

Pensiero Computazionale e Programmazione nella scuola primaria

Augusto Chiocciariello

Michael Lodi

Laura Vianello



pensiero computazionale e programmazione nella scuola primaria

- Cos'è il «pensiero computazionale»
- Il pensiero computazionale a scuola: cosa sta succedendo in Europa
- Coding alla primaria

Dove troverete il materiale



pensierocomputazionale.itd.cnr.it

Cos'è il pensiero computazionale secondo gli esperti

- Simon Peyton Jones (Computing At School)
- Mitch Resnick (MIT Media Lab)
- Leo Pahkin (Finnish National Board of Education)
- Judith Gal-Ezer (Open University of Israel)

[Developing computational thinking in compulsory education](#)
[Implications for policy and practice](#)

Simon Peyton Jones

Il pensiero computazionale è la capacità di:

- sviluppare astrazioni computazionali di problemi del mondo reale;
- progettare, sviluppare, perfezionare, ragionare di artefatti computazionali (programmi).

Il computing curriculum inglese dice che anche i bambini della primaria dovrebbero essere in grado di "*utilizzare il ragionamento logico per prevedere il comportamento di semplici programmi*". Cioè, essere in grado di:

- spiegare a qualcun altro che cosa un programma dovrebbe fare;
- capire perché un programma non si comporta come previsto.

Prevedere è importante. Programmare non è solo scrivere, ma anche essere in grado di eseguire mentalmente ciò che si è scritto. Questo è pensiero computazionale.

Mitch Resnick

Il pensiero computazionale è un modo di esprimersi e comprendere il mondo utilizzando computer e idee computazionali.

Possiamo scomporre il pensiero computazionale in tre categorie: concetti, pratiche e prospettive.

- I **concetti** sono quelli che emergono dall'uso e dalla comprensione dei computer e dell'informatica, ad es. iterazione e variabili.
- Le **pratiche** sono strategie utili per progettare e risolvere problemi (ad esempio, sperimentare e iterare, "debugging", "remixare" e riutilizzare).
- Nel creare cose con i computer gli studenti cominciano a vedere se stessi e il mondo in una nuova **prospettiva**. Si rendono conto che possono essere "creatori" e da questa nuova prospettiva ripensare a come le cose nel mondo sono fatte.

Leo Pahkin

È possibile tradurre *computational thinking* in finlandese, ma la traduzione suona troppo come se l'uomo cominciasse a pensare come una macchina. Pertanto, è meglio utilizzare un termine più generale: pensiero algoritmico.

Il pensiero algoritmico non deve necessariamente includere l'utilizzo di un computer. Non è qualcosa che è stato inventato con il computer, ma è sempre esistito nella storia umana. Ad esempio, la somma di due numeri è anche una sorta di algoritmo.

Utilizzando le fasi del pensiero algoritmico, gli studenti possono risolvere problemi usando strumenti semplici come i fogli di calcolo. Il termine copre l'idea che è necessario utilizzare una serie di procedure logiche strutturate per trovare una soluzione a un problema. Gli algoritmi, di solito, aiutano a trovare la soluzione in modo più veloce o più semplice.

Judith Gal-Ezer

Il pensiero algoritmico è lo spirito e l'arte dell'informatica, come è stato sottolineato da Donald Knuth e successivamente da David Harel. Significa guardare ai problemi attraverso molti livelli di astrazione. Cerchiamo di dividere un problema in moduli più piccoli, ognuno dei quali è di per sé un problema algoritmico. Una volta trovato un algoritmo, vogliamo sapere e possibilmente dimostrare che l'algoritmo è corretto, si blocca ad un certo punto, fornisce la risposta corretta ed è il più efficace possibile. In generale, dopo questi passaggi, traduciamo l'algoritmo in un linguaggio di programmazione, infine, il programma viene eseguito su un computer.

Ci sono altre forme di pensiero che gli studenti incontrano a scuola, ad esempio il pensiero matematico; i matematici sono felici di dimostrare che un problema ha una soluzione, sono ancora più soddisfatti di scoprire che la soluzione è unica. Gli informatici vogliono andare oltre, procedere attraverso tutti i livelli di astrazione e infine calcolare la soluzione. All'altro estremo, c'è la modalità di pensiero ingegneristico in cui si pone l'accento sul collocare i dati nelle formule giuste per ottenere un risultato.

Definizione di “pensiero computazionale” - c’è un consenso?

- Un passo indietro: cos’è, secondo voi, l’Informatica?

Una possibile definizione di Informatica (S. Martini)

- Un insieme di **applicazioni**
- Una **tecnologia** che rende possibili quelle applicazioni
- Una **scienza** che fonda quella tecnologia:
la scienza che studia l'elaborazione, la trasmissione e la memorizzazione dell'informazione in maniera automatica

Definizione di “pensiero computazionale” - c'è un consenso?

- Pensare come un informatico per risolvere problemi in ogni ambito della vita ([Wing, 2006](#))
- Come pensa un informatico?

