

TEMA

Problem Solving: metodo e creatività per l'informatica nella scuola

Luca Forlizzi*Link al webinar:* <https://youtu.be/1YiMNVr8m7g>

Nei discorsi sull'innovazione del sistema scolastico e nelle presentazioni delle offerte formative delle scuole, *Coding* e *Pensiero Computazionale* sono termini sempre più ricorrenti. Gli ultimi anni hanno visto consolidarsi numerose iniziative, nelle scuole e fuori di esse, volte a promuovere lo sviluppo del pensiero computazionale. All'orizzonte si intravede concreta la possibilità che lo studio del pensiero computazionale entri a pieno titolo nei curricoli scolastici. In questo quadro, sorge spontaneamente l'esigenza di fare una riflessione approfondita su questo concetto (Lodi, Martini & Nardelli, 2017). Cos'è esattamente il pensiero computazionale? È davvero importante per la società, la scuola, l'educazione in generale? È difficile dare risposte chiare a queste domande, stante il fatto che, in realtà, non vi è consenso universale neppure tra gli specialisti su una definizione chiara e operativa del pensiero computazionale. La mancanza di una definizione provoca confusione, tanto che si danno i significati più disparati al termine, completamente svincolati dalle legittime interpretazioni delle parole originarie con cui fu introdotto. L'esempio più chiaro è l'affermazione secondo la quale il pensiero computazionale sarebbe cosa diversa dall'informatica. Punto di vista certamente rispettabile, a patto di dare al termine *pensiero computazionale* un'interpretazione del tutto diversa da quella di chi tale termine ha popolarizzato su scala planetaria (Wing, 2006). A nostro parere, pensiero computazionale (*computational thinking* in originale) è un'espressione che nasce per risolvere un'ambiguità terminologica (informatica intesa come tecnologia e non come disciplina scientifica), ma che ne genera a sua volta un'altra (ossia che si stia parlando di qualcosa di diverso dall'informatica).

Punto di partenza dei discorsi sviluppatisi attorno al termine pensiero computazionale, è l'idea secondo cui l'informatica ha sviluppato (attorno ai concetti e alle conoscenze disciplinari) *modi di pensare e di operare* (strategie di pensiero, metodi operativi, pratiche) che potrebbero essere applicati vantaggiosamente ad altri campi della cultura e del sapere. L'acquisizione di tali strumenti concettuali, a nostro avviso, non può che passare attraverso l'insegnamento dei fondamenti disciplinari dell'informatica (ad un livello appropriato rispetto alle capacità dei discenti). Il pensiero computazionale è quindi "il sedimento concettuale di quell'insegnamento, quello che resta anche quando gli aspetti tecnici sono stati dimenticati" (Lodi, Martini & Nardelli, 2017).

Promuovere il Pensiero Computazionale

Le attività didattiche tese a promuovere la diffusione del pensiero computazionale, possono essere classificate, in linea di massima, in due grandi categorie:

1. orientate alla tecnica
2. orientate alla creatività

Le attività del primo tipo, si evidenziano soprattutto per aspetti tecnici: si introducono gli allievi a un qualche strumento tecnologico, più o meno complesso e ci si pone l'obiettivo di portarli a realizzare lavori che utilizzino in modo sofisticato tale strumento. Anche le attività orientate alla creatività possono partire dall'introduzione di uno strumento tecnologico nuovo, sebbene ciò non sia obbligatorio. L'elemento caratterizzante, però, è lasciare spazio all'uso creativo degli strumenti (o dei metodi), senza richiedere agli allievi sforzi finalizzati a raffinare gli aspetti tecnici dei loro prodotti, ma spingendoli invece a esprimere le proprie idee e le proprie passioni, a essere originali e a divertirsi esplorando le possibilità rese accessibili dalle conoscenze e abilità appena acquisite.

La comunità di docenti, professionisti e anche semplici appassionati che si è spontaneamente formata perseguendo il fine di promuovere la diffusione del pensiero computazionale nella società, e in particolare nel settore dell'educazione, ha recentemente iniziato ad analizzare pregi e difetti delle due categorie sopra delineate. Il dibattito è talvolta sfociato in aperta polemica, fatto piuttosto inconsueto per una comunità che, per quanto giovane, ha sempre esibito un'elevata coesione radicata nella condivisione degli obiettivi finali.

