



UNIVERSITÀ
DEGLI STUDI
FIRENZE

**Scuola di
Studi Umanistici
e della Formazione**

Corso di Laurea in
Scienze della Formazione Primaria N.O.

Il coding nella scuola primaria

**Uno studio sull'utilizzo della
programmazione per lo sviluppo dei
concetti geometrici**

Relatore

Prof. Andreas Robert Formiconi

Candidata

Claudia Giannelli

Abstract

Il pensiero computazionale è un argomento che oggi, nella scuola, acquisisce sempre più importanza. Nuovi metodi e nuovi strumenti sono utilizzati come ausili per l'apprendimento e come veri e propri oggetti per apprendere, grazie alle capacità di sviluppo di conoscenze, abilità e competenze necessarie alla crescita della persona. Questo studio sul coding si inserisce nel contesto di una ricerca più ampia, con lo scopo di contribuire a rendere il campione più rappresentativo. Le finalità della ricerca sono quelle di confrontare l'utilizzo dei due linguaggi di programmazione Logo e Scratch per valutarne gli effetti nella didattica, sull'apprendimento della geometria e per evidenziare le eventuali differenze tra il linguaggio testuale e quello visuale a blocchi. I risultati hanno mostrato una evidente preferenza nell'utilizzare Scratch, per la sua impostazione grafica e per la praticità. Per quanto riguarda la comprensione della programmazione e la consapevolezza delle proprie abilità nell'utilizzo del computer, Logo si è rivelato invece il linguaggio con i maggiori effetti positivi.

Abstract

Computational thinking is an increasingly important topic in school today. New methods and tools are aids for learning and also objects of learning, thanks to their abilities to develop skills and competences that are fundamental for the personal growth. This study about coding takes part of a broader research with the aim to expand the sample and make it more representative. The goal of the research is to compare the use of two programming languages, Logo and Scratch, to evaluate their effects on teaching, on learning geometric concepts and to find the differences between text-based and block-based languages. The results showed the preference of using Scratch, thanks to its design and practicality. However, as for the understanding of programming and the awareness of the ability in the use of computer, Logo is the language with most positive effects.

Indice

Introduzione	7
Capitolo 1	11
Coding: la programmazione e il pensiero computazionale a scuola	11
1.1 Tecnologie e apprendimento: l'educazione nell'era dei nativi digitali	11
1.2 Il coding e il pensiero computazionale.....	16
1.3 Seymour Papert e il Costruzionismo	17
1.4 Le influenze di Dewey e Piaget.....	20
1.5 Pensiero computazionale e Competenza Digitale	22
1.6 Il coding come attività trasversale: Indicazioni Nazionali e Nuovi Scenari	25
1.7 L'importanza del coding a scuola	26
1.8 Pensiero computazionale e coding nella realtà scolastica italiana	29
1.8.1 Programma il futuro.....	31
1.8.2 CodeMOOC	31
1.9 Le proposte internazionali	32
1.9.1 CodeWeek.....	33
1.9.2 Code.org.....	34
1.9.3 CoderDojo, una esperienza extra-scolastica	35
1.10 Strumenti per il coding	35
Capitolo 2.....	41
Linguaggi di programmazione	41
2.1 Linguaggio di programmazione: una definizione	41
2.2 LOGO: un linguaggio di programmazione testuale	45
2.2.1 LibreLogo, programmazione con software libero	46
2.3 Scratch e il linguaggio di programmazione visuale	48
2.4 La metodologia progettuale del bricolage e i micromondi.....	49
2.5 L'epistemologia dell'errore	51
2.6 L'apprendimento della geometria attraverso la programmazione	53
2.7 Linguaggi visuali e testuali a confronto	56
2.7.1 Lo studio di David Weintrop e Uri Wilensky.....	57
2.7.2 Lo studio di Lewis: Logo vs. Scratch	59
Capitolo 3.....	61
Un laboratorio di coding alla scuola primaria.....	61

