

Computational thinking e problem posing & solving

Sperimentazione informatica rivolta alle classi del 1° biennio di 21 istituti della secondaria superiore

Michele Ioffredo

Docente di Tecnologie Informatiche I Biennio I.T.I. "Giordani – Striano"

michele.ioffredo@istruzione.it

Introduzione

"L'attività del progetto *Problem Posing & Solving 100* (PP&S100) – Matematica e Informatica a Scuola – si inserisce nel solco delle azioni orientate a sostenere l'applicazione delle indicazioni nazionali per i licei e delle linee guida per i tecnici superiori e gli istituti professionali approvate nel 2010. A questo si affiancano gli indirizzi espressi dal Ministero in merito all'Agenda Digitale e considerazioni connesse alle emergenti dinamiche dell'economia."¹

¹ http://www.istruzione.it/dg_ordinamenti/Introduzione_PPS_living_lab.pdf

Il Manifesto dell'Association for Computing Machinery (ACM)

Il progetto PP&S100 richiama il Manifesto *"The New Educational Imperative: Improving High School Computer Science Education using worldwide research and professional experience to improve U.S. Schools"*. Lo stesso "White Paper" dell'ACM suggerisce quali debbano essere le strategie necessarie richiamando i seguenti punti, pienamente condivisibili:

- iniziare con la definizione di educazione alla Scienza dell'Informazione riconoscendo l'Informatica come disciplina scientifica che comprende un proprio quadro epistemologico chiaramente rappresentato;
- esprimere un programma nazionale di Scienze dell'Informazione per le scuole superiori basato sui seguenti punti: introduzione ai principi, trattare i contenuti di base e le competenze chiave, indicando anche le strategie necessarie per raggiungere gli allievi e promuovere gli apprendimenti;
- sostenere l'attuazione del nuovo programma con un piano fondato su un arco temporale realistico, individuando le risorse necessarie per realizzarlo;
- fare in modo che l'"Informatica" sia sostenuta da insegnanti preparati e competenti in possesso dei requisiti minimi per poter insegnare quella stessa disciplina, con adeguate attitudini a continuare ad accrescere e aggiornare le proprie conoscenze, abilità e competenze nell'ambito specifico².

Le indicazioni espresse dall'Europa in "Informatics education"

Seguono la stessa linea le indicazioni espresse dall'Europa in *"Informatics education: Europe cannot afford to miss the boat"*, rapporto reso pubblico nel gennaio del 2013, che "Alla luce delle indagini sviluppate, tenendo conto delle esperienze dei vari paesi" esprime "quattro raccomandazioni":

- Ogni studente deve poter beneficiare di un'educazione fondata sulle competenze digitali sin dalla scuola primaria, per padroneggiare i concetti fondamentali, sviluppando non solo abilità, ma anche i principi e le prassi sottese all'uso etico ed efficace di quelle stesse competenze.
- Tutti gli studenti devono poter beneficiare dell'istruzione in Infor-

² <http://minerva.miurprogettpps.unito.it/>

matica, intesa come disciplina scientifica indipendente, studiata sia per la sua valenza epistemologica e didattica che per la sua applicazione alle altre discipline.

- I docenti, preparati e competenti, dovrebbero poter fruire di un piano di formazione organizzato su vasta scala, attuato a breve termine, basato su specifiche soluzioni da individuare anche attraverso la costituzione di partenariati che coinvolgano insegnanti della scuola con esperti provenienti dall'Università e dall'Industria.
- La definizione dei profili può fondarsi sull'enorme disponibilità di materiale già realizzato nel dominio oltre alle specifiche raccomandazioni proposte nell'ambito dello stesso documento.”³