Elementi comuni delle principali definizioni di pensiero computazionale

Processi mentali: strategie mentali utili per risolvere problemi

- Pensiero algoritmico
- Pensiero logico
- Scomposizione di problemi
- Astrazione
- Riconoscimento di pattern
- Generalizzazione

Metodi: approcci operativi utilizzati dagli informatici

- Automazione
- Raccolta, analisi e rappresentazione dei dati
- Parallelizzazione
- Simulazione
- Valutazione
- Programmazione

Pratiche: usate tipicamente nell'implementazione di soluzioni informatiche

- Sperimentare, iterare, fare "tinkering"
- Testare e correggere gli errori (debug)
- Riuso e remix

Competenze trasversali: modi di vedere e operare nel mondo; utili competenze per la vita favorite dal "pensare come un informatico"

- Creare
- Comunicare e collaborare
- Riflettere, imparare, fare meta-cognizione
- Tollerare l'ambiguità
- Perseverare quando si ha a che fare con problemi difficili

Cos'è il «pensiero computazionale»

Il pensiero computazionale è l'insieme dei processi mentali usati per formulare un problema e esprimere la soluzione in modo tale che essa sia effettivamente eseguibile da un computer - essere umano o macchina.
([Wing, 2017](#))

- processo mentale
- formulare problemi ed **esprimere** soluzioni
- che siano **effettivamente** eseguibili

Bibliografia in italiano

- La [traduzione italiana](#) dell'articolo del 2006 di Jeannette Wing che ha proposto il pensiero computazionale
- La [prefazione](#) a M. Giordano, C. Moschetti. Coding e pensiero computazionale nella Scuola primaria. La Spiga (2016).
- Il libro R. Marchignoli, M. Lodi. EAS e pensiero computazionale. La Scuola (2016). ([un estratto sulla definizione è sfogliabile qui](#))
- Un [ciclo di seminari](#) e una [guida per insegnanti](#) a cura dell'ITD-CNR. La guida spiega alcuni dei concetti menzionati in precedenza.

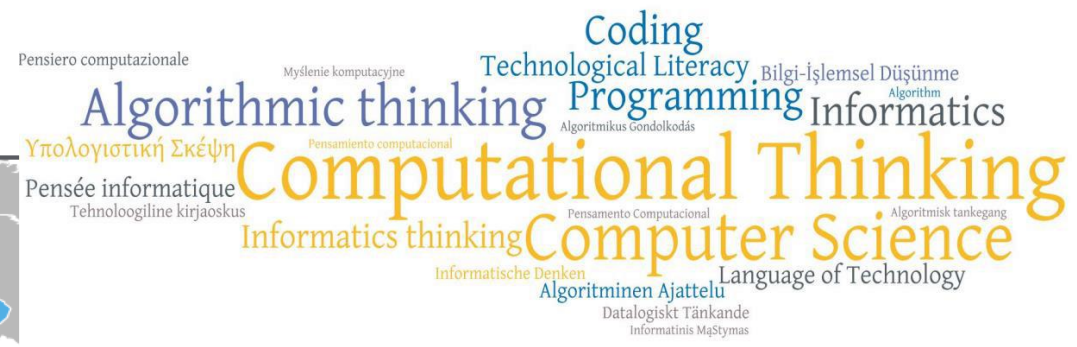
Cos'è il «pensiero computazionale»

- Domande?

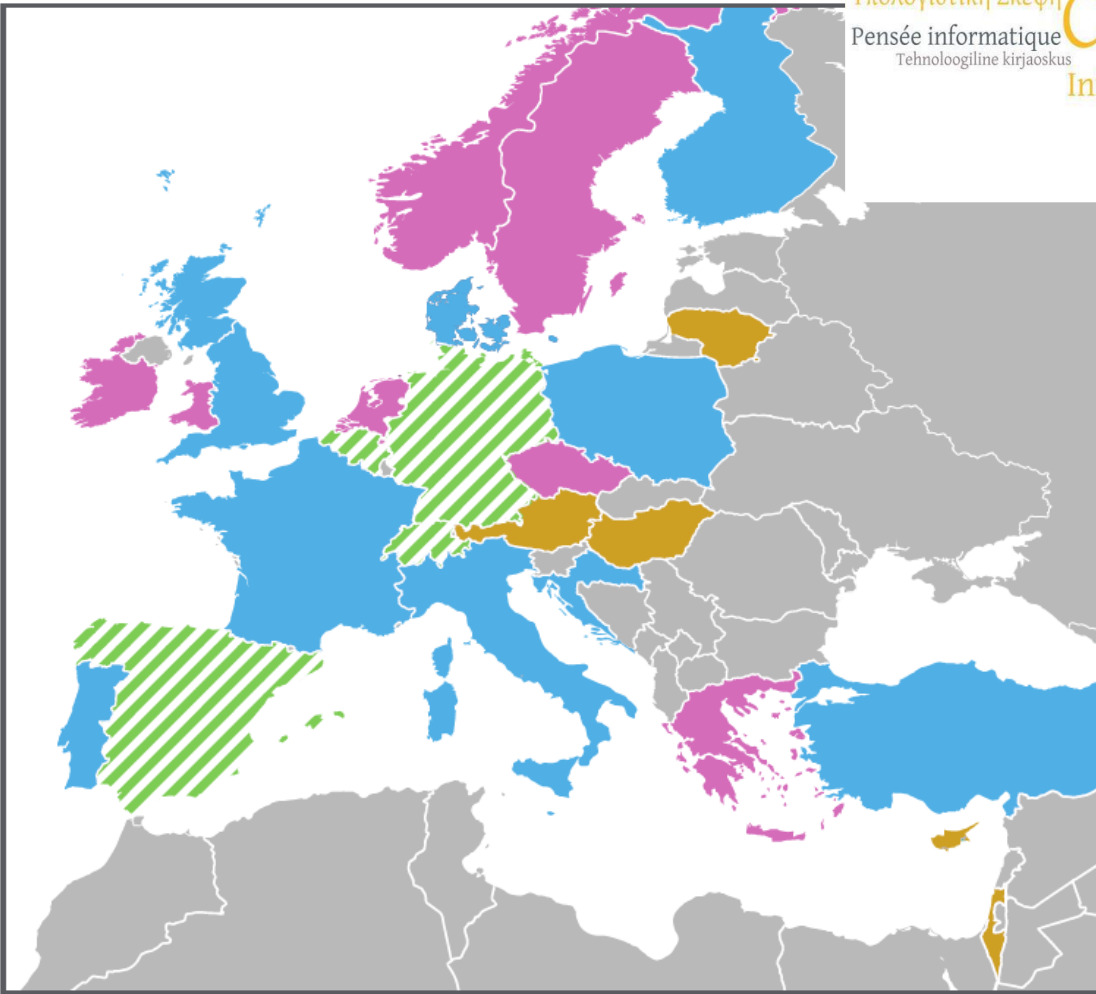
Il pensiero computazionale a scuola: cosa sta succedendo in Europa





Nel contesto dello studio [CompuThink](#) è stata condotta un'indagine sul grado di integrazione del pensiero computazionale nelle attività e politiche dei ministeri dell'educazione europei.