3.1 Il progetto.....	61
3.2 Finalità	61
3.3 Contesto	62
3.4 Descrizione delle attività	63
3.5 Progettazione delle attività.....	63
3.6 Test di geometria	80
3.6.1 Descrizione del campione	81
3.7 Risultati dei test	82
3.8 Logo e Scratch a confronto: i risultati	101
3.9 Presentazione del questionario.....	102
3.9.1 Analisi dei risultati del questionario	103
3.10 Interpretazione dei risultati	112
Conclusioni.....	115
Bibliografia.....	119
Sitografia	124
Appendice.....	125
Test 1.....	127
Test 2.....	133
Test 3.....	138
Questionario.....	144

Introduzione

Negli ultimi anni, il coding è entrato a far parte delle attività didattiche in tutti gli ordini di scuola, a partire da quella dell'infanzia. Il pensiero computazionale ha ottenuto un'importanza sempre maggiore, tanto che viene promosso allo stesso pari delle altre competenze di base per lo sviluppo della persona. Oggi i progetti a livello mondiale coinvolgono un numero sempre maggiore di realtà scolastiche, promuovendo l'attività di programmazione sotto diversi aspetti: corsi online, attività di coding *unplugged*, software di programmazione di basso livello, accessibili da tutti e per ogni età. Gli strumenti disponibili sono molteplici, come lo sono le loro possibilità di utilizzo. Gli studi e le ricerche su questo argomento trattano gli effetti diretti di questa pratica, non solo sullo sviluppo di capacità di problem solving, ma anche a livello strettamente didattico e per quanto riguarda l'apprendimento di pratiche informatiche. Questo lavoro nasce come contributo a una ricerca più ampia, con lo scopo di ampliarne il campione per un riscontro quanto più evidente e preciso. Le motivazioni che hanno spinto a svolgere uno studio su questo argomento sono molteplici: innanzitutto la passione per l'argomento nata grazie al corso svolto all'università, tenuto dal Professor Formiconi, il Laboratorio di Tecnologie Didattiche; inoltre è stato un valido motivo la possibilità di prendere parte ad uno studio già avviato e sperimentato in diverse scuole, vista l'opportunità di introdurre nelle classi strumenti come il computer e la programmazione informatica, validi ausili che negli istituti meno all'avanguardia sono molto poco utilizzati e sfruttati per le possibilità che offrono.

La finalità del progetto è quella di valutare gli effetti della programmazione, nei bambini di scuola primaria, per quanto riguarda l'apprendimento di concetti geometrici come l'angolo; inoltre vengono messi a confronto due ambienti di programmazione diversi, basati uno su un linguaggio testuale e l'altro su un linguaggio visuale. La ricerca si è ispirata in primis allo studio di Colleen Lewis, la quale ha sperimentato l'utilizzo dei due linguaggi di

programmazione *Logo* e *Scratch*, confrontandoli per valutarne l'efficacia e identificare le principali differenze; in secondo luogo, gli studi degli autori Weintrop e Wilensky sono stati un riferimento per quanto riguarda la ricerca di differenze significative tra l'utilizzo di un linguaggio rispetto all'altro per i principianti. Le ipotesi che si intendono verificare riguardano i vantaggi del linguaggio a blocchi rispetto a quello testuale, cercando quindi di capire le motivazioni principali per cui questo linguaggio risulta più semplice e preferito dagli studenti. I risultati inaspettati ricavati dagli studi finora effettuati evidenziano che, nonostante la maggiore facilità di programmazione con il linguaggio visuale, gli svantaggi evidenziati rispetto ad un linguaggio testuale riguardano la possibilità di poter creare con quest'ultimo programmi più complessi e personalizzati, percepiti come più "veri". Inoltre il grado di confidenza nelle proprie capacità di programmazione e di utilizzo di determinati strumenti si è rivelato maggiore negli studenti che hanno utilizzato il linguaggio testuale. Vengono richiamate inoltre le ricerche effettuate dagli autori Clements e Battista e Noss, relative agli effetti della programmazione sull'apprendimento della geometria. Da queste sono stati ricavati gli *items* utilizzati per rilevare i dati dello studio.

