

Pensiero computazionale

È candidato a buzzword dell'anno oppure dietro a questo nome si nasconde qualcosa di estremamente concreto e utile? Scopriamo cos'è e perché è importante sviluppare questa attitudine.

Il concetto di “pensiero computazionale” è stato introdotto per la prima volta da Seymour Papert nel 1996 parlando di LOGO, il linguaggio di programmazione da lui sviluppato al MIT per insegnare la programmazione ai bambini.

Il “pensiero computazionale” è tornato di recente agli onori della cronaca anche grazie a illustri testimonial che ne hanno evidenziato l'importanza, non ultimo il Presidente degli Stati Uniti Barack Obama che durante l'edizione 2013 della Computer Science Education Week si è rivolto ai giovani dicendo “Non limitatevi a giocare con i vostri smartphone, programmateli!”. Nell'edizione 2014 del medesimo evento Barack Obama si è spinto addirittura oltre dando lui stesso il buon esempio e scrivendo le sue prime righe di codice.

Che cos'è il pensiero computazionale?

A oggi non esiste una definizione universalmente condivisa. Quella che sembra mettere d'accordo il maggior numero di esperti è la definizione formulata dalla dottoressa Jeannette Wing, direttrice del Dipartimento di Informatica della Carnegie Mellon University, secondo cui “è il **processo mentale** che sta alla base della formulazione dei problemi e delle loro soluzioni così che le soluzioni siano rappresentate in una forma che può essere implementata in maniera efficace da un elaboratore di informazioni sia esso umano o artificiale”. Ovvero è lo sforzo che un individuo deve mettere in atto per fornire a un altro individuo o macchina tutte e sole le “istruzioni” necessarie affinché questi eseguendole sia in grado di portare a termine il compito dato.

*Se l'obiettivo che devo raggiungere è la costruzione di un tavolino a partire dalle quattro gambe e dal piano d'appoggio, dovrò innanzitutto individuare un linguaggio (per esempio verbale o visuale) attraverso cui comunicare le istruzioni a chi dovrà operativamente realizzare il tavolo e poi tradurre i singoli step necessari alla costruzione in **istruzioni precise e inequivocabili**.*

Perché è importante il pensiero computazionale?

Nonostante il termine “computazionale” possa indurre a pensare che il pensiero computazionale sia un'abilità utile solo a chi ha fatto dell'informatica la propria professione, si tratta di una **skill fondamentale che tutti dovrebbero possedere**, in particolare quei giovani che desiderano non farsi sfuggire le opportunità che il futuro porrà loro dinanzi nei prossimi anni. È ormai universalmente riconosciuto che per riuscire bene nel proprio futuro professionale i giovani dovranno “imparare a imparare” e non limitarsi a fornire risposte preconfezionate: in questa direzione si muovono le raccomandazioni dell'Unione Europea in materia di istruzione che sono state recepite dal MIUR anche con l'introduzione della programmazione nelle scuole a partire dalla primaria. Perché così come leggere, scrivere e contare sono abilità che è importante imparare fin da bambini anche **il pensiero computazionale deve essere appreso ed esercitato fin dai primi anni di scuola**.

Ciò che i nostri studenti universitari imparano oggi, tra cinque anni sarà probabilmente obsoleto: questo vale in maniera particolare per le discipline tecniche, ma anche per le altre discipline.

Inoltre, è davvero difficile pensare che gli studenti di oggi,

che domani troveranno un lavoro, saranno in grado di mantenerlo fino alla fine della propria carriera lavorativa, come accadeva in passato ai loro padri o nonni: è molto più probabile che i giovani saranno chiamati a cambiare lavoro piuttosto frequentemente, imparando a destreggiarsi anche nell'ambito di differenti discipline.

Anche per quanto riguarda le nuove professioni: i lavori oggi più richiesti solo sei anni fa non esistevano neppure. Proviamo a immaginare quante saranno le discipline che nasceranno nei prossimi anni senza che neppure esistano dei percorsi strutturati per poterle accostare.

La vera sfida per i nostri giovani sarà di porsi in un atteggiamento di lifelong learning e acquisire quelle abilità che consentiranno loro di sviluppare un'attitudine mentale utile ad affrontare problemi di ogni ordine e grado.

E quale palestra migliore se non l'esercizio quotidiano del pensiero computazionale per affrontare questa sfida?

Qual è il legame tra computer, informatica e pensiero computazionale?

Così come l'invenzione della stampa ha facilitato la diffusione dell'alfabetizzazione, così oggi la programmazione e i computer facilitano l'acquisizione e la diffusione del pensiero computazionale.

Il pensiero computazionale prende a prestito concetti e strumenti propri dell'informatica per trovare soluzioni innovative e creative ai problemi di ogni giorno.

Questo non significa che gli esseri umani devono imparare a pensare come i computer: il pensiero computazionale è il modo in cui gli esseri umani insegnano ai computer a risolvere i problemi e non viceversa. **I computer sono stupidi e noiosi** e solo grazie agli esseri umani possono diventare strumenti utili e interessanti: l'unico limite a quello che i computer sono in grado di fare è costituito dalla nostra **creatività e immaginazione**.