- 19 stati hanno risposto
- curricoli che integrano il pensiero computazionale sono stati analizzati



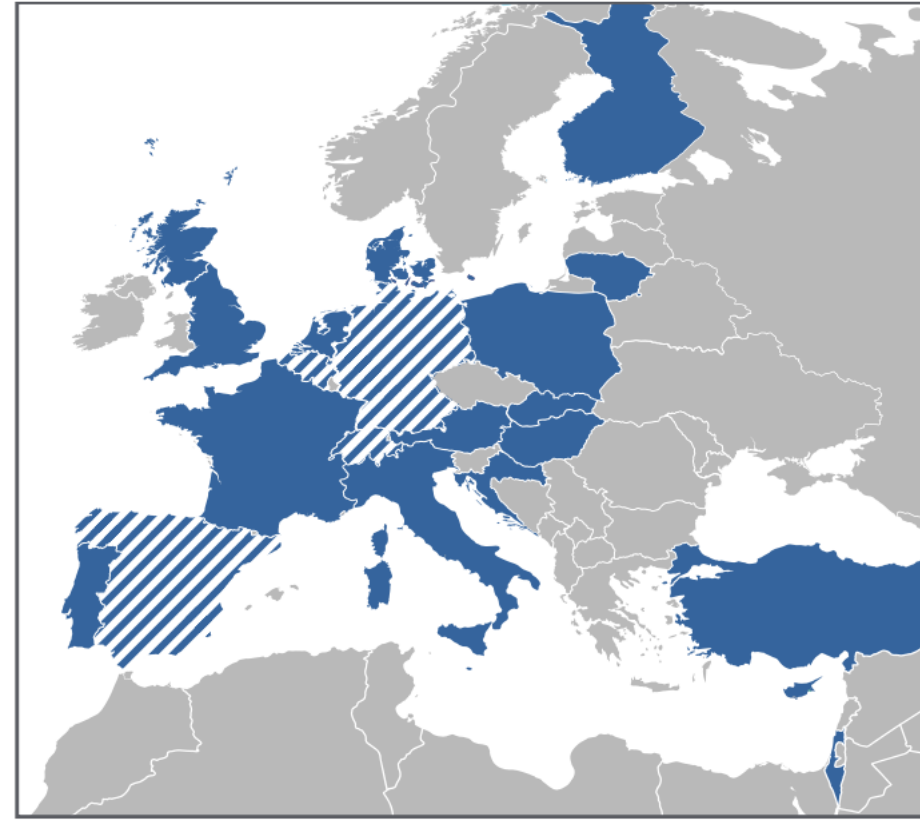
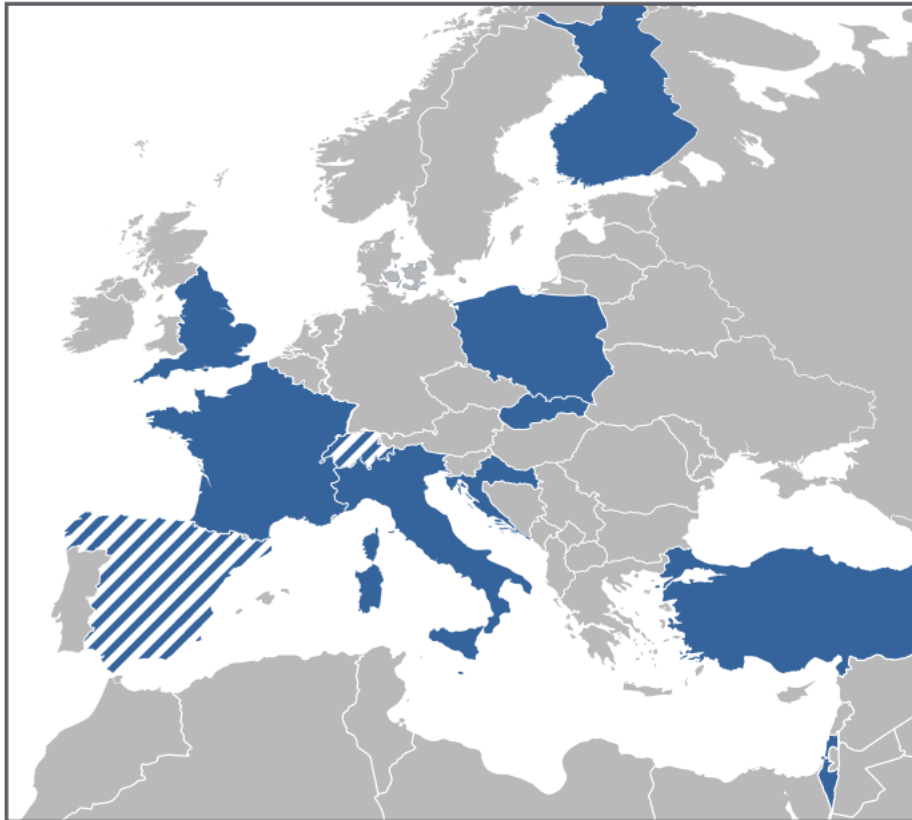
Le risposte di 19 ministeri



-  Curriculum renewal to integrate Computational Thinking
-  Planning to introduce Computational Thinking
-  Building on long-standing Computer Science tradition
-  Policies defined at regional level