Nel primo capitolo sono descritti i concetti di coding e pensiero computazionale, partendo da un quadro generale sull'utilizzo degli strumenti informatici e in generale delle tecnologie in ambito educativo. Si parla dei presupposti teorici e pedagogici che hanno favorito l'ingresso di queste pratiche nel mondo dell'istruzione, partendo da Seymour Papert, il padre di LOGO e della programmazione "didattica", per arrivare al Costruzionismo e alle filosofie che lo hanno influenzato. Inoltre questo capitolo spiega come ha agito la scuola italiana per far fronte a questa nuova e importante competenza, necessaria visto l'ingresso della tecnologia in ogni aspetto della vita delle persone: le istituzioni, a livello internazionale, favoriscono l'utilizzo delle pratiche di coding nelle scuole, danno indicazioni sulle migliori strategie da utilizzare e offrono gli strumenti adatti. I progetti e le campagne di alfabetizzazione, come corsi online rivolti agli insegnanti, sono molteplici e accessibili gratuitamente. Vengono descritti nello

specifico in questo capitolo, insieme agli strumenti pratici dei quali è utile farsi un'idea per poterne sfruttare le potenzialità nell'insegnamento.

Il secondo capitolo si concentra in modo specifico sulla programmazione informatica e sulle sue caratteristiche. Si approfondisce l'analisi di linguaggi semplici e adatti all'utilizzo nelle scuole, come il linguaggio testuale di LOGO e quello grafico a blocchi di Scratch; i due ambienti di programmazione vengono analizzati in un'ottica positiva di aiuto e motivazione all'apprendimento, grazie alle caratteristiche proprie dello scrivere un codice, per tentativi, ricercando l'errore, in un *micromondo* che simula il mondo reale e che stimola così la ricerca di soluzioni a problemi reali. In questo capitolo inoltre vengono descritti alcuni studi e ricerche effettuate per verificare l'efficacia di questa pratica nell'apprendimento degli studenti, per quanto riguarda in modo specifico i concetti della geometria e per rilevare le differenze tra le due modalità di costruzione del codice.

Il terzo ed ultimo capitolo tratta nello specifico la sperimentazione effettuata nella scuola primaria, relativa all'esperienza di coding con Logo e Scratch. Le due classi quinte hanno svolto un laboratorio sulla programmazione, lavorando una con il linguaggio testuale e l'altra con quello visuale, per poi invertirsi i ruoli a metà del progetto. Durante il percorso sono stati utilizzati dei test, somministrati all'inizio, a metà e alla fine dell'esperienza, per valutare i risultati conseguiti; le domande, relative agli angoli sono state formulate in modo da verificare la presenza e l'eventuale eliminazione di alcune comuni misconcezioni che si creano negli alunni nell'apprendimento della geometria; i quesiti, molto simili o uguali per tutti e tre i test, hanno consentito di confrontare le risposte in modo chiaro e le differenze, sia tra un gruppo e l'altro, sia dall'inizio alla fine del laboratorio. Inoltre un questionario è servito per avere un riscontro sulle percezioni e impressioni degli alunni, oltre che per avere una stima dell'utilizzo delle tecnologie fuori dalla scuola. Nella prima parte del capitolo sono illustrate le finalità del progetto, la sua progettazione e la descrizione delle attività svolte. Successivamente i dati ottenuti vengono raccolti in grafici e tabelle e vengono analizzati i risultati.

