



Spunti per lo sviluppo di moduli per l'introduzione del pensiero computazionale nell'ambito dei progetti elaborati dalle scuole in risposta ai bandi PON

Queste tracce non costituiscono documenti di progetto, ma vengono messe a disposizione delle scuole per permettere loro di verificare la compatibilità tra le proprie proposte progettuali e il supporto che verrà loro offerto in fase attuativa dall'Università di Urbino, nei termini espressi all'indirizzo <http://codemooc.org/uniurb-per-il-pon/>

CONTENUTI

Concetti che possono essere inclusi nei sillabi, elencati nell'ordine in cui potrebbero essere trattati e accompagnati da una breve definizione informale fornita al solo scopo di chiarirne l'accezione. I contenuti proposti riguardano solo gli aspetti più strettamente riconducibili al coding e al pensiero computazionale. Non hanno pretesa di completezza e possono essere inclusi (integralmente o parzialmente) nei sillabi di moduli di più ampio respiro eventualmente previsti dalle scuole nell'ambito dei propri progetti.

- **Oggetto programmabile** – oggetto concepito per eseguire istruzioni, e come tale utilizzabile per fare cose diverse usando istruzioni appropriate (distinzione tra hardware e software)
- **Automa** – esecutore automatico privo di intelligenza, di fantasia e di libero arbitrio
- **Istruzione** – comando convenzionale non ambiguo che scatena un'azione elementare univoca
- **Repertorio delle istruzioni** – insieme di istruzioni predefinite
- **Sequenza di istruzioni** – concatenazione di istruzioni elementari
- **Procedimento e algoritmo** – sequenza di passi da eseguire per raggiungere uno scopo
- **Linguaggio di programmazione** – linguaggio formale utilizzabile per descrivere un procedimento in modo da tale che possa essere interpretato ed eseguito da un automa programmabile
- **Programma** – descrizione di un procedimento nei termini di un linguaggio di programmazione
- **Programmazione ed esecuzione** – individuazione dei ruoli del programmatore e dell'esecutore
- **Ripetizione** – esecuzione ripetuta di una o più istruzioni
- **Sensori e input** – acquisizione di dati e consapevolezza del contesto di esecuzione
- **Condizione** – esecuzione condizionata di una o più istruzioni
- **Ripetizione condizionata** – ripetizione di una o più istruzioni condizionata alla verifica di una condizione
- **Evento** – circostanza esterna verificabile che può scatenare l'esecuzione di codice
- **Esecuzione concorrente** – esecuzione parallela di due o più sequenze di istruzioni
- **Sincronizzazione** – tecniche per sincronizzare l'esecuzione di due o più sequenze di istruzioni
- **Procedura** – codice definito una volta per tutte per risolvere un problema specifico in modo da poterlo riusare come elemento per risolvere problemi più complessi
- **Variabile** – rappresentazione simbolica del valore di un dato, che può essere modificato durante l'esecuzione di un programma
- **Parametro** – valore che può essere passato ad una procedura per condizionarne l'esecuzione
- **Struttura dati** – rappresentazione organica di un insieme di dati sui quali opera un programma
- **Digitale** – rappresentabile con un numero finito di caratteri (o cifre) definiti su un insieme finito di simboli (alfabeto)
- **Codifica** – associazione univoca tra significati e loro rappresentazioni convenzionali
- **Funzione** – procedura la cui esecuzione restituisce un risultato
- **Ricorsione** – tecnica di soluzione per riduzione di problemi con struttura autosimile, che prevede l'uso di funzioni o procedure che invocano se stesse
- **Debugging** – individuazione e correzione degli errori di programmazione
- **Sorgente aperto e riuso del codice** – pratica di condivisione del codice che ne favorisce il riutilizzo e il miglioramento continuo
- **Complessità** – stima del numero di passi elementari necessari a completare l'esecuzione di un programma



- **Astrazione e modello** – con riferimento ad un problema o a un fenomeno, l'astrazione serve a coglierne le caratteristiche generali indipendenti da ogni specifica istanza. Un modello offre una rappresentazione di tali caratteristiche astratte che offre supporto al ragionamento
- **Simulazione** – tecnica utilizzata per prevedere il comportamento di un sistema o l'evoluzione di un fenomeno di cui si possiede un modello.
- **Applicazione** – applicazione pratica del pensiero computazionale
- **Specifica** – specifica di progetto

MODULI

La seguente tabella mostra una possibile mappatura dei contenuti su 4 moduli: **Modulo 1** (infanzia e primi due anni della primaria), **Modulo 2** (triennio della primaria), **Modulo 3** (secondaria di primo grado), **Modulo 4** (secondaria di secondo grado). La mappatura proposta è incrementale, nel senso che ogni modulo introduce nuovi argomenti ma copre anche tutti quelli introdotti nei moduli precedenti. Questo serve sia a contemplare il caso in cui i moduli non vengano attivati in progressione (e pertanto ogni modulo debba essere auto-consistente anche in prima erogazione), sia ad evidenziare il fatto che anche i concetti più semplici si prestano ad essere trattati e approfonditi nei diversi ordini e gradi scolastici.