L'uso di attività di tipo creativo per l'introduzione al pensiero computazionale trova solide radici nella teoria costruzionista dell'apprendimento (Papert, 1993) ed è ormai un approccio consolidato che favorisce, negli allievi, la formazione di uno stile di apprendimento che permette in seguito di sperimentare e imparare da soli, approcciare meglio nuovi concetti e conoscenze. Importanti ricadute sono il percepire un sé più completo e autonomo, e un generale accrescimento di benessere, autostima e autoefficacia (Ricci & Colombi, 2017). Il rischio è, tuttavia, che gli allievi non approfondiscano a sufficienza i concetti fondamentali dell'informatica che soggiacciono al pensiero computazionale e che lo stile di apprendimento maturato sia adatto solo a catturarne gli aspetti più semplici. Le attività orientate alla tecnica, d'altro canto, manifestano una tendenza a sottovalutare l'importanza del benessere educativo degli allievi, ma possono permettere di costruire un percorso sistematico e strutturato che

indirizza verso un maggior approfondimento delle conoscenze e, soprattutto, dei *metodi* teorici e operativi dell'informatica (Lodi, Martini & Nardelli, 2017).

Problem Solving e Pensiero Computazionale

Le Olimpiadi di Problem Solving (nel seguito OPS) sono un'iniziativa promossa dal Ministero dell'Istruzione, dell'Università e della Ricerca, che tra i suoi obiettivi fondamentali, ha quello di favorire lo sviluppo e la diffusione del pensiero computazionale nelle scuole italiane (il sito web dedicato all'iniziativa è raggiungibile mediante www.olimpiadiproblemsolving.it). Già nel lavoro lo ha popolarizzato, il pensiero computazionale viene strettamente connesso con le più generali abilità di *problem solving* (Wing, 2006). Una delle definizioni originariamente proposte da Wing, successivamente rielaborata e precisata da numerosi autori, identifica il pensiero computazionale in un insieme di conoscenze, strategie e tecniche generate dallo sviluppo teorico e tecnologico dell'informatica che possono essere efficacemente utilizzate nella *soluzione di problemi* che si manifestano in altri campi del sapere e in generale nella società contemporanea. Le OPS, sin dalla prima edizione, proposta nell'a.s. 2008/2009, hanno sottolineato l'importanza del pensiero computazionale come strategia generale per affrontare i problemi, come metodo per ottenere la soluzione e come linguaggio universale per comunicare ad altri e con esattezza i risultati. In virtù dell'elevato numero di insegnanti e studenti che raggiungono (rispettivamente 730 e 22225 nell'edizione 2017/2018), possono essere annoverate tra i principali vettori del pensiero computazionale nel panorama scolastico italiano.

Le OPS tra Tecnica e Creatività

Rispetto ad altre iniziative, anche guardando al panorama internazionale, le OPS si caratterizzano per l'offerta di un ventaglio di attività didattiche che nel complesso, ma spesso anche individualmente, si pongono come sintesi tra creatività e tecnica. Nell'edizione 2017/2018, la proposta didattica delle OPS è stata articolata lungo tre direzioni:

1. Risoluzione di problemi ricorrenti
2. Sviluppo di un tema assegnato attraverso la programmazione di applicazioni software
3. Progettazione e realizzazione di artefatti dotati di componenti elettroniche e software basate su *board* Arduino compatibili, seguendo temi assegnati

Per trarre profitto dagli stimoli forniti dalla partecipazione a competizioni (Verhoeff, 1997), tutte le attività OPS si concludono con una serie di gare. Gli eventi competitivi, però, oltre a caratterizzarsi per un clima particolarmente amichevole e dunque a svolgere un'importante funzione di socializzazione tra ragazzi provenienti da ogni parte d'Italia, arrivano solo al termine di percorsi didattici non selettivi che vengono portati avanti lungo un intero anno scolastico, rendendo possibile quindi un apprendimento profondo, sviluppato con i tempi necessari.