Capitolo 1

Coding: la programmazione e il pensiero computazionale a scuola

1.1 Tecnologie e apprendimento: l'educazione nell'era dei nativi digitali

L'avvento tecnologico avvenuto negli ultimi 30 anni, ha dato inizio ad un periodo storico caratterizzato dalla diffusione degli strumenti tecnologici digitali, definito Era Digitale. Le infinite possibilità di comunicazione e di accesso all'informazione hanno portato ad un cambiamento economico, politico e sociale. I media digitali giocano un ruolo importante nella formazione dell'identità sociale e personale, per questo chi è nato negli ultimi trent'anni si differenzia nettamente dai genitori, non solo nell'essere e nel comportarsi, ma anche nel pensare. [cfr. Ranieri, pp.67-68]. I «nativi digitali»¹ sono profondamente diversi, dal punto di vista antropologico, rispetto alla generazione che li precede: costantemente immersi nella tecnologia, sarebbero dotati di nuove capacità cognitive, legate all'uso dei molteplici strumenti dell'era digitale (Computer, Internet, telefoni cellulari, videocamere, lettori MP3). Secondo Prensky, fautore della teoria dei *Digital Natives*, lo spartiacque generazionale è da collocare a metà degli anni Ottanta, quando il computer si è diffuso nei paesi industrializzati; altri autori individuano il 1995 come l'anno di nascita decisivo [Ferri, 2008]. Anche se le opinioni sono molteplici e discordanti, è condivisa l'idea che l'immersione totale nella tecnologia abbia trasformato gli stili cognitivi e di apprendimento delle nuove generazioni in modo permanente e sempre più incisivo. Questa forma nuova di essere e di pensare rimane lontana da chi, nato prima degli anni '80, si è invece dovuto adattare alle innovazioni tecnologiche, i cosiddetti «immigrati digitali», che mostrano ancora una certa diffidenza verso quei dispositivi, diventati indispensabili, ma per loro ancora poco comprensibili. Nascere nell'era digitale implica un apprendimento

¹ Da *Digital Natives*, espressione coniata da Prensky nel 2001, definisce la generazione di chi è nato nell'era digitale. Altri autori definiscono questa generazione Millennials, New Millennium Learners, Net Generation.

spontaneo dell'utilizzo della tecnologia, naturale come l'acquisizione del linguaggio, per questo i Millennials sarebbero tecnicamente più abili, ma anche più scaltri, grazie alla familiarità che si crea con le infinite informazioni a cui hanno accesso, capaci di imparare in modi sempre nuovi e veloci, divertenti e interattivi.

Nel dibattito sulla *Net Generation*, non manca chi mette in discussione le tesi fino ad ora citate, come i cosiddetti «*Net Gen scettici*»; secondo le loro teorie, non si può parlare di «generazione» caratterizzata in modo omogeneo da abilità tecnologiche e competenze digitali; studi e ricerche hanno fatto emergere un quadro molto più complesso del rapporto tra nuove generazioni e tecnologia. Il primo dato emergente è quello dell'accesso tecnologico: il cosiddetto *digital divide* (divario digitale) mostra le disuguaglianze digitali, nella possibilità di accesso tecnologico e accesso a Internet, esistenti tra i paesi (economicamente più arretrati e fragili) e all'interno di un paese (individui di fascia economica e sociale più svantaggiata) [Bennet e Maton, 2010, p. 323]. Altre indagini, come gli studi di Kennedy et al. (2010), si concentrano sulla frequenza dell'uso delle tecnologie, classificando gli studenti in quattro diverse tipologie: gli utenti avanzati, che utilizzano frequentemente un'ampia varietà di dispositivi e tecnologie; gli utenti ordinari, che usano regolarmente il web e le tecnologie mobili; gli utenti irregolari, che ne fanno un uso meno frequente rispetto a quelli ordinari; gli utenti di base, che, ad eccezione del cellulare, usano raramente le tecnologie comuni. Le statistiche mostrano che la maggior parte degli studenti si avvale di uno spettro limitato di tecnologie e che non appartengono alla fascia degli utenti avanzati. Altri studi che supportano le teorie dei *Net Gen scettici*, si concentrano sui fattori che influiscono sui livelli di adozione delle tecnologie. Bennet et al. (2008) affermano che non ci sono differenze significative tra i soggetti rispetto all'età anagrafica, ma i fattori che influiscono sono diversi, come il genere, lo status socio-economico, la collocazione geografica, il background culturale. Un aspetto importante su cui è posta l'attenzione, è quello della relazione tra l'uso intensivo delle tecnologie e sviluppo di abilità e competenze digitali. Contrariamente a quanto affermato in precedenza da Prensky, le ricerche dimostrano che i giovani in realtà non sarebbero in grado di interrogarsi sulla validità dei contenuti, che la mancanza di riflessione critica ne limita la capacità di valutarne l'affidabilità.