Il progetto Problem Posing & Solving 100

Il M.I.U.R., nell'ambito del *Progetto Problem Posing & Solving*, ha avviato nel corrente anno scolastico da settembre 2013, una sperimentazione nella disciplina Informatica, rivolta alle classi del I biennio di 21 Istituti tra cui l'I.T.I. "Giordani – Striano" di Napoli che hanno questa disciplina nel proprio curriculum (Licei Scientifici con opzione Scienze Applicate, Istituti tecnici e Professionali). La finalità della sperimentazione è di potenziare le possibili sinergie e complementarietà tra informatica, matematica e logica, per sostenere e consolidare il processo di riforma della scuola superiore di secondo grado, seguendo standard internazionali emergenti. Il diffondersi del pensiero algoritmico e la sua evoluzione nel computational thinking delinea una nuova competenza sia per svolgere studi e ricerche in ogni area disciplinare sia per affrontare e risolvere i problemi quotidiani. Come e in che modo il computational thinking e il problem posing e solving giocano un ruolo decisivo in questa prospettiva?

Il percorso "20-hour Intro to the Computer Science course" di code.org 2014

E' possibile un'integrazione tra il programma <http://code.org/> (US public school) disponibile secondo [Creative Commons License](#) e il programma del Living Lab Informatica del progetto PP&S100 che prevede di imparare a programmare con Scratch (<http://scratch.mit.edu/>)?

A queste domande la community dei docenti di Informatica del progetto PP&S100 ha cercato di rispondere attraverso un processo di ricerca azione in un Living Lab svolto in presenza e in rete su piattaforma Moodle

³ <http://code.org/>

dell'Università di Torino, partner del progetto. Su tale piattaforma ogni docente sperimentatore è accreditato per gestire un corso con i suoi studenti di Tecnologie Informatiche.

Nel forum "Gruppo di Informatica", come docente sperimentatore, ho accolto la proposta di un'integrazione tra il programma <http://code.org/> e il programma del Living Lab Informatica del progetto PP&S100 che prevede la programmazione con Scratch. Il percorso "20-hour Intro to the Computer Science course" di code.org 2014 è stato adottato in maniera sperimentale secondo la modalità di apprendimento "Flipped classroom" nella disciplina Tecnologie Informatiche, con tutti gli studenti della IB dell'I.T.I. "Giordani – Striano" con risultati sorprendenti sia dal punto di vista della loro motivazione che del loro profitto (vedi Fig.1).

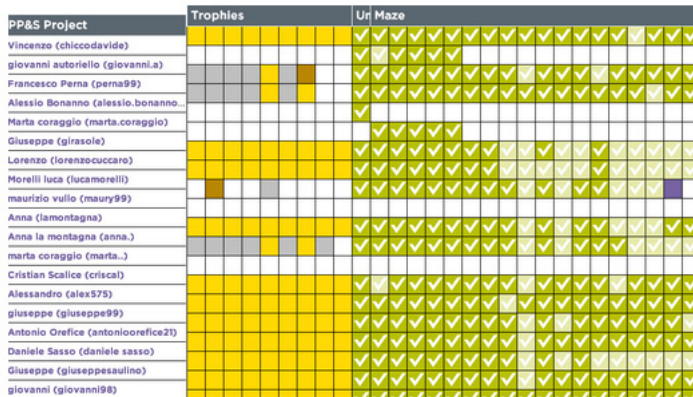


Figura 1 – Code.org Michele Ioffredo Cruscotto di avanzamento

In questo scenario lo studente guidato sia dal docente della disciplina che dal docente tecnico pratico nelle attività in laboratorio, ha seguito il percorso di apprendimento code.org, gradualmente in tutte le 20 fasi a partire dai video in lingua Inglese e sottotitolati in Italiano, sperimentando un nuovo ambiente di apprendimento ideato dal M.I.T. di Boston, "Learning Creative Learning (LCL)" che si realizza ponendo il focus su quattro aspetti chiave dell'approccio live code del MEDIA LAB: **Project, Peer, Passion, Play.**

Gli effetti di questa esperienza informale hanno predisposto gli studenti agli apprendimenti formali riguardo i costrutti fondamentali della programmazione visuale (Linguaggio Blockly), dando la possibilità all'oc-

correnza di vedere il codice nel linguaggio nativo Javascript.

