

Il valore del computational thinking a partire dalla scuola dell'obbligo: Il programma LOGIC

Pierfranco Ravotto¹ e Giuseppe Albano²

¹ AICA, p.ravotto@aicanet.it

² ANFOR, direzione@anfor.it

*Il presente articolo è una rielaborazione, in italiano, del paper in lingua inglese presentato a 9th IT STAR WS on ICT Strategies and Applications & Business Meeting (BM)
16-17 October 2015, Warsaw, Poland*

La scuola e le competenze informatiche in due call del CEPIS

Nel 2012 il [CEPIS](#) – Council of European Professional Informatics Societies – ha avviato il gruppo di lavoro "Computing in School", coordinato da Bob McLaughlin della BCS (*British Computer Society*). Nel 2014 il gruppo ha elaborato un documento che il CEPIS ha fatto proprio e diffuso: "**Computing in schools. A call for Action from Informatics Societies**". Si tratta di un appello ai decisori politici e ai responsabili dell'educazione e

della formazione che parte da una considerazione: *“Dal momento che le ICT pervadono sempre più la società e l’economia, la domanda di personale IT è in rapida crescita e supera l’attuale offerta di professionisti qualificati producendo una carenze cronica nell’industria informatica. Il basso numero di studenti che scelgono l’informatica come carriera sta esacerbando la situazione.*

È necessaria un’azione immediata per integrare l’informatica nel sistema scolastico e per cambiare la percezione che ne hanno i giovani” [CEPIS 2014¹]. L’appello propone un intervento sugli studenti, sugli insegnanti e sui genitori.

Secondo l’appello: *“I bambini e i giovani oggi usano computer, smartphone e tablet quotidianamente per comunicare con i loro coetanei, riprodurre musica, accedere a informazioni, musica e video. Tuttavia, essi sono raramente incoraggiati a comprendere i principi fondamentali su cui si basa questo fenomeno globale di dispositivi personali. Dove “informatica” è presente come disciplina nel piano di studi, spesso è focalizzata sulle competenze necessarie agli utenti, spesso attraverso programmi come ECDL”.* Provenendo dal CEPIS che è stato il promotore dell’ECDL e che è strettamente legato alla Fondazione ECDL, questo non è certo un giudizio negativo sulla “patente del computer”. L’ECDL certifica la competenza d’uso, la Digital literacy. Il nome “patente del computer” evoca la “patente automobilistica; e nessuno ha mai confuso la patente di guida dell’automobile con le competenze “meccaniche” di chi le automobili le costruisce o le ripara, né tanto meno di chi le progetta. Nessuno confonde una persona con la patente di guida con un meccanico d’officina o con un ingegnere meccanico. Saper “guidare” un computer è forse ancor più necessario per tutti – sul lavoro e nella vita di tutti i giorni – che non saper guidare l’auto. E molto c’è da fare anche in questa direzione, come rivelano le ricerche di AICA sul costo dell’ignoranza informatica. Ma se servono utenti esperti e consapevoli, servono anche manutentori, costruttori, progettisti.

Serve la *Digital literacy*, ma serve anche la Computer science: *“I percorsi informatici nella scuola devono garantire un equilibrio tra lo sviluppo della Digital literacy e l’Informatica. Gli studenti dovrebbero acquisire la competenza digitale (competenza d’uso) per partecipare ad una società che è sempre più digitale, ma dovrebbero anche **capire i fondamenti***

¹ CEPIS (2014). Computing in schools. A call for Action from Informatics Societies. (http://www.cepis.org/media/Computing_In_Schools_Paper_2014_ext1.pdf)

della tecnologia in modo che abbiano la possibilità di **sviluppare competenze informatiche più avanzate**".

