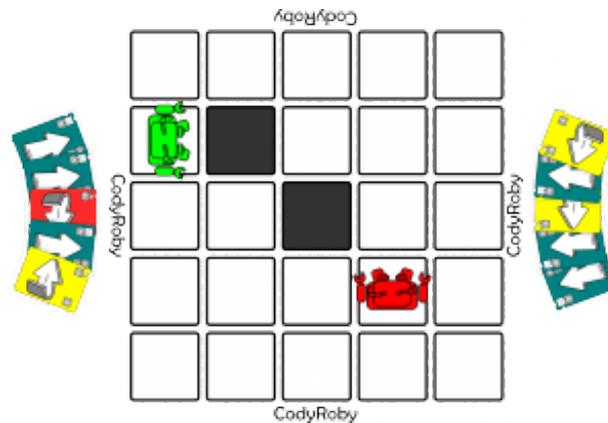




avigation

## *CodyRoby - Il duello*



### *Descrizione*

**Il duello** è un gioco da tavolo della serie [CodyRoby](#) per due giocatori o squadre. I due giocatori (o squadre) usano le carte Cody per muovere i propri robot su una scacchiera 5x5 nel tentativo di catturarsi l'un l'altro.

### *Materiale*

- 24 cody-cards **VAI AVANTI**
- 8 cody-cards **GIRA A SINISTRA**

- 8 **cody-cards** **GIRA A DESTRA**
- **Scacchiera** 5×5 (le dimensioni della scacchiera vanno dai 10 cm, se il gioco si gioca su un tavolo, ai 2.5 metri, se il gioco si gioca in terra muovendosi direttamente sulla scacchiera)
- 2 **blocchi** (pezzetti di carta o oggetti di ogni tipo, non più larghi di un riquadro della scacchiera, da utilizzare come ostacoli o barriere per rendere inaccessibili gli scacchi in cui vengono piazzati)
- 2 **pedine** (possono essere o immagini di Roby, stampate e ritagliate, o segnaposti di qualsiasi tipo, o giocatori che si muovono sulla scacchiera, nel caso in cui il gioco sia giocato a terra)

## *Regole del gioco*

Le due squadre piazzano le proprie **pedine** in due caselle agli angoli opposti della scacchiera. Inoltre ogni squadra può collocare un blocco dove vuole per rendere inaccessibile una casella. Le 40 carte formano un mazzo che deve essere mescolato e posto sul tavolo tra i giocatori, con le carte rivolte verso il basso.

Ora **il duello può iniziare.**

Ad ogni **turno** una squadra prende 5 carte dal mazzo senza mostrarle all'altra squadra. Se la squadra ha già carte in mano all'inizio del turno (perchè non utilizzate nel turno precedente) prende dal mazzo solo le carte necessarie ad arrivare a 5. La squadra valuta le 5 carte a disposizione e ne usa una o più per muovere il proprio robot nella scacchiera. Le mosse si fanno calando ad una ad una le carte che la squadra intende utilizzare in questo turno. La squadra decide quante carte usare, ma deve usarne almeno una.

**Vince** il duello la squadra che riesce a portare il proprio robot su una casella occupata dal robot dell'altra squadra.

Ci sono due eccezioni da gestire:

- se il **mazzo di carte finisce** prima della fine del duello, allora tutte le carte sul tavolo vengono usate e mescolate per formare un nuovo mazzo;
- se una squadra fa una **mossa sbagliata**, portando il proprio robot fuori dalla scacchiera o in una casella bloccata, perde il duello.