Un'indagine effettuata da Calvani, Fini e Ranieri (2010) mostra che è importante definire bene il concetto di competenza digitale: se per essa si intendono le semplici abilità tecniche, moltissimi giovani sono alfabetizzati dal punto di vista tecnologico; se invece si intendono capacità cognitive ed etico-sociali, i risultati sono deludenti, in quanto ci troviamo di fronte a scarsi livelli di competenza. Nell'ottica dei risultati degli studi finora descritti, si può concludere quindi che non esista una generazione di nativi digitali, che essi siano solo una minoranza e che l'opposizione tra nativi e immigrati digitali sia scorretta. Recentemente, lo stesso Prensky ha rivisto la sua posizione rispetto a questo tema, parlando adesso di «saggezza digitale»², intesa come capacità di usare in modo critico e responsabile le tecnologie digitali, slegata dai fattori generazionali e indipendentemente dall'età. Essa viene considerata un obiettivo educativo e non più una condizione connaturata di essere nati nell'era digitale. Questa affermazione è in netto contrasto con le attuali proposte educative delle istituzioni formative, non più in grado di accogliere le sfide del presente. I giovani sono ben consapevoli dell'esistenza di una «discontinuità digitale» tra scuola e extra-scuola e questa situazione risulta accettata dagli studenti, è chiaro però che in questi sono subentrate nuove esigenze, per affrontare questo diverso modo di essere e di pensare, esigenze ancora insoddisfatte dalla scuola, che necessita oggi di una riconfigurazione radicale, se non di una e vera e propria trasformazione.

Negli ultimi decenni, il mondo dell'educazione si è interrogato sull'effettiva efficacia dell'utilizzo delle ICT³ nell'apprendimento. Al centro del dibattito contemporaneo sulle tecnologie nell'educazione rimane il computer [Capponi, 2009, p14] e l'apporto che esso può dare ai processi di insegnamento e apprendimento [cfr. Antonietti, Cantoia, 2001, pp. 33-35]. Le ICT sono state introdotte nella scuola tra gli anni '70 e '80 e hanno portato con sé un'ondata di entusiasmo: da un lato l'uso del computer a scuola veniva visto come un modo per sviluppare quelle abilità tecnologiche essenziali per essere attivi nella società del

² Prensky (2009) distingue tre livelli di expertise digitale: il primo è la «saggezza digitale», il secondo l'«abilità digitale», tipica di un soggetto che possiede molta dimestichezza e familiarità tecnica con i vari dispositivi digitali, il terzo la «stupidità digitale» che caratterizza coloro che usano le tecnologie in modo improprio, dannoso e irresponsabile e i catastrofisti, ovvero chi si rifiuta a priori di avvicinarsi alle tecnologie, considerate all'origine di tutti i mali.

³ La sigla ICT indica le Tecnologie dell'Informazione e della Comunicazione.

futuro; dall'altro veniva visto come strumento in grado di promuovere il pensiero critico e la creatività. Nonostante questo generale ottimismo, l'utilizzo del computer non è entrato a far parte delle pratiche ordinarie di insegnamento, ma è rimasto sporadico e incoerente. Le critiche mosse contro l'utilizzo dei dispositivi digitali in ambito didattico sono state altrettanto numerose, sopra tutte la probabilità che le ICT favoriscano un abbassamento qualitativo degli apprendimenti, con una scuola più piacevole e allettante ma anche più povera di teoria, nella quale diminuisce la riflessività, l'interiorizzazione, la capacità di prospettare visioni ampie e rifiutando fatica e impegno [cfr. Calvani, 1999]. Sicuramente, la scarsa familiarità tecnologica degli insegnanti ha frenato la rivoluzione tecnologica della scuola. Questa categoria è stata fino a poco tempo fa e in alcune istituzioni scolastiche lo è tuttora, una generazione di immigrati digitali, caratterizzati da passività e standardizzazione e da un linguaggio tradizionale sempre più lontano dagli allievi odierni. Per riprendere le parole di Papert, il computer, all'interno del sistema scolastico, per almeno i primi vent'anni, è passato ad essere da strumento di cambiamento a strumento conservatore, rafforzando i tradizionali metodi scolastici: invece di superare le barriere tra le materie, è diventato esso stesso una materia. Così l'autore spiega il fenomeno:

"[...] il passaggio da strumento radicalmente sovversivo in classe a inoffensivo strumento conservatore in laboratorio di informatica non è imputabile alla mancanza di conoscenze né alla mancanza di software. Lo attribuisco invece all'innata intelligenza della scuola, che ha agito come un qualsiasi organismo vivente che si difende da un corpo estraneo [...] una reazione immunitaria il cui obiettivo finale era di digerire e di assimilare l'intruso" (Papert, 1993, p. 52)

Al di là di tutto, il computer è quasi sempre visto come strumento per insegnare, piuttosto che un ambiente in cui apprendere. Allora perché utilizzarlo a scuola? Nell'ottica della possibilità di accesso, comprensione e memorizzazione di contenuti, le tecnologie possono diventare strumenti per l'apprendimento validi e efficaci. Basta pensare alle LIM e all'aiuto che essa può dare in classe, grazie

all'interattività, alla possibilità di proiezione di immagini in tempo reale e all'uso del web. Ma limitarsi a considerare le nuove tecnologie semplicemente come dei facilitatori impedisce di vedere le infinite possibilità che i calcolatori ci danno ogni giorno, anche nella didattica. Tra le più importanti, esso può servire per abbattere la cosiddetta «*matofobia*»⁴, la paura che esiste nella nostra cultura nei confronti dell'apprendimento e che secondo Papert è un atteggiamento appreso, non spontaneo; essa infatti incide negativamente sull'immagine che ognuno ha delle proprie capacità di apprendere. Ma "i computer, in generale, non producono automaticamente un beneficio educativo; affinché costituiscano un contributo autentico è necessario creare situazioni di apprendimento che vadano ben oltre l'idea di considerarli come strumenti per domande e risposte, per conservare e ottenere dati, o per calcolare e disegnare" [Reggini, 1982, p. 159]. Se si vuole che il suo utilizzo a scuola abbia un senso e che la tecnologia offra delle opportunità a chi apprende, il computer deve passare dalle mani del docente a quello dello studente, affinché egli possa compiere scoperte ed esperimenti in un personale ambiente di ricerca, per poter essere in grado di costruire attivamente le proprie conoscenze ed esplorare un microcosmo ricco di esperienze logiche, di simulazioni ed approfondimenti [cfr. Capponi, 2003].

"Noi non pensiamo che il computer dia qualcosa al bambino. Per fare un'analogia, se si vuole imparare la musica, è bene suonare uno strumento. Cosa può dare un pianoforte a qualcuno, che non gli può dare un libro? È la stessa risposta. Il pianoforte consente di fare qualcosa con la musica, di renderla propria, di esprimere se stessi. Nel libro si può leggere sulla musica, ma non è la stessa cosa. Con la conoscenza matematica, il computer è come il pianoforte. Consente di suonare la conoscenza; il libro ce la può solo dare." (Papert, 1997)

Sebbene sia innegabile che oggi i ragazzi siano sempre più abili ad utilizzare le tecnologie, non è scontato che le sappiano comprendere. È possibile che ne facciano un uso passivo, scorretto, che le subiscano. L'alfabetizzazione informatica, anche se sembra sempre meno necessaria, è indispensabile per

⁴ Termine coniato da Seymour Papert per indicare la paura di apprendere, nello specifico la paura della matematica.

crescere persone che sappiano utilizzare le tecnologie in modo consapevole e critico, perché sappiano controllarle e non si lascino controllare da esse.