	Modulo 1	Modulo 2	Modulo 3	Modulo 4
Oggetto programmabile	X	X	X	X
Automa	X	X	X	X
Istruzione	X	X	X	X
Repertorio delle istruzioni	X	X	X	X
Sequenza di istruzioni	X	X	X	X
Procedimento e algoritmo	X	X	X	X
Linguaggio di programmazione			X	X
Programma	X	X	X	X
Programmazione ed esecuzione	X	X	X	X
Ripetizione	X	X	X	X
Sensori e input	X	X	X	X
Condizione	X	X	X	X
Ripetizione condizionata		X	X	X
Evento		X	X	X
Esecuzione concorrente			X	X
Sincronizzazione			X	X
Procedura		X	X	X
Variabile		X	X	X
Parametro		X	X	X
Struttura dati			X	X
Digitale		X	X	X
Codifica			X	X
Funzione			X	X
Ricorsione			X	X
Debugging	X	X	X	X
Sorgente aperto e riuso del codice		X	X	X
Complessità			X	X
Astrazione e modello				X
Simulazione				X
Applicazione			X	X
Specifica			X	X



1506
UNIVERSITÀ
DEGLI STUDI
DI URBINO
CARLO BO

APPROCCIO METODOLOGICO

Si consiglia di impostare ogni modulo tenendo conto delle effettive dotazioni di ambienti, strumenti e infrastrutture, in modo da non doverne condizionare l'erogazione a fattori esterni. Qualunque sia lo strumento o l'ambiente di sviluppo adottato, si consiglia di non farne un uso esclusivo in modo da renderne evidente il ruolo strumentale e non rischiare di renderlo centrale. Si consiglia di proporre comunque anche attività unplugged.

A titolo di esempio:

- per il **modulo 1** ci si potrebbe avvalere di attività unplugged (Dresscode, CodyRoby) eventualmente abbinate ai corsi 1 e 2 di Code.org e a semplici strumenti di robotica educativa;
- per il **modulo 2** ci si potrebbe avvalere di attività unplugged (CodyRoby, CodyWay) eventualmente abbinate ai corsi 3 e 4 di Code.org, a Scratch, e a strumenti di robotica educativa;
- per il **modulo 3** ci si potrebbe avvalere di attività unplugged concettuali, del Corso rapido e di PlayLab di Code.org, di Scratch e di eventuali attività di robotica educativa;
- per il **modulo 4** ci si potrebbe avvalere di attività unplugged concettuali, di AppLab di Code.org o di Applinventor2, o di ambienti di sviluppo in JavaScript.

Dovunque vengano attivati i moduli si consiglia di organizzare un gruppo di lavoro di insegnanti della scuola interessati ad applicare il pensiero computazionale in classe coerentemente con i contenuti dei moduli stessi. In tal caso si consiglia di organizzare incontri con i docenti interessati in modo da renderli pienamente consapevoli delle attività di coding svolte all'interno dei moduli, degli obiettivi formativi e dei contenuti, affinché possano sfruttarli proficuamente nella propria pratica didattica, valorizzando e consolidando le competenze acquisite dagli alunni nell'ambito dei moduli e contestualizzando il pensiero computazionale in diversi ambiti disciplinari.

Ovunque possibile si consiglia di mettere in atto strategie collaborative a livello di Istituto che consentano di estendere l'impatto dei progetti al di là del numero minimo di partecipanti ai moduli.

MATERIALI E STRUMENTI FORNITI DALL'UNIVERSITA' DI URBINO

L'Università di Urbino fornirà riferimenti a risorse online su ciascuno degli argomenti in elenco. Ovunque possibile fornirà anche spunti utili all'applicazione interdisciplinare. Infine fornirà gli strumenti di valutazione e banche dati online, come previsto nella lettera di intenti.

ESTENSIONE DELLA COLLABORAZIONE CON L'UNIVERSITA' DI URBINO

Tutti gli strumenti che l'Università di Urbino metterà a disposizione delle scuole che partecipano al PON potranno essere utilizzati liberamente anche da altre scuole. In particolare, benché dal punto di vista formale la dichiarazione di intenti dell'Università di Urbino riguardi il PON sul pensiero computazionale, i materiali saranno anche a disposizione delle scuole dell'infanzia che intendano introdurre il coding nell'ambito del PON sulle competenze di base.

Buon lavoro

Alessandro Bogliolo