## Condividi questa pagina



*Impara a programmare! Per il tuo futuro e per l'Europa.*

Neelie Kroes

**code's cool**  
programmare è forte!



1506  
UNIVERSITÀ  
DEGLI STUDI  
DI URBINO  
CARLO BO





Agenzia per l'Italia Digitale  
Presidenza del Consiglio dei Ministri



# CodeWeek.it

Il sito CodeWeek.it e' curato da Alessandro Bogliolo, CodeWeek  
ambassador per l'Italia

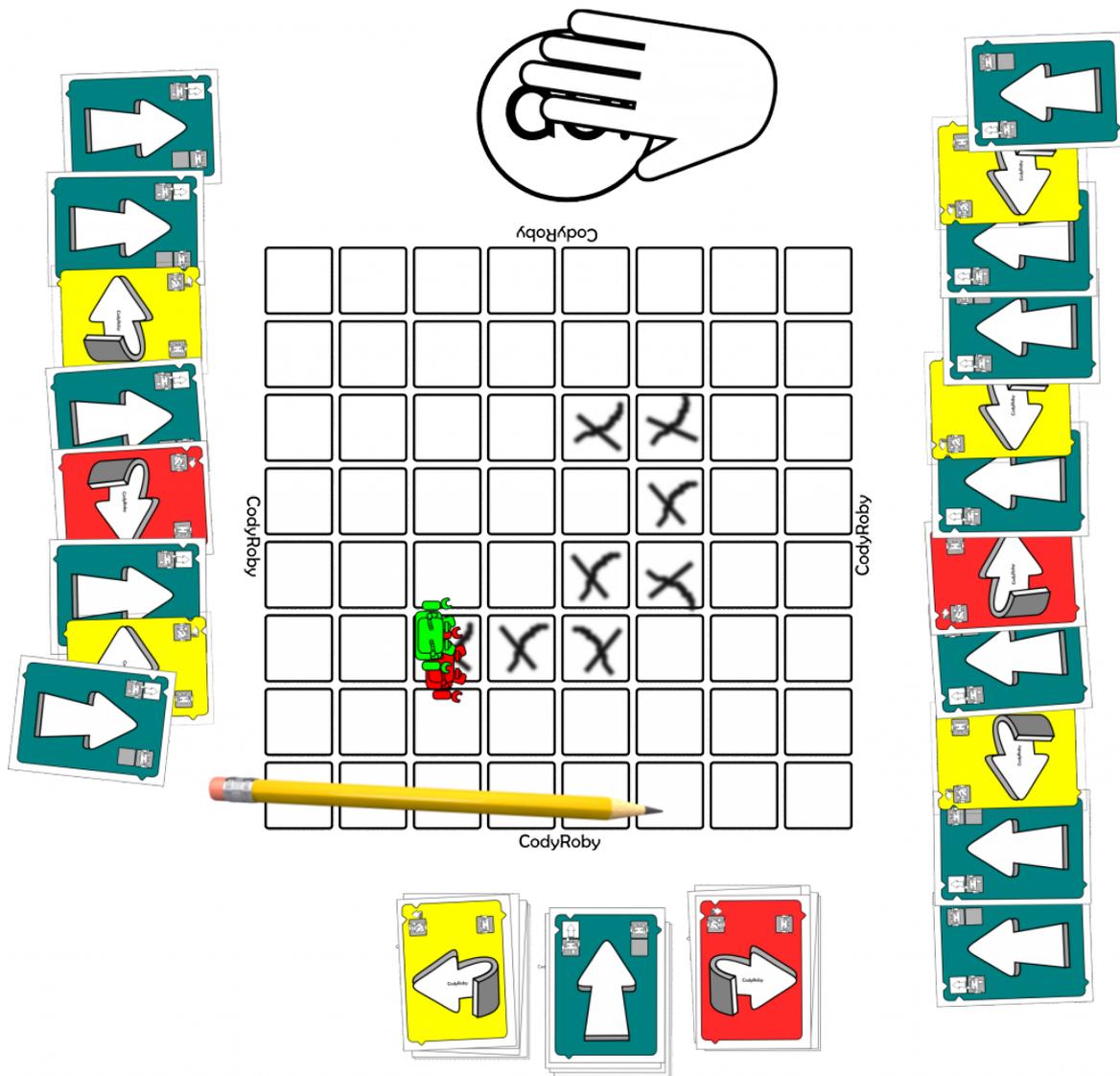
Search ...

