulti.

L'attività, come in molti metodi di coding, si svolge su una griglia o **scacchiera**. La dimensione standard della griglia è di 5x5 caselle.

Ogni attività di coding prevede che ci sia una netta divisione di ruoli tra **programmatore** ed **esecutore ideale**. Il programmatore scrive il programma e l'esecutore lo esegue fedelmente. Entrambi si attengono alle regole di un linguaggio di programmazione rigoroso.

L'elemento base di ogni linguaggio di programmazione è il **repertorio delle istruzioni elementari**, che stabilisce sia come debbano essere rappresentate (nei linguaggi testuali ogni istruzione è rappresentata da una parola convenzionale) sia come debbano essere eseguite (cioè a quale azione elementare corrispondano).

Nel coding su scacchiera, la dimensione delle caselle determina il passo di avanzamento, mentre l'ortogonalità determina l'angolo delle rotazioni sul posto, di 90° a destra (in senso orario) o a sinistra (in senso antiorario). Questo consente di essere rigorosi in modo intuitivo (dopo averlo chiarito una volta per tutte) senza dover specificare la lunghezza degli spostamenti e gli angoli di rotazione.

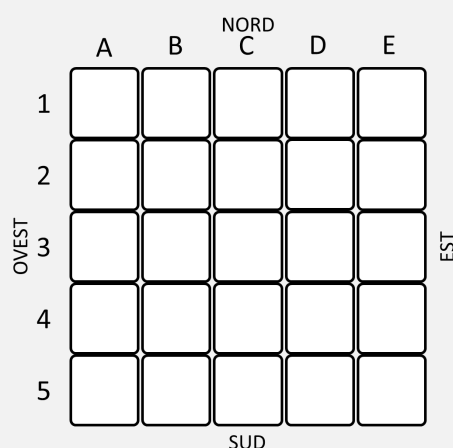
Generalmente i linguaggi di programmazione, anche unplugged, prevedono che le istruzioni siano composte in **sequenza** per formare il **programma**, che determina l'ordine in cui l'esecutore dovrà eseguirle.

Per poter eseguire un'istruzione, l'esecutore deve prima **leggerla**, cioè venirne a conoscenza (se il programmatore gli impedisse di vedere l'istruzione il robot non potrebbe eseguirla) e **interpretarla**, cioè riconoscerla come una delle istruzioni convenzionali a cui è associata una specifica azione (se il programmatore usasse istruzioni che non sono nel repertorio il robot non saprebbe cosa fare).

Poiché gli esecutori ideali non fanno nulla senza un programma, è come se le istruzioni le leggessero per istinto e se, sempre per istinto, dopo l'esecuzione di un'istruzione passassero alla successiva. Lo chiamo istinto solo per dire che sono progettati per comportarsi così, non hanno bisogno che sia il programmatore a dirgli di leggere le istruzioni, altrimenti si entrerebbe in un paradosso.

La scacchiera

La scacchiera 5x5 può essere da tavolo o da pavimento. Quella da tavolo può essere semplicemente disegnata su un foglio. La sua realizzazione è talmente semplice che non richiede spiegazioni, ma chi volesse utilizzare quella del kit CodyRoby e non l'avesse già, può stampare l'ultima pagina di questa guida.



È utile attribuire alla scacchiera delle etichette di riga e di colonna, e assegnare ai 4 lati i nomi dei punti cardinali. Per farlo, si sceglie un punto di vista, si assegnano i punti cardinali come se si guardasse una carta geografica e si assegnano le etichette di colonna (lettere) e di riga (numeri) come in figura, partendo dall'angolo in alto a sinistra (NORD-OVEST).

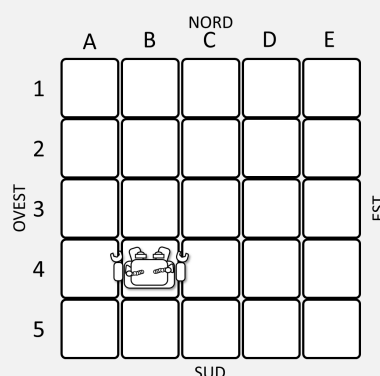
Il robot

Chiamiamo **robot** l'esecutore ideale. Per le **attività da tavolo**, il robot da posizionare sulla scacchiera sarà una **pedina**, che può essere realizzata in qualsiasi modo, purché renda evidente la direzione verso cui sta guardando. A interpretare ed eseguire le istruzioni muovendo la pedina sarà un giocatore che svolge il ruolo di esecutore. Chi volesse usare **Roby** come pedina, lo trova da stampare e ritagliare nell'ultima pagina di questa guida.



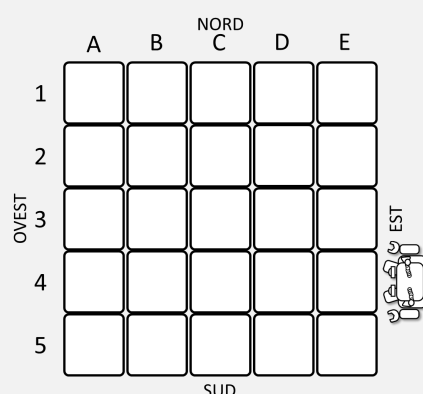
Per le **attività da pavimento**, il robot è una **persona** che svolge direttamente il ruolo di esecutore.

In ogni caso, il robot deve assumere una posizione ben definita all'interno della scacchiera. Questo significa che deve essere chiaramente posizionato su una casella (non a cavallo di due) e orientato secondo gli assi della scacchiera e in una precisa direzione. Questo consente di descriverne la posizione utilizzando le coordinate di colonna e di riga e il punto cardinale verso cui guarda. Ad esempio **B4 Nord** come in figura.