Indirizzare i giovani verso professioni informatiche dipende da scelte politiche – per esempio con l'introduzione di una disciplina Informatica – dai genitori e dagli insegnanti. Per quanto riguarda questi ultimi il documento del CEPIS fa la seguente considerazione: *"La proliferazione di giochi e social media ha motivato i giovani a familiarizzare con l'uso della tecnologia. Come risultato gli studenti sono percepiti come più esperti nell'uso dei dispositivi e gli insegnanti sono in imbarazzo per la propria mancanza di abilità in questo campo. Molti non usano nemmeno la tecnologia disponibile per la didattica. Pochissimi hanno una formazione informatica in quanto la preparazione degli insegnanti normalmente non richiede alcuna conoscenza del pensiero computazionale"*. Perché ci possa essere un intervento efficace sugli studenti è necessario un intervento efficace sugli insegnanti.

La proposta **LOGIC**, elaborata in Italia da AICA e ANFOR, è un tentativo di rispondere proprio all'esigenza di diffondere nella scuola il computational thinking partendo da un'azione rivolta agli insegnanti.

Recentemente CEPIS e Fondazione ECDL sono tornati su questo tema con il documento **"Computing and Digital literacy – A call for a holistic approach"** in cui si ribadiscono gli stessi concetti sottolineando l'importanza di non tenere separati i due aspetti – *Digital literacy* e *Computing* – ma di agire contemporaneamente su entrambi.



Computing and Digital Literacy
Call for a Holistic Approach



Supported by the CEPIS Computing in Schools
Special Interest Network

Figura 1 – Il documento della Fondazione ECDL e del CEPIS.

Il documento suggerisce come approccio quello recentemente adottato in UK: *“L’Inghilterra è l’unico Paese con un approccio coerente nell’insegnamento dell’informatica come disciplina a partire dalla scuola primaria. La programmazione viene spiegata come uno degli elementi della disciplina e non come argomento separato. Le competenze di Digital Literacy vengono sviluppate come parte del curriculum di informatica in modo da poter fornire le basi per tutti gli altri argomenti. Questo approccio contraddistingue l’Inghilterra come la nazione che ha assunto un approccio olistico nei confronti dello sviluppo delle competenze digitali”* [Fondazione ECDL-CEPIS, 2015²].

Digital literacy e Computational thinking

La competenza digitale – o *Digital Literacy* – è ben definita. E’ una delle 8 competenze chiave indicate nella Raccomandazione del Parlamento europeo e del Consiglio relativa alle competenze chiave per l’apprendimento permanente [Parlamento europeo, 2006]: *“la competenza digitale*

² Fondazione ECDL-CEPIS (2015), Computing and Digital literacy. Call for a holistic approach (<http://www.ecdl.org/media/Position%20Paper%20-%20Computing%20and%20Digital%20Literacy.pdf>)

consiste nel saper utilizzare con dimestichezza e spirito critico le tecnologie della società dell'informazione (TSI) per il lavoro, il tempo libero e la comunicazione. Essa, supportata da abilità di base nelle TIC: l'uso del computer per reperire, valutare, conservare, produrre, presentare e scambiare informazioni nonché per comunicare e partecipare a reti collaborative tramite Internet". La Raccomandazione³ esplicita in dettaglio le Conoscenze, Abilità e Attitudini che concorrono a tale Competenza (Nella Raccomandazione *"le competenze sono definite in questa sede alla stregua di una combinazione di conoscenze, abilità e attitudini appropriate al contesto"*). Si tratta di una definizione a tutt'oggi valida se con il termine computer intendiamo anche tablet e smartphone.

Secondo [Wikipedia](#), *"L'informatica è la scienza che si occupa del trattamento dell'informazione mediante procedure automatizzabili. In particolare ha per oggetto lo studio dei fondamenti teorici dell'informazione, della sua computazione a livello logico e delle tecniche pratiche per la sua implementazione e applicazione in sistemi elettronici automatizzati detti quindi sistemi informatici"*⁴. In quanto tale è oggetto di studi per coloro che intendono diventare professionisti informatici. Quando pensiamo a un insegnamento non specialistico, rivolto a tutti, ci riferiamo alle basi dell'informatica, o *computer science*, generalmente definite come *computational thinking*.