CT policy initiatives in primary schools

CT policy initiatives in secondary schools



● CT policy initiatives already in place in primary schools

▨ CT policy initiatives at regional level

● CT policy initiatives already in place in secondary schools

▨ CT policy initiatives at regional level

Country	Within a subject	Across all subjects	Depends on regional or school curricula
Austria	Informatics (upper secondary level)		
Denmark	Information/technology (in grades 10-12)	(in grades 0-9)	X
Finland	Mathematics (grades 1-9) Crafts (grades 7-9)	Transversal competences (e.g. ICT competences)	X
France	Mathematics (Cycle 2-3, primary level) Math and Technology (Cycle 4- lower secondary)		
Hungary	Information technology (grades 1-4; and grades 9-12)		X
Italy	Informatics/ technology IT Curriculum - Applied Science	X	X
Israel	Computer Science		X
Lithuania	Informatics and Information Technology (IT) (grades 5 -12)		
Malta	ICT subject	Part of Digital Literacy (primary level)	X
Poland	Informatics (grades 0-12)	X	
Portugal	- ICT subject (grades 7-8) - Informatics (grades 10-12)		
Switzerland	X	X (primary and lower secondary level German speaking schools)	
Turkey	ICT and Informatics (grade 5-6)		

Integrazione nel curriculum

- Inghilterra - ha varato il [computing programme of study](#) operativo dal 2014
- Finlandia – ha rivisto il curriculum nel 2014 con l’inclusione di coding e pensiero algoritmico, operativo tra il 2016 e il 2018
- Francia – ha rivisto il curriculum nel 2015 con l’inclusione di coding e pensiero algoritmico, operativo dal 2016

Raccomandazioni

Il report finale dello studio CompuThink raccomanda che, nell'integrare il Pensiero Computazionale nell'istruzione obbligatoria, i responsabili politici e le parti interessate dovrebbero concentrarsi su quattro aree:

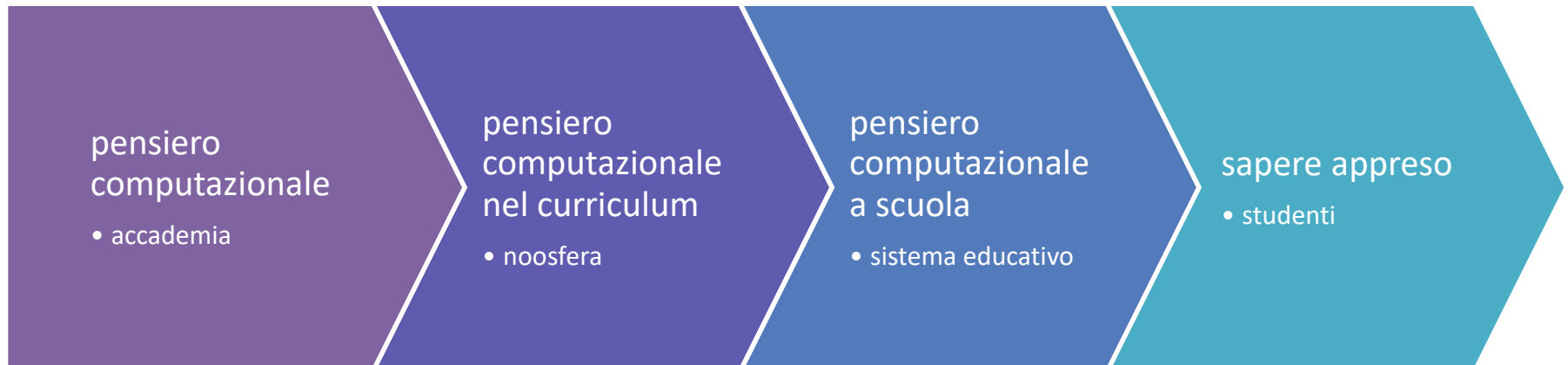
- Supporto politico
- Conoscenza consolidata del Pensiero Computazionale
- Integrazione completa
- Rollout sistematico

Come sta evolvendo la situazione: Irlanda del Nord

- **Supporto politico** - il 6 dicembre 2016 il Ministro dell'educazione per l'Irlanda del Nord ha formalmente accettato e approvato l'insegnamento del pensiero computazionale per tutte le 800 scuole elementari
- **Conoscenza consolidata del Pensiero Computazionale** - i concetti che sottendono le risorse CAS Barefoot si concentrano sui processi mentali (Wing 2006) e sulle "capacità mentali" come suggerito dalla British Computer Society (2014). Entrambe le definizioni concettuali si basano sul lavoro di Papert (1996).
- **Integrazione completa** – la banca dati di risorse all'interno del sito Barefoot, adottata dalle scuole del Regno Unito, è stata sviluppata da insegnanti esperti delle primarie
- **Rollout sistematico** - Tutte le risorse CAS Barefoot sono gratis per le scuole e le scuole possono richiedere gratuitamente formatori provenienti dal mondo dell'educazione e da quello industriale sia per i contenuti sia per le modalità di erogazione dei materiali.

[Il Pensiero Computazionale nelle Scuole Primarie nell'Irlanda del Nord](#)

Trasposizioni didattiche del pensiero computazionale



Il ruolo delle associazioni

- Movimento dal basso
- Spesso supportato da genitori “illuminati” o da aziende tecnologiche
- Può sperimentare didattiche innovative poiché fuori dalla scuola
- Interazione con le scuole del territorio
- Difficoltà a fronteggiare la domanda, a scalare
- Difficoltà ad attrarre pubblico da diversi contesti sociali e culturali
- Diversi livelli di preparazione tecnica e didattica
- Difficoltà a mantenere unito il gruppo, visto che si tratta di volontari

Bebras

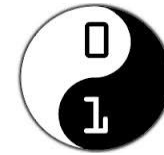


- gara informatica a squadre, gratuita, non competitiva
- aperta a scuole primarie (IV e V) e secondarie (I e II grado)
- non richiede conoscenze specifiche; i giochi vertono su: codifica delle informazioni, logica, pensiero algoritmico, elaborazione dei dati
- si svolge dal 13 al 17 novembre 2017, in concomitanza con le analoghe edizioni nel resto del mondo
- online, dura al massimo 45 minuti e ciascun istituto può scegliere il giorno e l'orario di inizio per le varie squadre.