Solo conoscendo la posizione di partenza dell'esecutore il programmatore può prevedere l'effetto dell'esecuzione del programma che sta scrivendo.

Nelle attività che prevedono che il punto di partenza del robot sia esterno alla scacchiera, il robot assumerà comunque una posizione di partenza coerente con la scacchiera, vale a dire, rivolto verso la scacchiera e allineato come se si trovasse su una casella che estende la dimensione della scacchiera stessa.

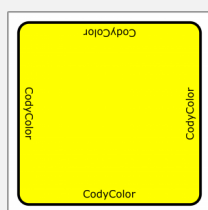


Il repertorio delle istruzioni di CodyColor

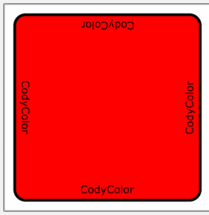
A rendere particolarmente semplici le regole di CodyColor sono due caratteristiche:

1. **le istruzioni elementari sono solo 3 e sono rappresentate da colori** molto facili da distinguere, senza nessuna parola scritta sopra (quindi non bisogna saper leggere, non pone barriere linguistiche e l'istruzione si riconosce a colpo d'occhio);
2. **le istruzioni sono disposte direttamente sulle caselle in cui devono essere eseguite** (quindi non c'è bisogno di tenerle in memoria o di leggerle altrove).

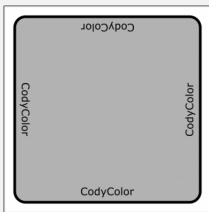
Tessera gialla: girati a sinistra. La tessera gialla rappresenta la rotazione sul posto di 90° in senso antiorario e si legge "girati a sinistra".



Tessera rossa: girati a destra. La tessera rossa rappresenta la rotazione sul posto di 90° in senso orario e si legge "girati a destra".



Tessera grigia: prosegui dritto. La tessera grigia rappresenta l'assenza di rotazione sul posto e si legge "prosegui dritto".



Le tessere hanno dimensioni coerenti con le caselle della scacchiera, poiché vanno disposte al loro interno. Quindi possono avere le stesse dimensioni, in tal caso è come se la scacchiera si colorasse per intero, oppure essere più piccole, nel qual caso vanno disposte al centro della casella a cui sono attribuite.

Le tessere possono essere realizzate con fogli colorati, ricavandone quadrati delle dimensioni desiderate, ma chi volesse utilizzare direttamente il **CodyColorKIT** trova anche le tessere stampabili e ritagliabili nell'ultima pagina di questa guida.

Inizio dell'esecuzione

Quando il robot viene avviato, si sposta automaticamente nella casella che ha di fronte per cercare un'istruzione. Questo lo fa sia che si trovi già su una casella con una tessera colorata, sia che si trovi su una casella vuota o addirittura all'esterno della scacchiera. Questo avanzamento automatico corrisponde a quello che prima abbiamo chiamato "istinto" del robot a procedere nella lettura del programma. Se il robot si trova già su una tessera colorata all'inizio dell'esecuzione si limita ad avanzare sulla casella che ha di fronte, senza eseguire la rotazione eventualmente prevista dal colore su cui si trova. Per evitare fraintendimenti, **è bene che il punto di partenza del robot sia privo di tessere o contenga una tessera grigia** che non prevede alcuna rotazione.

Dinamica di esecuzione

Il robot **legge l'istruzione** quando ci arriva sopra. A quel punto **la interpreta** e **la esegue** effettuando la rotazione sul posto a sinistra (se la tessera è gialla) o a destra (se la tessera è rossa), oppure non effettuando alcuna rotazione (se la tessera è grigia).

Dopo aver eseguito l'istruzione su cui si trova, il robot verifica che nella casella di fronte ci sia una tessera colorata e ci si sposta sopra per leggere l'istruzione successiva. Questa azione è implicita, cioè non codificata da alcuna istruzione, e la chiamiamo "**passaggio alla casella successiva**". Se la casella che ha di fronte è vuota, il robot si ferma in attesa che vi venga posizionata una tessera colorata (se previsto dalla dinamica dell'attività in corso).

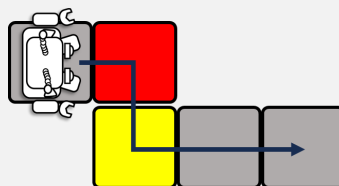
Se, dopo l'esecuzione di un'istruzione, il robot si trova rivolto verso l'esterno della scacchiera, **l'esecuzione termina** e il robot esce dalla scacchiera.

Programmazione

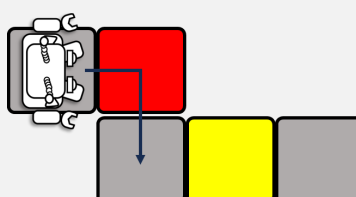
Il programmatore scrive il codice disponendo le tessere colorate sulla scacchiera. In tal modo dispone le istruzioni direttamente lungo il percorso che il robot percorrerà eseguendole.

Detta così sembra banale, ma non lo è né per il programmatore (che deve tener conto dei colori nella costruzione del percorso per garantire coerenza), né per l'esecutore (perché il percorso può non essere evidente quando la scacchiera è piena di colori).

Questo è un **esempio di un percorso semplice, ben riconoscibile e correttamente programmato**.



Questo è un **esempio di un errore di programmazione di un percorso apparentemente riconoscibile**. Se si prova a percorrerlo rispettando le regole di CodyColor ci si accorge che si finisce fuori percorso.



Questo è un **percorso correttamente programmato, ma non riconoscibile a colpo d'occhio**, perché forma una zona compatta di tessere tutte colorate.