In risposta alla Consultazione pubblica indetta dal governo Renzi sulla "Buona scuola", AICA – insieme con ANDINF (Associazione Nazionale Docenti di Informatica), Confindustria, CINI (Consorzio Interuniversitario Nazionale per l'Informatica), CNR-ITD (Consiglio Nazionale delle Ricerche – Istituto per le Tecnologie Didattiche) e Rete Robotica a Scuola – ha presentato un documento – "Informatica e pensiero computazionale nei curricula" – in cui si propone l'inserimento del *Computational thinking* nei curricula di tutte le scuole italiane: *"il pensiero computazionale ... fa riferimento ad un insieme di saperi, concettuali e metodologici, e di capacità di contestualizzazione applicativa che devono diventare patrimonio di tutti e devono quindi essere presenti nei curricula dalla scuola primaria fino alla secondaria di secondo grado. Saper usare il pensiero computazionale in una pluralità di contesti – negli aspetti scientifici, sociali e applicativi – permette di sperimentarne appieno le potenzialità, i limiti e le implicazioni*

³ Parlamento europeo (2006), Raccomandazione del Parlamento europeo e del Consiglio relativa alle competenze chiave per l'apprendimento permanente (<http://eur-lex.europa.eu/legal-content/IT/ALL/?uri=CELEX:32006H0962>)

⁴ <https://it.wikipedia.org/wiki/Informatica>

sociali e di cittadinanza attiva" [AICA e altri 2014⁵].

Il *Computational thinking* è la base dell'informatica: saper analizzare un problema, ipotizzarne una soluzione, descriverla in modo formale, implementarla in uno specifico linguaggio per farla eseguire da una macchina (*coding*), sottoporla a test ed eventualmente correggerla, documentarla. Si possono proporre attività di *computational thinking* a studenti di ogni età, e dunque in ogni ordine di scuola. Cambieranno, di volta in volta, la complessità dei problemi, le macchine e i linguaggi.

Il programma **LOGIC** è studiato per la Fascia prescolare e dell'obbligo: Infanzia (3-6 anni), Primaria (6-11 anni), Secondaria di 1° grado (11-14 anni) e primo biennio della Secondaria di 2° grado (14-16 anni).

Il Syllabus italiano di Informatica per la scuola dell'obbligo

Sebbene in Italia una disciplina Informatica sia prevista solo in alcune Scuole Secondarie di 2° grado, non nella Primaria né nella Secondaria di 1° grado, il Ministero ha prodotto, nel 2010, un "Syllabus di informatica per la scuola dell'obbligo" [MIUR, 2010⁶]. Tale syllabus ha in sé quell'aspetto olistico che il documento Fondazione ECDL-CEPIS propone. E' infatti strutturato in due campi: "Strumenti" – quello che abbiamo definito come Competenze di Informatica – e "Elementi di informatica", ovvero *Computational thinking*.

⁵ AICA e altri, (2014), Informatica e pensiero computazionale nei curricula, in La Buona scuola – Consultazione pubblica, Pensiero computazionale e coding – Proposte: raccomandazioni per l'introduzione dell'informatica e del pensiero computazionale nei curricula (<https://labuonascuola.gov.it/area/m/16838/-> http://www.itd.cnr.it/download/pensiero_computazionale.pdf)

⁶ MIUR (2010), Syllabus di Elementi di informatica – la scuola dell'obbligo – anno 2010, (www.olimpiadiproblemsolving.com/documenti/SYLLABUS.pdf)



Figura 2 – Il syllabus articola gli obiettivi di apprendimento in due aree: Strumenti informatici e Elementi di Informatica.

AICA ha una proposta per gli Istituti Comprensivi (Infanzia, Primaria, Secondaria di 1° grado) che corrisponde al syllabus ministeriale. L'aspetto Digital literacy/Strumenti di Informatica è centrato sull'ECDL, quello Elementi di informatica/Computational thinking è centrato su LOGIC. La proposta è far accedere gli studenti alla certificazione a partire dal 5° anno della Primaria, con un percorso Welcome negli anni precedenti.