La realizzazione di applicazioni software, viene effettuata adattando gli strumenti al grado scolastico e al livello di conoscenze degli allievi. In particolare nella scuola primaria e nella secondaria di primo grado, viene adottata la forma di programmazione che in Italia, e soprattutto nel mondo della scuola, assume il nome di Coding (a nostro avviso impropriamente), ovvero quella che fa uso di linguaggi visuali a blocchi manipolabili e componibili, quali Scratch o Snap!. Le scuole secondarie di

secondo grado, preferiscono in maggioranza indirizzarsi verso linguaggi di programmazione testuali, con tipologia variabile a seconda dell'indirizzo di studi.

Le attività della seconda e terza tipologia sono dunque finalizzate all'ideazione e creazione di oggetti con precise caratteristiche tecnologiche. Per ciascun progetto vengano assegnate, dal comitato tecnico-scientifico che organizza le OPS, delle specifiche di massima; tuttavia esse tendono ad essere molto generali e a lasciare ampio spazio di interpretazione. Quindi le attività di queste due tipologie possono essere inquadrare come orientate alla creatività. Naturalmente il fatto che possano essere proiettate in *setting* tecnologici piuttosto spinti (in modo particolare l'uso di Arduino o linguaggi di programmazione a livelli di astrazione relativamente bassi come C, C++) offre l'opportunità, per chi vuole, di approfondire anche la dimensione tecnica e sperimentare il rigore metodologico necessario a dominarla.

Le attività di risoluzione di problemi ricorrenti, pur non richiedendo l'utilizzo di alcuno strumento tecnologico, ma solo di intuizione, ragionamento e capacità di pensiero algoritmico, spingono in realtà a sviluppare tecniche (matematiche, logiche, algoritmiche) che vadano a supporto e potenziamento dell'intuizione, nel momento in cui il crescere della complessità dei problemi affrontati (attentamente graduato nel dipanarsi del percorso didattico) ne rende evidente la necessità (senza che ciò sia esplicitato, lasciando agli allievi il raggiungimento di questa consapevolezza). Ma la creatività non è assente neppure in queste sfide: continuamente vengono proposte variazioni sui problemi presentati, e di tanto in tanto quesiti completamente nuovi. Ciò rende impossibile basarsi unicamente sulle tecniche già note e invita alla continua ricerca di soluzioni innovative.

Conclusioni

Metodo e creatività sono dimensioni dell'apprendimento ortogonali, talvolta messe in contrapposizione ma a nostro avviso invece entrambe indispensabili per preparare al futuro le nuove generazioni. È proprio la loro sinergia che conduce alla formazione di solide competenze e metodi concettuali che consentono di analizzare, formalizzare e risolvere problemi in ogni campo. Le OPS costituiscono un'opportunità di costruire ambienti di apprendimento in cui sviluppare tale sinergia ed esplorarne le potenzialità. Il crescente interesse che esse riscuotono nel mondo della scuola, nell'ambito della ricerca sulla didattica dell'informatica e presso aziende innovative (dunque sensibili ai temi della formazione di competenze che mettano in grado di affrontare le sfide dei settori in continua evoluzione) favorisce il lavoro di continuo rinnovamento e potenziamento delle proposte formative e dei temi che le OPS portano avanti, nonché una loro progressiva integrazione nella dimensione curricolare.

Riferimenti Bibliografici

Lodi, M., Martini, S., Nardelli, E. (2017). *Abbiamo davvero bisogno del pensiero computazionale?*. Mondo Digitale. Novembre 2017.

Papert, S. (1993). *The Children's Machine: Rethinking School In The Age Of The Computer*. Basic Books- HarperCollins Publisher, New York.

Ricci, M., Colombi, A. E. (2017). *Dalla mano al video. Esperienze e osservazioni di costruzione del pensiero astratto, analitico e computazionale nella formazione linguistica della scuola primaria*. Italiano LinguaDue, n. 2, 291-314.

Resnick, M. (2017). *Lifelong Kindergarten: Cultivating Creativity Through Projects, Passion, Peers, and Play*. MIT Press.

Verhoeff, T. (1997). *The role of competitions in education*. Future World: Educating for the 21st Century, a conference and exhibition at IOI, 97.

Wing, J. M. (2006). *Computational Thinking*. Communications of ACM, 49 (3), 33-35.
competenze per la vita. In questo paradigma, le tecnologie diventano abilitanti, quotidiane, ordinarie, al servizio delle attività orientate alla formazione e all'apprendimento degli studenti, e non solo.