Computing At School

- Fornire una guida a tutti coloro che sono coinvolti nell'insegnamento dell'Informatica a scuola
- Raggiungere l'eccellenza attraverso conoscenze, competenze e attitudini legate al curriculum
- Formazione degli insegnanti
- Sviluppo del curriculum inglese

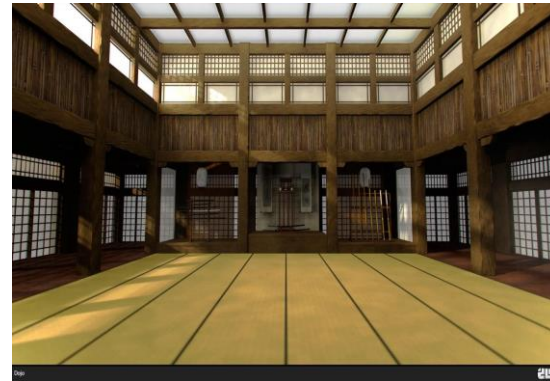
CoderDojo



coder



dojo



+

=

coderdojo

CoderDojo



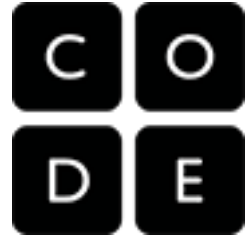
- Tutorial
- Merenda
- Esperimenti

... ma ogni dojo è indipendente

- Computer
- Merenda
- Genitore

uno o due sabati al mese

[Code.org](https://code.org)



- Associazione no profit
- Diffondere la programmazione (coding) e il pensiero computazionale nel mondo
- Garantire l'accesso a categorie svantaggiate (donne, persone di colore, persone con basso reddito, ...)
- Sostenuto e finanziato dalle più importanti multinazionali dell'informatica
- Materiali accessibili a tutti per una diffusione massiva
- Testimonial famosi e personaggi accattivanti

Cosa sta succedendo in Italia

Legge 107 (“La buona scuola”) Articolo 1, comma 3, lettera h
... sviluppo delle competenze digitali degli studenti, con particolare riguardo al pensiero computazionale ...

Piano nazionale Scuola Digitale



**PORTARE IL PENSIERO
LOGICO-COMPUTAZIONALE A
TUTTA LA SCUOLA PRIMARIA**



Programma il Futuro

Blocks Workspace: 1 / 3 blocks

- when run
- move forward
- turn left
- turn right

▶ Run

Can you help me to catch the naughty pig? Stack a couple of "move forward" blocks together and press "Run".

- Versione italiana di Code.org + Sito di supporto
- Programmazione visuale + Attività unplugged

Il pensiero computazionale a scuola: cosa sta succedendo in Europa

- Domande?

Coding: cominciamo dalla primaria



L'ORA DEL CODICE

Cosa possiamo imparare dalla storia della programmazione a scuola



```
TO ME
1Ø PRINT "YOUR NAME"
2Ø REQUEST "ME"
3Ø PRINT SENTENCE OF /ME/ AND "IS SILLY"
4Ø ME
END
```

```
←ME
YOUR NAME
*HENRY
HENRY IS SILLY
YOUR NAME
*HENRIETTA
HENRIETTA IS SILLY
YOUR NAME
*BILLY
BILLY IS SILLY
⋮
```

Possiamo cominciare dalla primaria?

- Se l'**astrazione** è un concetto chiave del pensiero computazionale, come lo si affronta con i bambini?

Le preoccupazioni di una ricercatrice in didattica dell'informatica

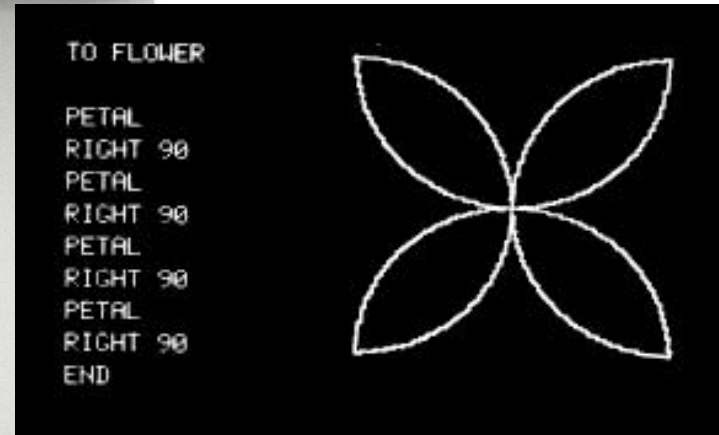
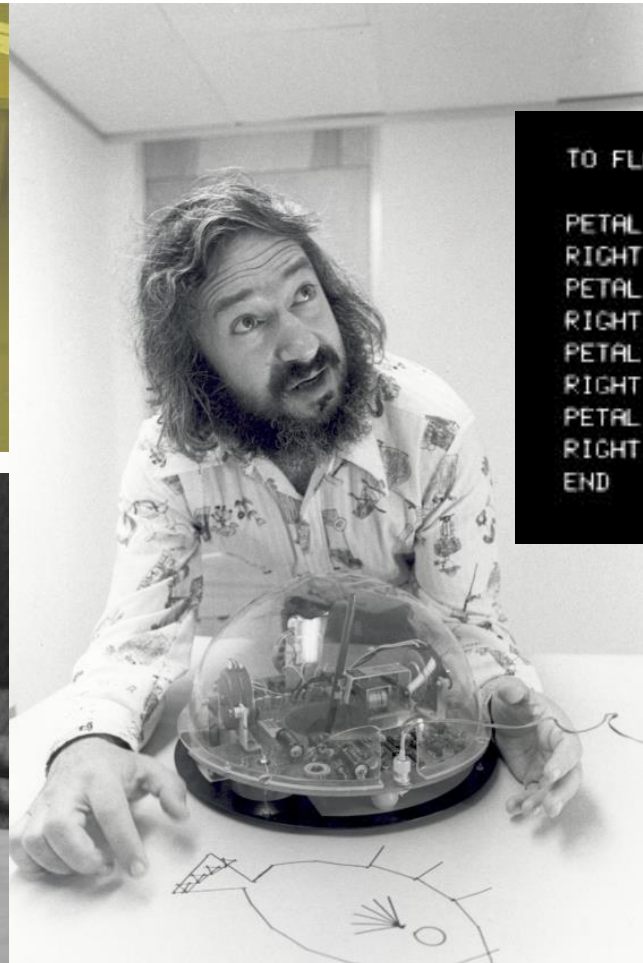
- Anche bambini più grandi (dai sette agli undici anni) che sono alla fase operativa concreta possono solamente risolvere problemi che si riferiscono a oggetti o eventi reali (concreti) e non a concetti astratti o compiti ipotetici. ([M. Armoni, 2012](#))