I processi mentali tipici del pensiero computazionale sono favoriti dall'approccio alla risoluzione dei problemi che viene messo in atto da coloro che sviluppano programmi per il computer: in altre parole il pensiero computazionale è quello che adotta un informatico quando affronta un problema. Esercitare il pensiero computazionale significa quindi molto di più che saper scrivere righe di codice.

*Un informatico direbbe che la moltiplicazione è una somma ripetuta e quindi frutto di un'iterazione, uno dei concetti fondamentali alla base del pensiero computazionale. Inoltre direbbe che, nonostante per la moltiplicazione valga la proprietà commutativa, in termini di **efficienza***

(pratica fortemente ricercata da chi applica il pensiero computazionale) è preferibile sommare tre volte sei piuttosto che sei volte tre.

La programmazione come strumento per sviluppare il pensiero computazionale

Sono in molti oggi a credere che il pensiero computazionale costituisca la **quarta abilità di base** oltre a saper leggere, scrivere e fare di calcolo. Ed è per questo motivo che va facendosi strada sempre più la convinzione che il pensiero computazionale debba essere insegnato a ogni bambino.

Quali sono oggi gli strumenti per insegnare il pensiero computazionale in maniera strutturata? Purtroppo non esiste una risposta univoca a questa domanda: quel che mette d'accordo un po' tutti coloro che a vario titolo studiano la materia (informatici, psicologi, neuroscienziati e pedagogisti) è la convinzione che, fino a che non emergerà un metodo riconosciuto (corredato di strumenti per l'insegnamento e metriche per la valutazione dell'apprendimento), la pratica della programmazione resta il veicolo più efficace.

Papert, colui che per primo coniò il termine "computational thinking" è il padre di una teoria dell'apprendimento nota come **costruzionismo** che sostiene che **la mente umana per poter imparare bene ha bisogno di creare artefatti**, ovvero rappresentazioni reali del mondo con cui interagisce.

E il computer, secondo Papert, è un ottimo strumento didattico poiché, grazie alla programmazione, può creare questi artefatti.

*Il testimone di Papert è stato raccolto da Mitchel Resnick responsabile del Lifelong Kindergarten del MIT MediaLab che con i suoi collaboratori ha realizzato un framework per l'insegnamento del pensiero computazionale e la valutazione dell'apprendimento che si fonda sulla convinzione che **i bambini possano acquisire il pensiero computazionale programmando storie interattive e videogiochi** (gli artefatti di cui parlava Papert).*

*Il lavoro di questi anni di Resnick e dei suoi collaboratori ha portato alla nascita di **Scratch**, un ambiente di programmazione visuale che consente ai ragazzi di creare in maniera semplice e intuitiva le proprie storie animate, giochi e simulazioni: oggi Scratch conta una community di giovani sviluppatori estesa in tutto il mondo ed è di fatto lo strumento di riferimento per insegnare ai bambini il pensiero computazionale attraverso la programmazione.*

Il pensiero computazionale secondo il framework sviluppato dal Lifelong Kindergarten del MIT MediaLab.

Concetti di pensiero computazionale:

- **Sequenza:** un'attività può essere espressa attraverso una serie consecutiva di singoli step o istruzioni.
- **Ciclo:** è un meccanismo per eseguire più volte la medesima sequenza in maniera iterativa.
- **Evento:** il verificarsi di un'azione causa lo scatenarsi di un'altra azione.
- **Parallelismo:** significa eseguire sequenze di istruzioni differenti allo stesso tempo.
- **Condizione:** è la possibilità di prendere decisioni sulla base del verificarsi di determinate situazioni.
- **Operatore:** fornisce supporto per la manipolazione di numeri e stringhe di caratteri.
- **Dati:** sono valori che possono essere salvati, recuperati e modificati durante l'esecuzione di un programma.

Pratiche di pensiero computazionale:

- **Essere incrementali e iterativi:** la progettazione è un processo adattativo dove la pianificazione può cambiare man mano che ci si avvicina alla soluzione del problema.
- **Testare e debuggare:** individuare problemi ed errori e correggerli.
- **Riusare** (*pattern recognition*): riconoscere come alcune parti di soluzione possono essere riusate nella stessa o riapplicate a problemi simili.
- **Remixare** (copiare per migliorare): grazie alla rete e all'ampia disponibilità di lavori di altri autori, è possibile prendere spunto da idee e codice per costruire cose più complesse di quelle che si sarebbero potute realizzare per conto proprio, dando un'ulteriore spinta alla propria creatività.
- **Astrarre:** è il processo di riduzione della complessità, per far emergere l'idea principale mantenendo solo alcuni aspetti e tralasciandone altri.
- **Modularizzare** (scomporre): è il processo che consente di scomporre un problema complesso in problemi più semplici, per cui risolvendo i problemi più semplici si risolve anche il problema complesso.

Attitudini di pensiero computazionale:

- **Esprimere se stessi:** una persona dotata di pensiero computazionale vede nella tecnologia uno strumento per esprimere se stessi, la propria creatività e dire qualcosa di sé agli altri.
- **Essere connessi:** saper comunicare e lavorare con gli altri per raggiungere un obiettivo o una soluzione condivisa.
- **Porre domande:** saper sviluppare una mente vigile grazie alla quale è sempre viva la domanda di come un oggetto incontrato nel mondo reale possa funzionare.