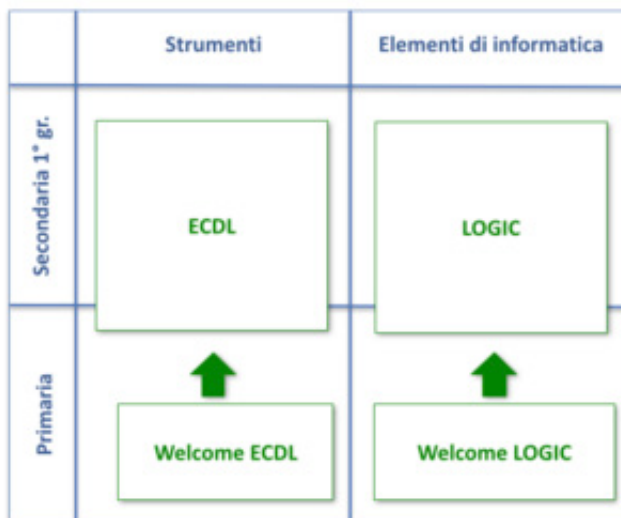


Figura 3 – La proposta AICA per gli Istituti comprensivi.

Il programma LOGIC

Il programma LOGIC è, in primo luogo, una proposta di formazione dei docenti affinché comprendano il valore pedagogico, meta-cognitivo, del *computational thinking* e di specifici "ambienti mediatori" ed acquisiscano le conoscenze e abilità necessarie per formare al *computational thinking* i propri allievi. (L'espressione "Ambienti Mediatori" è stata espressamente coniata al fine di inquadrare l'impiego di hardware e software a matrice robotica e afferente specifici linguaggi di programmazione in un contesto educativo e strategico più ampio rispetto all'ambito della cosiddetta "Robotica Educativa").

I docenti potranno successivamente progettare Ambienti di apprendimento – adeguati all'età degli allievi – in cui far crescere le loro capacità di *problem posing* e *problem solving* e in cui sviluppare abilità di coding. Il programma LOGIC propone una metodologia basata sulla didattica dell'errore e sul costruzionismo di Simon Papert, l'inventore di LOGO che è uno degli ambienti di apprendimento proposti.

Il programma LOGIC assume come punto di partenza la Robotica Educativa con il carattere prevalentemente laboratoriale delle attività proposte e con le caratteristiche delle risorse impiegate (piccoli robot programmabili in modalità "stand-alone" o attraverso un computer). Propone l'impiego di risorse di facile reperimento (per la maggior parte ambienti open, gratuiti e forniti direttamente ai docenti). Sollecita negli studenti la dovuta attenzione a quel complesso di attività logico-procedurali a matrice informatica che lo pongono in grado di " porsi e risolvere problemi" prescindendo dal contesto disciplinare di riferimento, perseguendo così l'obiettivo-chiave dell'"Imparare a imparare" che è un'altra delle otto competenze chiave per l'apprendimento permanente.

I moduli LOGIC

Il programma LOGIC è composto da una serie di Moduli (a ciascuno dei quali corrisponde un esame). I 5 moduli riferiti a specifici ambienti – Bee bot, Drape, Logo, Robomind e Snap! – sono indifferenziati, mentre il Modulo A è riservato ai docenti e il modulo B agli studenti.

Questo l'elenco dei moduli e degli esami:



Figura 4 – I Moduli (e gli esami) LOGIC.

Modulo A: Teorie di riferimento (solo per i docenti)

Il modulo, destinato ai Docenti già in possesso del titolo “Teacher” , analizza quali siano i fondamenti teorici che giustificano e inquadrano l’impiego a fini educativi e didattici le attività proposte in materia di *coding* e *computational thinking*. Pone le basi per un corretto approccio alle attività proposte in seno ai restanti moduli, inquadrandole sotto il profilo educativo e costituendone il contesto di riferimento.