Papert - un linguaggio di programmazione per bambini

- I miei amici nel settore della psicologia dello sviluppo sono stati tranchant sulla possibilità che una versione significativa di attività di “programmazione” potesse essere gestita da bambini che non avevano ancora raggiunto la cosiddetta fase formale dello sviluppo, il che significa l'età a cui si frequenta la scuola media.
- Io ero più consapevole di quanto questo dipendesse da ciò che si intende per "programmare". [...] "programmare" è qualcosa che si può costruire in modi radicalmente diversi?

[The Children's Machine: Rethinking School in the Age of the Computer](#)

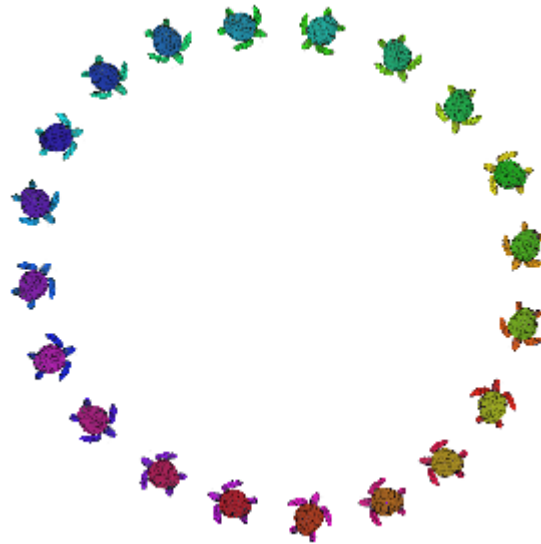
il papà della tartaruga



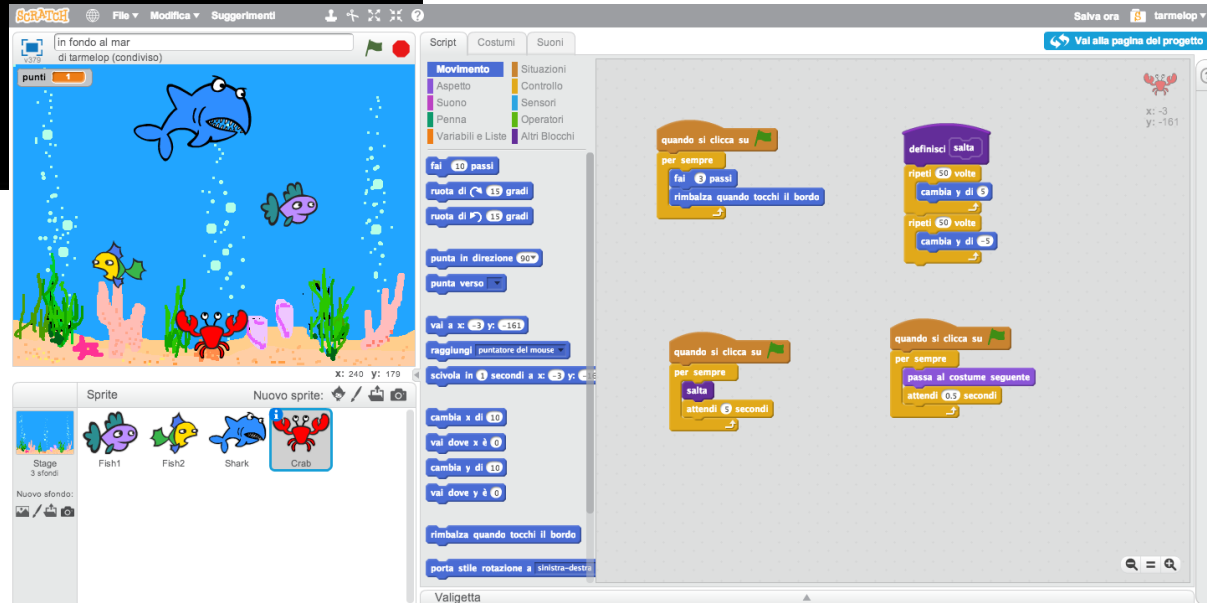
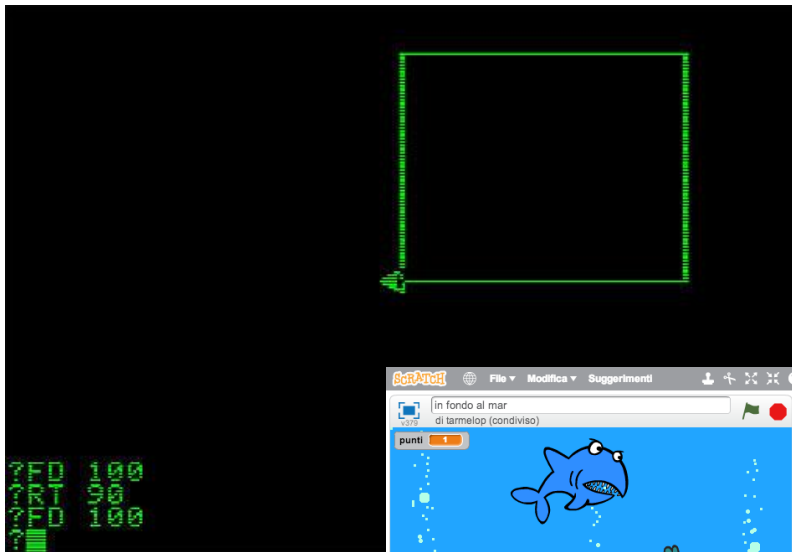
apprendere in sintonia con il corpo