Search



Navigation

# *CodyRoby - La corsa*



## Descrizione

**La corsa** è un gioco di velocità della serie **CodyRoby**. E' concepito per essere giocato da due giocatori o da due squadre, ma se ne possono concepire delle varianti a più squadre. Nella descrizione parleremo di due giocatori. I due giocatori vengono posti di fronte ad un percorso e hanno a disposizione carte di tutti i tipi. Devono pensare alla soluzione nel minor tempo possibile, comporla procurandosi le carte che servono e mettere la mano sul tasto GO! Il giocatore che mette per primo la mano sul tasto GO può provare se la soluzione funziona muovendo il suo robot sul percorso. L'altro giocatore controlla se la soluzione è giusta e se ha usato il numero minimo di carte. Se la soluzione è sbagliata o l'avversario riesce a proporre una soluzione con meno carte, la gara è vinta dall'avversario, altrimenti è vinta dal giocatore che ha premuto GO!.

## ***Materiale***

- **Carte:** servono [cody-cards](#) di ogni tipo, in numero almeno doppio rispetto a quelle necessarie a comporre la soluzione. I tipi di carte devono comprendere almeno **VAI AVANTI**, **GIRA A SINISTRA** e **GIRA A DESTRA**. Altri tipi di carte (**RIPETI** e **SE**) potranno essere inseriti in base alle finalità didattiche e al livello di preparazione dei giocatori.
- **Percorso:** il percorso deve essere disegnato su una [scacchiera](#) 5×5 o 8×8 (la scacchiera si può disegnare in un foglio a quadretti o stampare). Si consiglia di disegnarlo a **matita** per poter riusare la scacchiera.
- **Tasto GO!** Può essere disegnato nel foglio accanto alla scacchiera o [stampato](#) e appoggiato sul tavolo.
- **Pedine:** 2 [pedine](#) (possono essere o immagini di Roby, stampate e ritagliate, o segnaposti di qualsiasi tipo, o giocatori che si muovono sulla scacchiera, nel caso in cui il gioco sia giocato a terra)

Tutto il materiale necessario è disponibile anche nello [starter kit](#).

## ***Regole del gioco***

Le due squadre disegnano un **percorso** sulla scacchiera segnando con una matita due caselle a testa a turno. I turni possono ripetersi per disegnare il percorso della lunghezza desiderata. Se si gioca senza usare carte RIPETI, la lunghezza del percorso non deve superare la metà delle carte AVANTI a disposizione.

Si pone (o si disegna) il **tasto GO!** accanto alla scacchiera.

Si dispongono sul tavolo **pile di carte** di ogni tipo a disposizione dei giocatori.

Si dispongono le due **pedine** all'inizio del percorso (anche sovrapposte).

Quando tutto è pronto si dà il via. I giocatori giocano contemporaneamente

prendendo dai mazzi le carte di cui hanno bisogno per **comporre** sul tavolo (ognuno per suo conto) **la soluzione** che guiderà la pedina fino alla fine del percorso. I giocatori devono fare attenzione a non prendere dai mazzi più carte di quelle necessarie, e devono prenderle solo per inserirle nella sequenza (non possono accumularle per riserva). L'incetta e l'uso di carte superflue comportano la squalifica e fanno vincere l'avversario.

Il giocatore che per primo pensa di aver composto la soluzione preme il tasto **GO!** mettendoci sopra una mano e alzando l'altra dal tavolo per indicare di aver ultimato la partita.

Il giocatore che ha premuto GO! è candidato alla vittoria, ma deve **dimostrare di aver trovato la soluzione giusta**. Seguendo le sue carte deve muovere la pedina lungo il percorso, mentre l'altro verifica che la soluzione sia corretta e che non usi carte inutili. Se l'avversario trova un errore o riesce a ridurre il numero di carte (senza pensarci oltre la fine della dimostrazione), la **vittoria** passa all'avversario, altrimenti vince il giocatore che ha premuto GO!

### Condividi questa pagina



*Impara a programmare! Per il tuo futuro e per l'Europa.*

Neelie Kroes

**code's cool**  
programmare è forte!



CodeWeek. 