L’esame certifica nel candidato il possesso di adeguate conoscenze in ordine ai seguenti argomenti:

- Syllabus Ministeriale del 2010: quali sono le finalità del documento, quale la sua struttura e ripartizione, quali sono gli “Elementi di informatica” indicati e da tenere in considerazione nei diversi ordini di Scuola per una corretta impostazione dell’attività didattica.
- Teorie dell’Apprendimento: quali sono, in cosa consistono e si differenziano fra loro, quali sono le caratteristiche dei diversi “artefatti cognitivi” di cui si può disporre nell’impostare l’azione educativa e didattica ecc.
- Teoria degli Algoritmi: cosa sono, come si realizzano, in cosa consistono le loro caratteristiche, come procedere alla loro progettazione e realizzazione, come applicarli in situazioni problematiche.

Modulo B: Algoritmi e problem solving (solo per gli studenti)

Il modulo è centrato sulla struttura e realizzazione di un algoritmo e ne esamina le possibili applicazioni in ambito matematico, dalle "macchine di calcolo" al concetto di relazione e funzione.

Si consiglia di far affrontare agli studenti questo modulo, più astratto, dopo aver concretamente sviluppato algoritmi nell'ambito di altri moduli.

L'esame certifica nel candidato il possesso di adeguate conoscenze e abilità in ordine ai seguenti argomenti:

- Gli Algoritmi: come riconoscere, analizzare, impostare un algoritmo, quali sono le proprietà che lo caratterizzano e la simbologia utilizzata; come utilizzare un algoritmo per descrivere e impostare processi o risoluzione di problemi, come individuare e rimuovere gli errori presenti in un algoritmo dato, ecc.
- Il *Problem Solving*: come individuare situazioni problematiche di diversa natura, analizzarle e pervenire alla soluzione; come impostare in forma algoritmica la risoluzione di un problema di tipo matematico, come porsi un problema, come completare un algoritmo di calcolo in presenza di dati mancanti, ordinare la sequenza di operazioni utile a risolvere un problema ecc.

Modulo 1: Bee Bot

Il Bee bot è un piccolo robot a forma di ape, programmabile per mezzo di pochi pulsanti presenti sul dispositivo stesso e in grado di memorizzare una sequenza di 40 istruzioni. Focus on Bee bot è un equivalente ambiente software. Se ne propone l'uso ai docenti della Scuola dell'Infanzia o dei primi anni della Primaria.

Il bambino, proiettandosi nell'oggetto manipolato e controllandone i movimenti, impara gradualmente a impostare corretti rapporti con lo spazio circostante, orientandosi rispetto ai punti di riferimento che questo offre. Contestualmente, impara ad analizzare le fasi del movimento, familiarizza con i concetti di direzione, verso e rotazione, progetta percorsi e imposta correttamente rapporti e relazioni topologiche, utili per qualsiasi forma di dominio dello spazio, sia esso "percettivo" che "rappresentativo".

L'esame certifica nel candidato il possesso di adeguate conoscenze e abilità in ordine ai seguenti argomenti:

- *Hardware*: quali sono le sue caratteristiche e funzioni, come impiegare i tasti di controllo, come far muovere il dispositivo in percorsi

preventivamente strutturati.

- *Software*: quali sono le caratteristiche dell'ambiente virtuale, come utilizzare opportunamente le diverse funzioni e voci di menu, come utilizzare opportunamente i diversi strumenti per creare nuovi percorsi ecc.

Modulo 2: Drape

DRAPE è un software espressamente concepito per favorire un primo approccio alla programmazione. Propedeutico all'ambiente LOGO, è idoneo ad essere impiegato sin dalla Scuola Primaria, e propone un ambiente di programmazione in cui istruzioni e procedure sono costituite da icone, corrispondenti alle diverse funzioni utili a far muovere la "Tartaruga" nel suo spazio operativo. Il software dispone di tre livelli di difficoltà (facile, medio, avanzato), cui corrispondono funzioni sempre più complesse che pongono in campo le diverse primitive e procedure di cui è formato.