1.2 Il coding e il pensiero computazionale

Nella realtà scolastica odierna si parla sempre di più di *pensiero computazionale*, come processo mentale da promuovere, insieme alle competenze fondamentali che ognuno deve possedere. Il concetto è stato introdotto per la prima volta da Seymour Papert in *Mindstorm*, nel 1980, ma una definizione più condivisa è stata formulata da Jeannette Wing nel 2006, nel suo articolo "*Computational Thinking*": il pensiero computazionale è il processo di pensiero coinvolto nella risoluzione di problemi, nella progettazione di sistemi e nella comprensione del comportamento umano, attingendo ai principi fondamentali dell'informatica; si basa sul potere e sui limiti dei processi informatici, siano essi eseguiti da un essere umano o da una macchina, consente di riformulare un problema apparentemente difficile in uno che sappiamo come risolvere, utilizzando meccanismi di astrazione e di scomposizione. Un'altra definizione descrive il pensiero computazionale come la capacità di concepire ed esprimere procedimenti che portino a risolvere un problema attraverso la concatenazione di passi elementari che possono essere eseguiti da qualcun'altro, che sia un essere umano o un computer [Bogliolo, 2016]. Questa modalità di pensiero intesa come competenza fondamentale, necessita ormai di essere appresa e esercitata come tutte le altre competenze, fin dai primi anni di scolarizzazione. È opportuno quindi introdurre il concetto di *Coding* come strumento di apprendimento, attività che sta alla base di tutti questi processi e nella quale il pensiero computazionale può trovare un ampio sviluppo. Con il termine "coding" si intende l'uso di strumenti e metodi di programmazione [Bogliolo, 2016, p. 13]; codificare, appunto, significa scrivere un codice, una sequenza di dati, istruzioni scritte in un linguaggio e in un modo che il computer riesce a comprendere ed eseguire. La programmazione informatica, intesa come il lavoro dell'informatico, è un processo che parte da un problema da analizzare, quindi è necessario pensare un algoritmo, ovvero una sequenza di istruzioni per

risolvere il problema; dopo averne verificata la correttezza, questo deve essere tradotto in un linguaggio di programmazione, scrivendo un codice comprensibile ed eseguibile dal calcolatore. [Marchignoli, Lodi, 2016] Il termine coding non si riferisce solamente all'effettivo atto di scrittura del codice, ma si riferisce all'intero processo della programmazione, allo specifico modo di ragionare e di risolvere i problemi che caratterizza il pensiero computazionale che, come sottolinea la Wing nel suo articolo, non consiste semplicemente nel saper programmare, ma nel pensare a diversi livelli di astrazione, per questo è una competenza fondamentale per tutti, non solo per gli informatici.

1.3 Seymour Papert e il Costruzionismo

L'idea di inserire la programmazione informatica tra le attività educative è dovuta a Seymour Papert, matematico sudafricano esperto di cibernetica, di Intelligenza Artificiale, di psicologia e di educazione. Dopo un periodo di studio a Ginevra, durante il quale incontra e lavora con Jean Piaget, tra gli anni '60 e '70 si sposta negli Stati Uniti, dove viene assunto come ricercatore al MIT, il Massachusetts Institute of Technology, nel laboratorio di Intelligenza Artificiale, diventandone il co-direttore insieme a Marvin Minsky⁵. In questi anni si dedica alla creazione del linguaggio LOGO, un linguaggio di programmazione ideato appositamente per i bambini, costruendo la prima tartaruga robotica in grado di essere programmata attraverso comandi dati tramite il computer. Questa, muovendosi, creava dei disegni con un pennarello. Il programma è stato un grande successo e allo stesso tempo un fallimento: ci sono molte versioni di LOGO oggi sul web, ma purtroppo non è mai entrato nel mondo scolastico, dove, a parte qualche eccezione, è addirittura sconosciuto. Lo scopo di Papert è stato fin dall'inizio quello di creare un "ambiente" in cui imparare meglio la matematica e avere maggiore confidenza nella propria capacità di pensiero. La cosiddetta *Matofobia*, spiega il matematico in *Mindstorms*, non è mai stata una paura intrinseca all'individuo, ma una paura

⁵Marvin Minsky (1927-2016) è stato uno dei fondatori dell'Intelligenza Artificiale. Insieme a Papert ha contribuito all'invenzione del linguaggio di programmazione LOGO.