il cerchio e la tartaruga



da Logo a Scratch



Resnick e il Lifelong Kindergarten



Invece di rendere gli asili come il resto delle scuole, dobbiamo rendere il resto delle scuole (in realtà, il resto della vita) più simili all'asilo.

[All I Really Need to Know \(About Creative Thinking\) I Learned \(By Studying How Children Learn\) in Kindergarten, 2007](#)

[Lifelong Kindergarten: Cultivating Creativity through Projects, Passions, Peers, and Play, 2017](#)

le 4 P del Creative Learning

projects



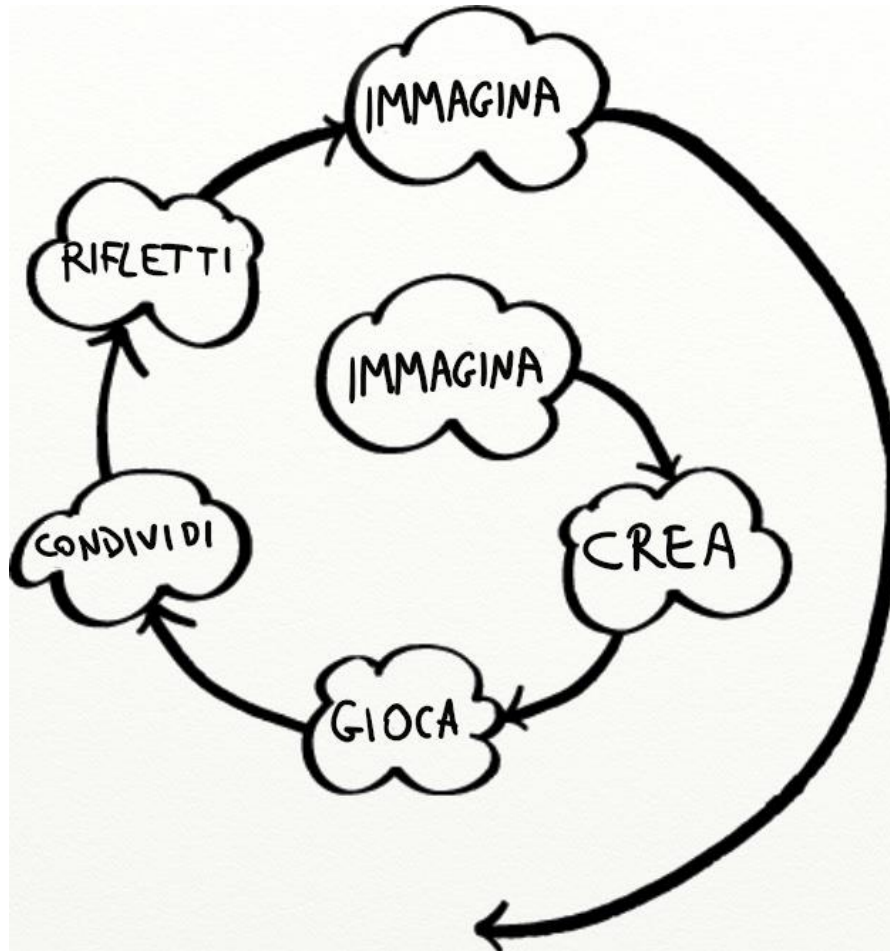
peers

play



passion

la spirale dell'apprendimento creativo



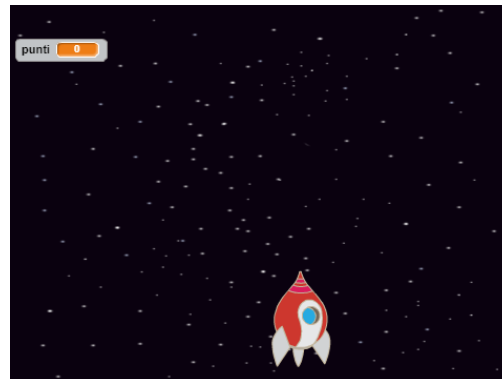
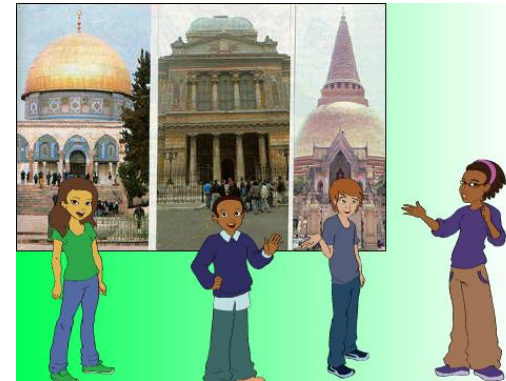
Learning Creative Learning