Agenzia per l'Italia Digitale  
Presidenza del Consiglio dei Ministri



# CodeWeek.it

Il sito CodeWeek.it e' curato da Alessandro Bogliolo, CodeWeek  
ambassador per l'Italia

Search ...

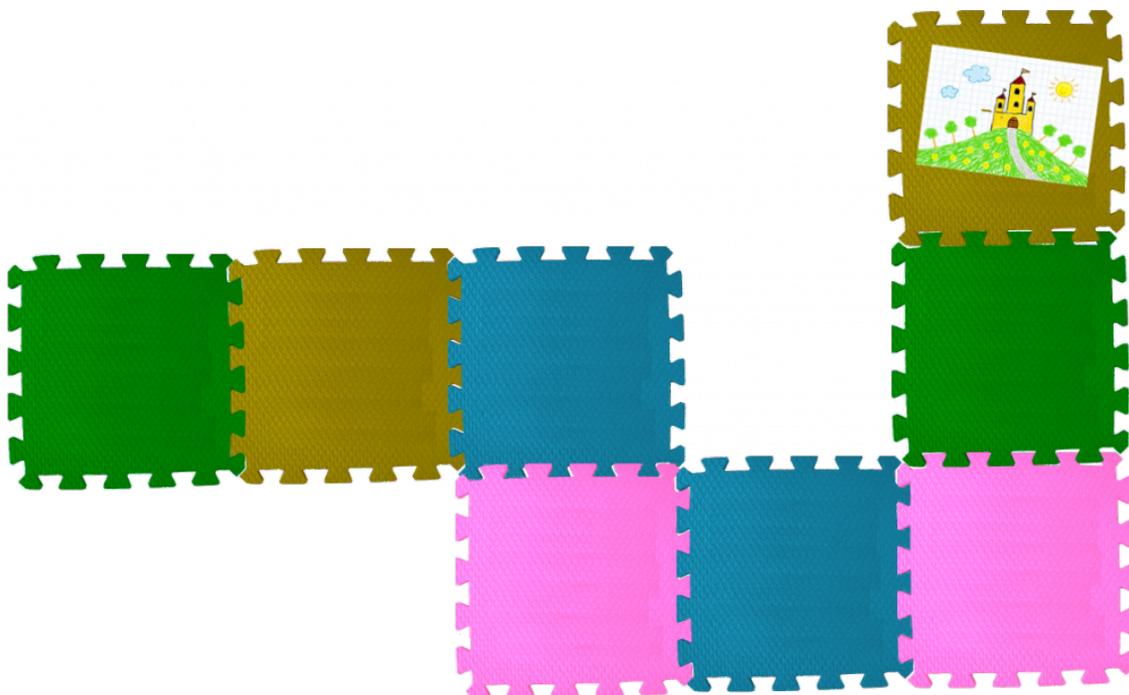
Search





Navigation

# *CodyRoby - La turista*



## *Descrizione*

**La turista** è un gioco a squadre della serie **CodyRoby**. Una turista (potrebbe essere addirittura una principessa) si è persa nella tua città. Quale sarà la squadra che per prima riuscirà ad aiutarla a ritrovare la strada per il castello con le cody-cards?

## ***Materiale***

- **Carte:** servono **cody-cards** di ogni tipo, in numero almeno doppio rispetto a quelle necessarie a comporre la soluzione. I tipi di carte devono comprendere almeno **VAI AVANTI**, **GIRA A SINISTRA** e **GIRA A DESTRA**. Altri tipi di carte (**RIPETI** e **SE**) potranno essere inseriti in base alle finalità didattiche e al livello di preparazione dei giocatori.
- **Percorso:** il percorso deve essere fatto a scacchi e deve essere tracciato sul pavimento. Il materiale ideale per realizzarlo sono i tappeti morbidi a incastro, ma bastano anche semplici fogli di carta appoggiati uno accanto all'altro.
- **Monumento:** il monumento che la turista cerca può essere disegnato su un foglio (magari ispirandosi ai monumenti della tua città) e appoggiato alla fine del percorso.

## ***Regole del gioco***

Le due squadre compongono il **percorso** sul pavimento appoggiando i riquadri (tappeti morbidi o fogli) a turno uno di seguito all'altro.