Tramite DRAPE è possibile, inoltre, realizzare procedure che includono a loro volta sotto procedure per la realizzazione di figure anche molto complesse.

L'esame certifica nel candidato il possesso di adeguate conoscenze e abilità in ordine ai seguenti argomenti:

- *Analisi dell'ambiente operativo*: come sono strutturate le diverse sezioni in cui è suddiviso, a cosa servono le diverse icone presenti nella barra strumenti, come funzionano e a cosa servono i pulsanti presenti ai diversi livelli del programma.
- *Operatività ai diversi livelli*: come procedere per elaborare una procedura, associandola correttamente al risultato della stessa, come "leggere" un programma espresso in pseudocodice traducendolo in icone (e viceversa), ecc.
- *Procedure con subroutine*: come descrivere e riconoscere i vantaggi derivanti dall'impiego, in una procedura, di subroutine, come realizzare una subroutine e utilizzarla per la realizzazione di disegni complessi ecc.

Modulo 3: Logo

Creato, negli anni 60, da Seymour Papert presso il Laboratorio di Intelligenza Artificiale del MIT di Boston, LOGO può essere definito il primo esempio di "artefatto cognitivo" in grado di simulare il comportamento di un robot (la "Tartaruga") in un ambiente virtuale (il "micromondo").

Il linguaggio utilizzato, formalmente rigoroso ma nello stesso tempo di semplice intuizione, si caratterizza per la capacità di "apprendere" attraverso la costruzione di procedure che "istruiscono" il robot su cosa fare in presenza di una specifica istruzione.

L'esame certifica nel candidato il possesso di adeguate conoscenze e abilità in ordine ai seguenti argomenti:

- Caratteristiche del robot e ambiente operativo: come è nato e si evoluto l'ambiente LOGO, quali sono i suoi tratti peculiari rispetto ad altri ambienti di programmazione, come si utilizza la finestra comandi, quali sono le sue principali funzioni ecc.
- Operatività di base: quali sono le regole della sintassi LOGO, quali i comandi primitivi e le funzioni di base utili a far muovere la "Tartaruga", come creare una semplice figura geometrica scoprendone le proprietà, come muoversi utilizzando la mappa cartesiana ecc.
- Programmi e applicazioni: come procedere nella verifica di una procedura, eliminando gli errori che non ne consentono il corretto funzionamento, come utilizzare una procedura nella procedura (routine e subroutine), come realizzare disegni anche complessi procedendo alla loro preventiva progettazione ecc.

Modulo 4: Robomind

Con "Robomind" – realizzato all'Università di Amsterdam – l'allievo si cimenta nel governo di un vero e proprio robot virtuale, dotato di "sensori" e "attuatori" che gli consentono di interagire con un ambiente articolato e irto di ostacoli: opportunamente programmato, "Robo" si muove in ogni direzione, afferra e rilascia oggetti, trova la via d'uscita in un labirinto, disegna per terra righe bianche o nere di cui si serve per seguire un percorso, riuscendo a "percepire" il colore della guida tracciata.

L'esame certifica nel candidato il possesso di adeguate conoscenze e abilità in ordine ai seguenti argomenti:

- Caratteristiche del robot e ambiente operativo: come riconoscere e impiegare correttamente le funzioni connesse ai diversi menu, come riconoscere e impiegare correttamente le diverse sezioni della finestra operativa, come utilizzare le diverse opzioni di visualizzazione, come utilizzare efficacemente i controlli manuali e i sensori virtuali del robot, ecc.
- Operatività di base: come scrivere una semplice procedura, come

inserire in un programma dato un ciclo, una condizione, una procedura, come programmare il robot in modo che "percepisca", seguendola, una linea bianca, ecc.