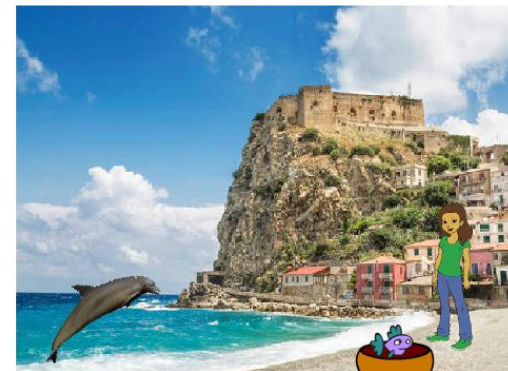
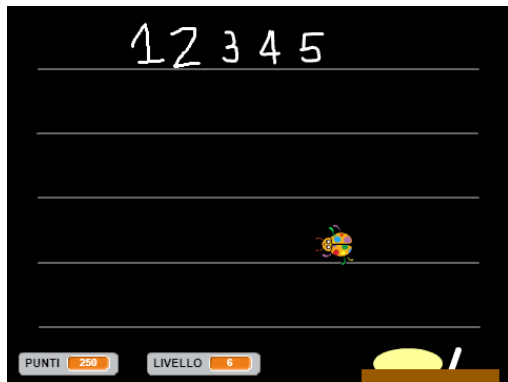
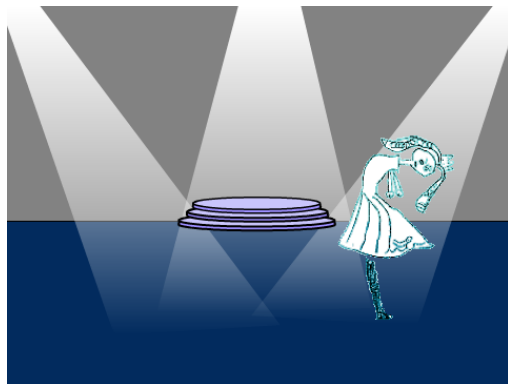
Esempi di progetti: i bambini



Esempi di progetti: futuri insegnanti



Esempi di progetti: proposte didattiche



Indicazioni Nazionali e coding

- L'allievo sperimenta fin dai primi anni le potenzialità espressive della lingua italiana e apprende come sia possibile intrecciare la lingua scritta con altri linguaggi, anche attraverso la produzione di testi multimediali. **(storie multimediali interattive)**
- Riconosce e rappresenta forme del piano e dello spazio, relazioni e strutture che si trovano in natura o che sono state create dall'uomo. **(geometria della tartaruga)**
- Nella scuola primaria si potrà utilizzare il gioco, che ha un ruolo cruciale nella comunicazione, nell'educazione al rispetto di regole condivise, nell'elaborazione di strategie adatte a contesti diversi. **(videogiochi)**

Bibliografia in italiano

- [Il papà della Tartaruga](#) (Papert su geometria della tartaruga e epistemologia)
- La pagina Facebook [Apprendimento Creativo](#)
- [Mani Digitali](#): sito web fatto da maestre, per maestre (e maestri ;)
- Una [guida](#) per partire con Scratch (di Carmelo Presicce, uno che Scratch lo costruisce) - con anche riferimenti ad altre guide in Italiano
- “[Flash cards](#)” utili per stimolare i bambini (da ProgrammaBol)
- Michael Lodi risponde a quattro domande sul pensiero computazionale ([video](#))
- M. Giordano, C. Moschetti. [Coding e pensiero computazionale nella Scuola primaria](#). La Spiga (2016)
- Una [serie di slide](#) che spiegano passo passo come usare Scratch (per insegnanti, non per studenti!)
- Gallerie di esempi per diversi ordini di scuola
<https://scratch.mit.edu/studios/1918506/>
<https://scratch.mit.edu/studios/421797/>
<https://scratch.mit.edu/studios/973416/>

Per una raccolta più estensiva di riferimenti (esempi, guide, libri “di testo”) [clicca qui](#).

Coding: cominciamo dalla primaria

- Domande?

Cos'è il pensiero computazionale

Bibliografia in inglese

- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35..
- National Research Council (2010). Report of a workshop on the scope and nature of computational thinking. Washington, DC: The National Academies Press.
- National Research Council (2011). The report of a workshop on pedagogical aspects of computational thinking. Washington, DC: The National Academies Press.
- Aho, A. V. (2011). Ubiquity Symposium: Computation and Computational Thinking. *Ubiquity*, 2011(January)
- Barr, V., & Stephenson, C. (2011). Bringing Computational Thinking to K-12: What is Involved and What is the Role of the Computer Science Education Community? *ACM Inroads*, 2(1), 48–54.
- Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., ... Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads*, 2(1), 32–37.
- Grover, S., & Pea, R. (2013). Computational Thinking in K–12 A Review of the State of the Field. *Educational Researcher*, 42(1), 38–43.
- Selby, C. C., & Woollard, J. (2013). Computational Thinking: The Developing Definition. University of Southampton (E-prints).
- Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K-6 Computational Thinking Curriculum Framework- Implications for Teacher Knowledge. *Educational Technology & Society*, 19(3), 47–57.
- Tedre, M., & Denning, P. J. (2016). The long quest for computational thinking (pp. 120–129). Presented at the Proceedings of the 16th Koli Calling International Conference on Computing Education Research.
- Rapaport, W. J. (2017). What Is Computer Science. *American Philosophical Association Newsletter on Philosophy and Computers*, 16(2), 2–22.
- Wing, J.M. (2017). Computational thinking influence on research and education for all. *Italian Journal of Educational Technology*, 25(2).