Si dispongono su un tavolo tutte le **carte**, a mucchietti suddivisi per tipo.

Si disegna su un foglio il **monumento** che la turista vuole andare a visitare.

Si pone il disegno alla fine del percorso.

La bambina o il bambino che fa **la turista** si mette in piedi sulla prima casella del percorso e dà il via.

Le due squadre giocano contemporaneamente prendendo dai mazzi le carte di

cui hanno bisogno per **comporre** (ognuna per suo conto) **la soluzione** che guiderà la turista fino al monumento. Le squadre devono fare attenzione a non prendere dai mazzi più carte di quelle necessarie, e devono prenderle solo per inserirle nella sequenza (non possono accumularle per riserva). L'incetta e l'uso di carte superflue comportano la squalifica e fanno vincere la squadra avversaria. Le carte che fanno la soluzione devono essere messe una sotto all'altra, in modo che alla fine la prima carta (in cima al mazzo) indichi la prima mossa da fare.

La squadra che per prima pensa di aver composto la soluzione porta il **mazzo di carte con le istruzioni** alla turista.

L'altra squadra si avvicina al percorso per controllare la soluzione.

La turista segue le istruzioni passo per passo e svolge il ruolo di **arbitro**. L'altra squadra controlla che la soluzione sia giusta e che la turista usi tutte le carte.

Se le indicazioni sono giuste e la turista usa tutte le carte per raggiungere il monumento, la squadra **vince**.

Se le indicazioni sono sbagliate o ci sono istruzioni inutili, vince l'altra squadra.

## ***Varianti***

Per rendere più divertente il gioco e più bello il percorso si possono costruire percorsi con **diramazioni** che portano a monumenti diversi. Ogni volta si sceglie un monumento con cui giocare.

Per rendere il gioco più dinamico, il percorso può essere arricchito con **esercizi motori**. Quando la turista ha raggiunto il monumento, per festeggiare invita tutti i bambini a seguirla e tutti fanno il percorso.

**Condividi questa pagina**

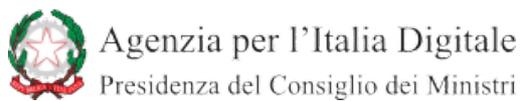
## *La programmazione e' il linguaggio degli oggetti "smart" che ci circondano*

Alessandro Bogliolo

**code's cool**  
programmare è forte!



CodeWeek. 



# *CodeWeek.it*

Il sito CodeWeek.it e' curato da Alessandro Bogliolo, CodeWeek  
ambassador per l'Italia

Search ...

Search



---

avigation

# *CodyRoby - Seguimi*

## *Descrizione del gioco*

Gioco a squadre con [Cody&Roby](#). Ogni squadra è composta da un giocatore nel ruolo di Roby e da uno o più giocatori nel ruolo di Cody.

La squadra A disegna un percorso articolato dando istruzioni al proprio Roby e sfida la squadra B a trovare l'algoritmo che consente al suo Roby di ripercorrere il tracciato. E' utile che ci sia un arbitro.

## *Materiale*

- 20 [carte](#) VAI AVANTI,
- 10 [carte](#) GIRA A SINISTRA
- 10 [carte](#) GIRA A DESTRA.
- Un pavimento con piastrelle larghe quadrate o un pavimento con scacchiera.
- 40 foglietti colorati da lasciare lungo il cammino come tracce.

- Un orologio o un cronometro con i secondi.

## *Regole*

Si stabilisce un punto di partenza sul pavimento. La squadra A ha tutte le carte in mano e decide in che ordine darle al proprio giocatore che svolge il ruolo di Roby per farlo muovere sulla scacchiera. Ogni volta Roby lascia una casella del pavimento ci si posa un foglietto colorito per indicare la traccia.

Il numero di tessere da usare dipende dall'età e dell'esperienza dei giocatori. Al termine del percorso la squadra A raccoglie tutte le carte utilizzate, le mescola e le passa alla squadra B.