- Programmi e applicazioni: come far muovere il robot in un ambiente evitando ostacoli o effettuando scelte, come generare scelte casuali, come analizzare un programma dato scoprendo e correggendo eventuali errori che non ne consentono il corretto funzionamento, ecc.

Modulo 5: Snap! By Scratch

Soap! è un ambiente di programmazione derivato da Scratch, da cui mutua la grafica e la struttura "a blocchi" che evolve verso forme di programmazione più complesse. Da sottolineare l'impiego di uno "Sprite" (ovvero di un robot virtuale) a forma di cursore che ricorda molto da vicino la "Tartaruga" LOGO, e la possibilità offerta all'Allievo di creare i propri blocchi di comando, vere e proprie subroutine, da impiegare nei propri listati.

Non inganni l'interfaccia grafica, molto amichevole e accattivante: SNAP! si rivela in grado di condurre l'allievo verso forme di programmazione molto complesse, dando origine a prodotti la cui elaborazione non ha nulla da invidiare rispetto agli ambienti di programmazione professionali: anche in questo caso il procedimento euristico (per prove ed errori) si rivela il più efficace.

Le certificazioni LOGIC

Certificazione LOGIC Teacher

Per conseguire la certificazione Logic Teacher è necessario superare tre moduli a scelta, tra quelli indicati in blu nell'immagine sottostante.

Per gli insegnanti della scuola dell'Infanzia e della Scuola primaria si consigliano il modulo 1, Bee Bot, Drape o Logo.

Agli insegnanti della scuola Secondaria si suggerisce di tralasciare il modulo Bee bot.

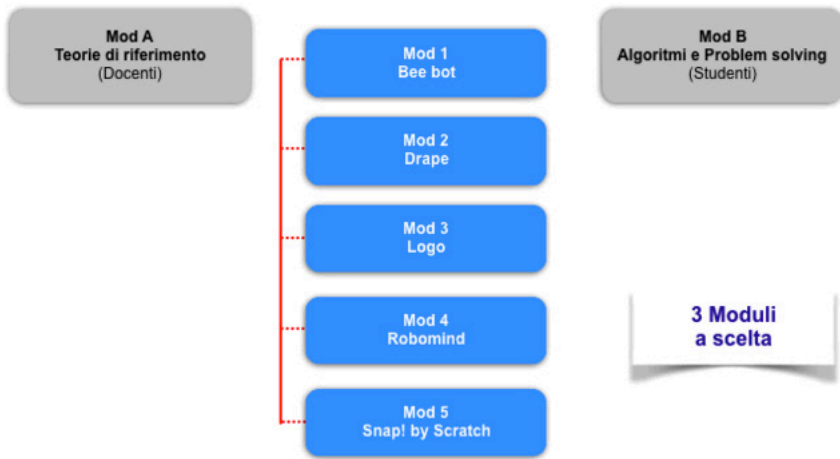


Figura 5 - La certificazione LOGIC Teacher.

Certificazione LOGIC EXPERT

La certificazione LOGIC Expert è rilasciata a chi, oltre agli esami necessari per la certificazione Teacher, supera l'esame del modulo A.

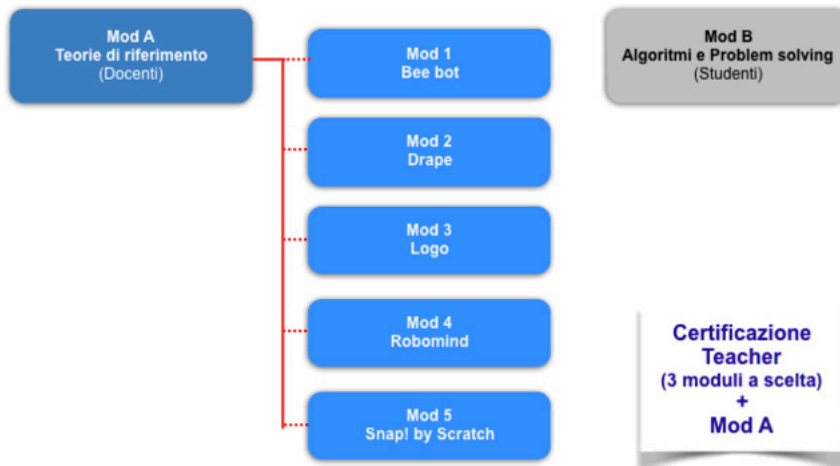


Figura 6 - La certificazione LOGIC Expert.