Il percorso di apprendimento code.org è stato progettato per gli studenti US public school K-8, per cui dopo il primo approccio, è stato proposto l'uso del linguaggio Scratch.

Scratch è un progetto del *Lifelong Kindergarten Group* dei Media Lab del M.I.T.. È reso disponibile in maniera completamente gratuita. Imparando a programmare con Scratch, lo studente ha imparato allo stesso tempo importanti strategie su come risolvere problemi, su come creare progetti e come comunicare le proprie idee. Una delle attività didattiche strutturate più significativa è stata la realizzazione della Learnin activity "PI Greco day" declinata sul contesto interdisciplinare di Matematica e Informatica, permettendo a gruppi di studenti, di sviluppare competenze interpersonali attraverso la collaborazione tra pari, l'esercizio della critica e l'uso di sprite in modo creativo attraverso l'ideazione di un gioco interattivo che illustra l'importanza della costante pi greco.

Come sviluppare la valenza interdisciplinare della Scienza dell'Informazione?

Il corso introduttivo di informatica di code.org 2014 ha messo in evidenza "i fondamenti utili per esprimere il ruolo della Scienza dell'Informazione rispetto alla comprensione dei problemi, alla loro formulazione e alla successiva elaborazione di simulazioni e soluzioni. L'apprendimento ha interessato gli aspetti connessi al "problem solving" promuovendo l'elaborazione algoritmica, strumentale alla soluzione del problema. In questo quadro i principi sono stati appresi indipendentemente dalle particolarità mostrate dalle applicazioni e dalla specificità del linguaggio di programmazione scelto nel progetto".

Uno degli scenari reali posto in essere è il Pi greco Day (14/03/2014) per ideare, progettare, realizzare il codice in ambiente Scratch di un gioco che riguardi il numero più famoso e misterioso del mondo matematico (vedi Fig.2).

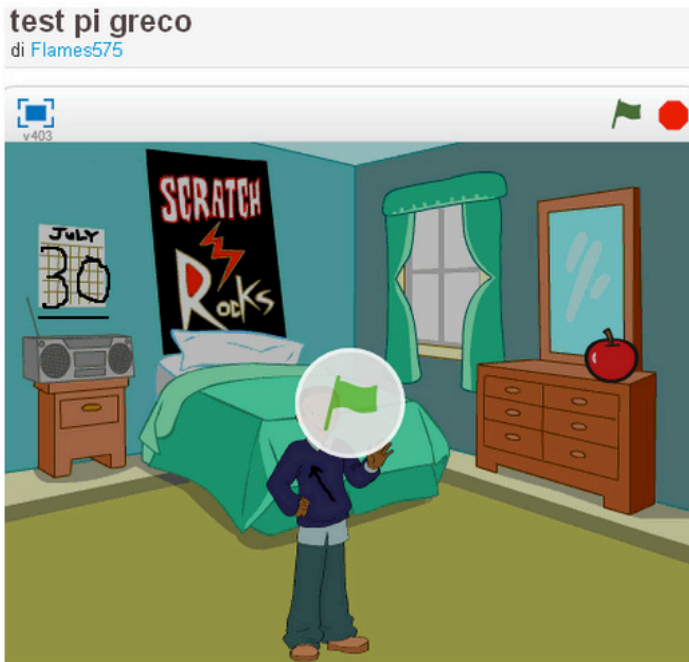


Figura 2 – Un piccolo test per vedere se sapete calcolare il pi greco.

Conclusioni

Il primo anno di sperimentazione ha permesso di offrire agli studenti "una serie di atteggiamenti che esprimono le dimensioni essenziali del Computational Thinking:

- predisposizione al confronto con la complessità,
- persistenza nell'affrontarla,
- tolleranza rispetto all'ambiguità,
- orientamento alla comunicazione e al lavoro di gruppo per raggiungere un obiettivo o una soluzione condivisa.

Pensare come un "informatico" significa molto di più che essere semplicemente in grado di programmare un computer".