La squadra B deve rimettere le tessere in ordine per trovare l'algoritmo che guiderà il proprio Roby lungo il sentiero. Le carte vanno impilate in modo che la prima corrisponda alla prima mossa e l'ultima all'ultima. Quindi la pila va consegnata al giocatore che impersona Roby perchè provi ad eseguirla. Ad ogni mossa corretta la squadra B raccoglie un foglietto-traccia e guadagna un punto. La prova termina se Roby della squadra B esce di pista o completa il percorso.

Al termine della prova si invertono i ruoli delle due squadre. Vince la squadra con il punteggio più alto o, a parità di punteggio, quella che ha impiegato meno tempo.

### **Condividi questa pagina**



*La programmazione e' il linguaggio degli oggetti*

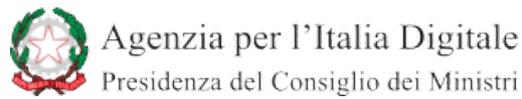
## *"smart" che ci circondano*

Alessandro Bogliolo

**code's cool**  
programmare è forte!



CodeWeek. 



# *CodeWeek.it*

Il sito CodeWeek.it e' curato da Alessandro Bogliolo, CodeWeek  
ambassador per l'Italia

Search ...

Search

# CoderKids, un laboratorio di programmazione unplugged per i bambini di 5-6 anni

**Agnese Addone<sup>1</sup>, Caterina Moschetti<sup>2</sup>**

<sup>1</sup> Insegnante scuola Primaria presso IC Alfieri Lante della Rovere, Roma

[agnese.addone@gmail.com](mailto:agnese.addone@gmail.com)

<sup>2</sup> Insegnante scuola Primaria presso IC Sigillo, Perugia - [catemoschetti@gmail.com](mailto:catemoschetti@gmail.com)

*CoderKids* è un laboratorio di informatica senza uso del computer, nato dall'idea di due insegnanti di scuola Primaria.

Ci siamo incontrate in Rete via Twitter ([@agaddone](https://twitter.com/agaddone), [@catemoschetti](https://twitter.com/catemoschetti)) poco più di un anno fa; in comune abbiamo la passione per la didattica attiva con la tecnologia. Da anni ci occupiamo, nei nostri rispettivi ambiti di insegnamento (Italiano poi Matematica l'una, L2 Inglese, l'altra), di utilizzare l'informatica per le varie discipline. Nel tempo, spinte dalla curiosità personale e da un atteggiamento creativo verso la tecnologia e l'innovazione didattica, abbiamo notato che la scuola generalmente insegna ad usare un insieme di applicativi (di solito software progettati per gli uffici) per raccontare e presentare le proprie esperienze e registrare il risultato finale delle attività di laboratorio o dei progetti.

In questo contesto, che pure ci ha consentito per anni di tradurre le attività in digitale, è maturata sempre più la nostra consapevolezza che fosse necessario compiere un ulteriore passo avanti per far maturare le competenze dei bambini. Dal semplice uso della videoscrittura e dei

software per il disegno, abbiamo sperimentato via via le altre possibilità di realizzazione digitale dei momenti della didattica: abbiamo seguito i ragazzi nella ricerca delle fonti su Internet e nello scambio di materiale per mezzo della posta elettronica; nella creazione di un sito web o della narrazione per mezzo dello *storytelling*; nella creazione di un *wiki* di classe per la scrittura collaborativa; abbiamo creato video e partecipato ad iniziative in Rete usando i social media.

Qui una [presentazione delle attività](#).

Poi abbiamo capito che utilizzare la tecnologia non è sufficiente; che i bambini sono generalmente dei semplici fruitori di strumenti e software anche nel loro privato, ma che non hanno mai la possibilità di crearli da soli.

In una parola, era il momento di insegnare loro a programmare. Cosa per noi difficile, perché non ne sapevamo nulla.

Da qui le ricerche su Internet e, quasi in contemporanea, la scoperta di [CoderDojo](#): i laboratori in cui, in tutto il mondo, si insegna a bambini e ragazzi a programmare, per imparare a non essere solo utilizzatori ma anche a diventare creatori competenti e consapevoli di tecnologia.

