

ITALO
FRUSTINELLI

IVANA SACCHI

BLOCKLY: ATTIVITÀ DI CODING NELLA SCUOLA PRIMARIA

**IL SOFTWARE ALLEGATO FUNZIONA CON
TUTTI I SISTEMI OPERATIVI**



SOMMARIO

3 Introduzione al testo

4 Il pensiero computazionale

5 L'algoritmo

7 Il coding

8 Ambienti strutturati e ambienti aperti

10 Il catalogo - Istruzioni scritte e ambienti visuali

12 Competenze, Abilità e Conoscenze

14 Blockly

15 Menù di navigazione

16 Interfaccia delle attività

18 Blockly su tablet

20 Betta Coniglietta

20 Betta Coniglietta - Trova le carote

22 Betta Coniglietta - Quale carota?

22 Betta Coniglietta - Massimo ... blocchi 1°

23 Betta Coniglietta - Riordina i blocchi 1°

23 Betta Coniglietta - Ottimizza il codice

24 Betta Coniglietta - Tanti passi

25 Betta Coniglietta - Massimo ... blocchi 2°

26 Betta Coniglietta - Riordina i blocchi 2°

26 Betta Coniglietta - Usa ... blocchi

27 Betta Coniglietta - Betta torna a casa

27 Betta Coniglietta - Betta sbaglia strada

29 Milly

30 Ranocchia

30 Ranocchia - Raggiungi la ninfea

31 Ranocchia - Nuovi stagni

33 Ape Logica

33 Ape Logica - Ripeti fino al fiore

36 Ape Logica - Avanti e indietro

36 Ape Logica - Se c'è strada

38 Strade

40 Collane

44 Missione spaziale

45 Problemi

48 Problemi 1

48 Problemi 2

49 Problemi 3

50 Problemi 4

51 Problemi liberi

52 Tartaruga

55 Download e avvio di Blockly

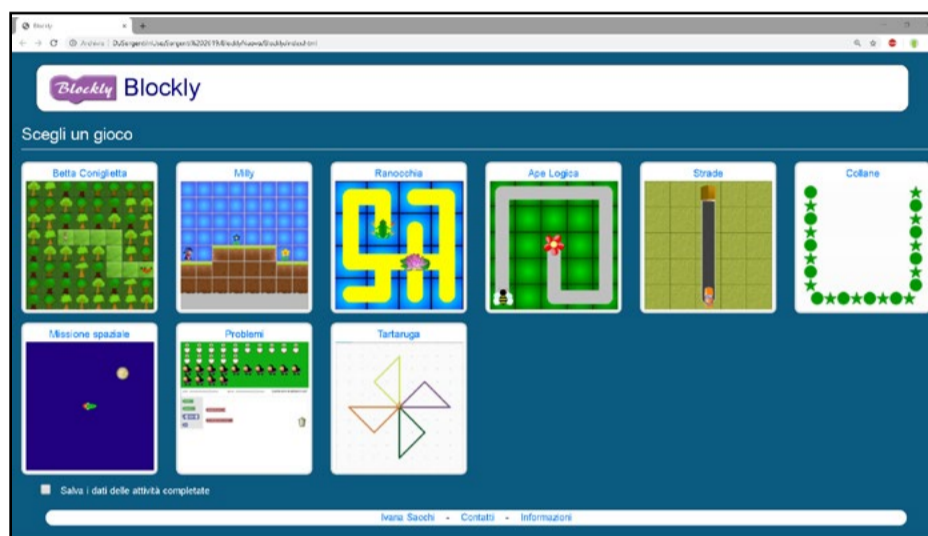
57 Dati di accesso per il download del software

INTRODUZIONE AL TESTO

Questo testo è una guida all'utilizzo di Blockly, che può essere utilizzato on-line all'indirizzo www.ivana.it/bl, oppure scaricato ed utilizzato off-line. Per il download del testo seguire le indicazioni nel capitolo Download di Blockly.