Certificazione LOGIC Student

La certificazione LOGIC Student viene rilasciata al superamento di due esami – scelti fra i moduli 2, 3, 4 e 5 – più l'esame relativo al Modulo B.

(Si suggerisce ai Docenti di dare la precedenza ai moduli operativi prescelti, riservando gli approfondimenti previsti dal modulo B alla parte finale del percorso di formazione, in considerazione del fatto che quest'ultimo richiede maggiore capacità di astrazione e concettualizzazione dei contenuti analizzati).

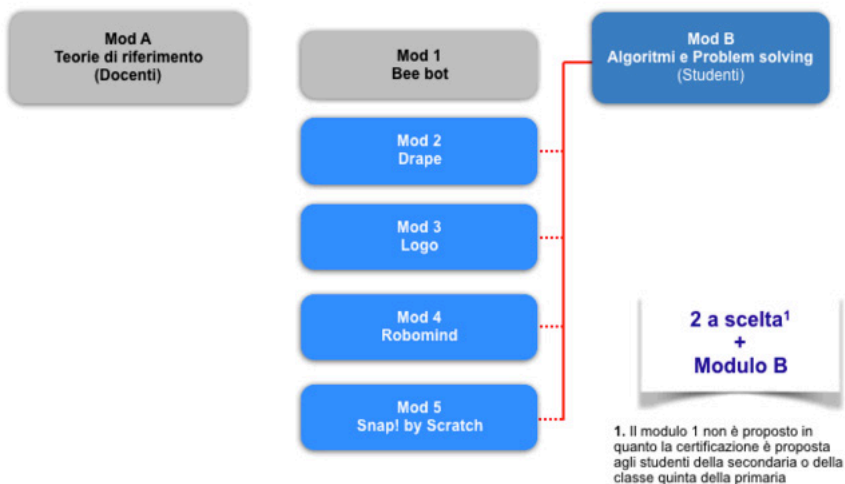


Figura 7 – La certificazione LOGIC Student.

I percorsi formativi per i docenti

Il Programma LOGIC prevede per i docenti un adeguato percorso di formazione, propedeutico al conseguimento della certificazione e funzionale al buon esito dell'azione educativa e didattica destinata agli allievi.

Al fine di garantire la qualità e l'efficacia dell'azione formativa, nonché il buon esito dell'esame finale, AICA, in collaborazione con ANFOR, mette a disposizione degli Insegnanti, e delle Scuole interessate alla corretta erogazione del Programma (aderenti alla rete LOGINET), corsi di formazione erogati da esperti preliminarmente certificati e inseriti in un apposito Registro Nazionale.

Gli esperti, oltre a coordinare gli incontri, forniranno ai corsisti adegua-

ta consulenza e assistenza per una corretta preparazione all'esame finale.

I corsi hanno la durata media di 30 ore, sono erogati in presenza e supportati da materiali eLearning. I corsi hanno carattere prevalentemente laboratoriale e prevedono, fra l'altro, momenti di verifica e simulazioni, oltre alla formulazione di ipotesi progettuali per la corretta impostazione dell'attività didattica. A ulteriore supporto dell'azione formativa, è prevista anche la partecipazione a Seminari Web (Webinar), condotti da esperti e tenuti a gruppi di docenti con il supporto di tutor, con l'obiettivo di avviare ulteriori momenti di approfondimento sui contenuti e sulle strategie oggetto dei corsi.