Abbiamo studiato, moltissimo. Abbiamo seguito per un anno discorsi difficili di esperti ingegneri, sviluppatori, *maker*. Poi abbiamo cominciato a fare nostra questa esperienza ed abbiamo iniziato ad usarla nella didattica.

La programmazione o *coding* è quella disciplina per mezzo della quale si scelgono e combinano delle istruzioni che, collocate in ordine logico, permettono al computer di funzionare. La giusta sequenza, la cosiddetta *stringa* di codice, segue una sintassi estremamente complessa e articolata, che si apprende solo dopo anni di studio e molta pratica.

Negli ultimi anni, a partire dalle teorie estremamente innovative di Seymour Papert (successivamente riprese, ampliate e approfondite dal [Lifelong Kindergarten Group](#) del MIT Media Lab di Boston, USA), la programmazione ha assunto anche una nuova fisionomia educativa, soprattutto per le ricadute che ha sull'ambito logico-matematico nella risoluzione dei problemi e nello sviluppo della logica: la scrittura e lettura di codice richiede l'uso del pensiero computazionale ed algoritmico.

Saper tradurre idee in codice è un'espressione di creatività ed incentiva la responsabilizzazione ed il ruolo attivo dell'individuo. Per un bambino, imparare a programmare concorre a fargli raggiungere il traguardo della competenza digitale.

I bambini, che oggi sono considerati impropriamente *nativi digitali*, in realtà sanno fare un uso esclusivamente strumentale della tecnologia ma non ne conoscono in modo approfondito le potenzialità e, soprattutto, i segreti.

*Saper programmare è importante quanto conoscere una lingua straniera o come imparare a leggere e scrivere (Mitch Resnick), perché la lingua dei computer è parlata in tutto il mondo .*

*CoderKids* è un progetto di *computational thinking* o pensiero computazionale; nella sua formulazione vuole essere un esempio di esperienza pratica di scrittura di codice senza l'uso diretto del PC: per questo motivo può essere definito *unplugged*, senza strumenti.

La scelta di realizzare un laboratorio di questo tipo è stata fatta per due ordini di motivi: la fascia d'età del target di riferimento richiede un approccio ludico, *game oriented*, di tipo strettamente esperienziale; in secondo luogo l'assenza del pc esclude tutte le possibili difficoltà legate sia all'uso dell'hardware (in particolar modo il mouse e la necessità di avere un'ottima motricità fine per utilizzarlo) che del software per la scrittura del programma.

L'interazione sociale tra pari e il *peer learning* sono i due aspetti educativi che hanno guidato e sostenuto l'elaborazione e progettazione dell'attività: ogni bambino partecipa attivamente e collabora con i compagni.

Il gioco richiede di ideare e realizzare sequenze logiche per raggiungere lo scopo, di tradurle utilizzando un codice di riferimento e infine di renderle attive attraverso il gioco.

Per la realizzazione pratica, il laboratorio *Coderkids* richiede materiali di semplice utilizzo e facili da reperire. Se il tempo a disposizione è limitato ad un incontro, i sussidi vanno preventivamente realizzati a cura dei mentor che lo guideranno, altrimenti si consiglia di prevedere un momento iniziale in cui costruire i singoli elementi insieme ai bambini. L'attività di preparazione aumenta l'aspettativa e la curiosità nei bambini, stimola l'attitudine alla costruzione attiva e allo spirito *maker*, sollecita l'abilità di tradurre in oggetti reali i prodotti del proprio pensiero.

I materiali necessari sono:

- un tappeto nero 270 X 180 cm (realizzabile con bristol o pannolenci) con una griglia bianca, sul quale sono posti degli ostacoli (pozzanghere disegnate e blocchi di costruzioni);
- un foglio nero A4 per ogni bambino, sul quale viene riprodotto il tappeto/griglia;

- gessetti colorati;
- una scheda con i quattro comandi (codice) da utilizzare;
- i *Fuzzes*, pon pon di lana colorata, che riproducono i personaggi di *Kodable Surfscore*, un'app per bambini dai 5 anni, utile per apprendere i primi concetti della programmazione in modo semplice e divertente;
- un tablet con l'app installata (opzionale).