La prima parte del testo introduce l'argomento: cosa vogliono dire parole come *pensiero computazionale*, *algoritmo* e *coding*, cosa sono gli ambienti strutturati e gli ambienti aperti, quale rapporto hanno con la didattica.

Nella seconda parte del testo viene presentato *Blockly*, un ambiente strutturato per il coding. Dopo le prime informazioni, prevalentemente "tecniche", vengono descritti i vari ambienti e la finalità degli stessi.



IL PENSIERO COMPUTAZIONALE

Pensiero computazionale e *coding* sono diventati lessico comune nella scuola. Sovente vengono utilizzati anche come sinonimi, come se si parlasse della stessa cosa. In realtà sono due cose diverse, anche se vi sono attinenze.

Che cosa vuol dire *pensiero computazionale*?

Il pensiero computazionale è un metodo di ragionamento strutturato: è un processo mentale che consente di risolvere problemi organizzando metodi e strumenti. Sostanzialmente è la capacità di risolvere problemi pianificando strategie.



Essenziale è la capacità di scomporre un problema complesso in problemi più semplici, che possono essere analizzati e risolti individualmente. Risolvere le varie parti permette di risolvere il problema complesso.

Nella didattica potremmo dire che è la differenza tra il dire genericamente ad una persona “ragiona”, oppure permettergli di comprendere passo passo cosa si intende per effettuare un ragionamento. Sottolineo che non ho detto “spiegare cosa si intende per ragionare”, ma “permettere di comprendere”; questo perché il pensiero computazionale non si spiega. Si apprende operando.

Ma quali sono i metodi caratteristici del pensiero computazionale?

- Individuare i dati di un problema.
- Organizzare i dati in base a criteri logici saperli rappresentare tramite opportune astrazioni.
- Imparare a scomporre un problema complesso in problemi più semplici.
- Ipotizzare la soluzione del problema .
- Definire un algoritmo per la soluzione, definendone chiaramente i passaggi .
- Verificare la correttezza del proprio algoritmo.
- Individuare eventuali errori e correggerli.
- Confrontare varie soluzioni possibili per identificare la soluzione migliore secondo un criterio (quindi vanno anche definiti i criteri).
- Riconoscere problemi simili già affrontati.
- Generalizzare un processo di risoluzione di un problema per poterlo trasferire ad altri problemi.

L'ALGORITMO

Ho introdotto un nuovo termine: *algoritmo*. Cosa vuol dire?

L'algoritmo è un procedimento utile a risolvere un problema. Viene descritto tramite una serie di istruzioni chiare che possono essere utilizzate anche da altri (uomo o macchina).

Le istruzioni per risolvere il problema devono:

- Essere scomposte in istruzioni semplici (“una cosa per volta”).
- Essere chiare, comprensibili dall'esecutore e non prestarsi ad interpretazioni dubbie.
- Essere ordinate (cosa viene prima, cosa viene dopo).
- Essere in numero finito e terminare dopo un determinato tempo (non vi sono algoritmi infiniti).
- L'esecuzione dell'algoritmo deve portare a un risultato univoco.

Da queste indicazioni possiamo dire che, ad esempio, in un algoritmo che descrive la ricetta del pane potremmo inserire “aggiungere 300 ml di acqua”, ma non “aggiungere acqua quanto basta”, dato che questa ultima indicazione è ambigua.

Ugualmente in un algoritmo per riordinare alcune automobili potremmo dire “se l'automobilina è blu inseriscila nella scatola più grande”, ma non “inseriscila in una scatola”: questa seconda istruzione porterebbe a risultati diversi, non ad un risultato univoco.

Un algoritmo può essere descritto attraverso l'uso di un diagramma di flusso oppure attraverso codice di programmazione (serie di istruzioni presenti in un catalogo comprensibile da chi deve eseguire le istruzioni).

L'immagine seguente illustra un diagramma di flusso per realizzare una spremuta d'arancia. Risulta evidente che all'interno di un diagramma possono essere talvolta necessarie alcune scelte.