Figura 1 – La realizzazione dei pon pon-Fuzzes.



Figura 2 – I Fuzzes.



Figura 3 – Il tappeto-griglia

Dopo aver disposto a terra il tappeto, ogni bambino riceve alcuni gessetti colorati ed un cartoncino con la riproduzione del tappeto-griglia.

Inizia il momento dell'ascolto. Viene innanzitutto dichiarato lo scopo del gioco, che consiste nel far muovere sul tappeto un bambino nel ruolo di robot sulla base delle indicazioni riportate nella stringa di codice e di farlo arrivare alla fine del percorso (individuato da una croce disegnata sul tappeto), evitando gli ostacoli.



Figura 4 – Il briefing iniziale.

Quindi si illustra come procedere: il percorso va ideato e poi riportato con il gesso sul cartoncino che riproduce il tappeto, utilizzando i quattro comandi a disposizione (il codice) in sequenza.

I comandi sono:

- freccia-avanti, per procedere in avanti;
- cerchio e freccia destra-rotazione destra, per fare passi verso destra;
- cerchio e freccia sinistra-rotazione sinistra, passo a sinistra;
- doppia freccia-salto, per superare con un salto un ostacolo lungo il percorso.

In questa fase resta a disposizione dei bambini il cartoncino che riassume i comandi. Per assicurarsi che i tutti bambini abbiano compreso il significato e le rispettive azioni da compiere, è consigliabile svolgere delle simulazioni preventive.



Figura 5 – Le istruzioni per lo script.

Durante l'elaborazione dello *script*, i bambini traducono in codice il percorso che hanno scelto di proporre al robot per guidarlo a destinazione sul tappeto, evitando con cura gli ostacoli.



Figura 6 e 7 – L'elaborazione dello script.

In questa fase è estremamente importante che gli adulti svolgano la funzione di mentor, che lascino cioè liberi i bambini di sperimentare, provare, sbirciare, scrivere e cancellare. Dovrebbero rispondere alle domande senza fornire soluzioni immediate alle singole richieste. I bambini possono e devono essere liberi di fare errori, di individuarli e di correggersi da soli.

Quando tutti hanno terminato la scrittura del codice, entra in scena il costume da robot, se è stato realizzato; in alternativa può essere sostituito da un Fuzz tenuto in mano dai bambini lungo il percorso.

E' questo il momento in cui la narrazione degli educatori sposta l'attenzione sul contesto ed i bambini si trovano immersi all'interno del videogioco. A turno, ciascuno sceglie un compagno a cui far indossare il costume e da guidare sulla griglia/tappeto attraverso la dettatura del codice: avanti, salta, gira a destra, gira a sinistra.



Figura 8 – La lettura del codice.

Quando tutti i bambini hanno avuto la possibilità di sperimentare entrambi i ruoli e il gioco si è sostanzialmente concluso, si può prevedere un momento di gioco libero con la app, che richiede l'utilizzo di comandi simili. Durante questa attività si consolidano i concetti topologici ed i quattro comandi utilizzati; si entra in contatto con il mondo dei bit; si crea un parallelismo tra la app ed il gioco programmato dai bambini.



Figura 9 – Il gioco libero.

I piccoli programmatori che hanno sperimentato il *CoderKids* hanno lasciato feedback positivi sull'esperienza, disegnando sorrisi su cartoncini rossi! Ognuno di loro ha ricevuto un certificato di partecipazione, inviatici da SurfScore che ha condiviso l'iniziativa dalla Silicon Valley: <http://www.kodable.com/2014/02/05/4-ways-to-make-a-kodable-fuzz.html>



Figura 10 – I feedback.

Il laboratorio *CoderKids* è stato presentato per la prima volta in occasione del [CoderDojo Roma](#) alla *Global Game Jam* del gennaio 2014.

Una seconda presentazione invece si è svolta in occasione di [Didamatica 2014](#), organizzato dall'AICA a Napoli, 7-9 maggio 2014.