che si sviluppa successivamente; anche l'idea piagetiana dell'apprendimento spontaneo, spiega quell'attrazione per l'apprendimento tipica dei bambini, che in seguito si trasforma in fobia. Tra la cultura umanistica e scientifica, infatti, c'è una separazione netta, integrata ormai non solo nell'educazione, ma nella visione del mondo in generale. Molte persone sono ostacolate da opinioni negative sulle proprie capacità, identificandole nelle difficoltà personali; queste opinioni si rafforzano in un circolo vizioso, in quanto la convinzione di perdere porta all'insuccesso, che conferma e rafforza la convinzione di partenza. I bambini attribuiscono questi insuccessi alle proprie incapacità: in questa occasione nasce l'idea che ci siano persone portate per la matematica e altre per la linguistica, l'italiano o la grammatica. Papert spiega la dissociazione della matematica paragonando l'argomento a una lezione di danza "scritta", in cui ai bambini, anziché danzare, è richiesto di disegnare i passi di danza e svolgere operazioni con essi: questa situazione creerebbe sicuramente dei "danzofobi". Allo stesso modo, la Matofobia si sviluppa a causa della concezione della matematica come dissociata dagli eventi e dalle situazioni della realtà quotidiana, materia a cui non si riesce dare un senso, visto che difficilmente ha senso anche per chi deve insegnarla. Lo scopo di *Matelandia*, il paese in cui la matematica è una lingua naturale, è quello di eliminare la discrepanza tra ciò che è verbale e ciò che è matematico, facendo in modo che le attitudini verbali vengano usate per facilitare lo sviluppo matematico e che l'attitudine per la logica stimoli l'interesse per la linguistica. L'elaboratore diventa lo strumento in grado di abbattere il muro tra questi due mondi, apparentemente così diversi e lontani, conducendo i bambini a stabilire una relazione più umanistica con la matematica: esso permette un cambiamento nella didattica perché "parla matematica", per questo può aiutare a trovare quei legami tra ciò che è matematica e la realtà.

Nel pensiero di Papert l'impiego delle tecnologie a scuola è inquadrato in un solido contesto teorico, che si rifà alle teorie dell'apprendimento classiche, in particolare il Costruttivismo e la sua declinazione papertiana, detta *Costruzionismo*. L'obiettivo di Papert, quando ha creato LOGO è stato quello di usare il computer come ambiente per «imparare a imparare» e la programmazione come attività cognitiva indispensabile per esprimere progetti personali, per

costruire artefatti concreti e condivisibili. Programmare consente la comprensione e la manipolazione di simboli e codici comunicativi e questo porta l'allievo alla scoperta di nuove conoscenze in modo autonomo. Papert sostiene che *"insegnando all'elaboratore a pensare, i bambini si lanciano in una esplorazione del loro stesso modo di pensare"* e che *"riflettere, si pensa, trasforma il bambino in un epistemologo, un'esperienza che molti adulti non hanno mai vissuto"*. [Papert, 1980, p. 25]. In un'ottica costruttivista, nell'interazione tra l'allievo e la tecnologia, il controllo e l'iniziativa devono appartenere a chi apprende: non deve essere più il computer che programma il bambino, ma il bambino che programma il computer, in questo modo si può avere un apprendimento attivo. Il matematico definisce il Costruzionismo un modello di apprendimento tipico degli ambienti informatici; se esaminiamo lo stesso nome, in esso è contenuta la parola "costruzioni", da intendere nel significato di *set* da costruzioni, considerando i linguaggi di programmazione come «set» con cui creare programmi. Questa corrente di pensiero si basa sull'assunto che i bambini debbano scoprire da soli le specifiche conoscenze di cui hanno bisogno, e il sapere di cui essi necessitano è quello che porterà ad acquisire altro sapere. Un vecchio proverbio africano richiamato spesso negli scritti di Papert, dice che se un uomo ha fame, gli si può dare un pesce, ma sarebbe ancora meglio dargli una lenza e insegnargli a pescare. In questo contesto, egli afferma che è opportuno sviluppare la *matetica*, ovvero l'arte di apprendere, che manca nel sistema scolastico tradizionale, il quale ha sempre posto l'enfasi sull'insegnamento anziché sull'apprendimento. Papert contrappone il Costruzionismo all'Istruzionismo, inteso come modalità che fa prevalere l'insegnamento e che considera la mente del bambino come qualcosa da riempire, promuovendo invece un modello il cui obiettivo è insegnare offrendo il minimo insegnamento per ottenere il massimo apprendimento. Gli aspetti su cui Papert insiste possono essere riassunti in alcuni principi, che stanno alla base del Costruzionismo [Harel, Papert, 1991]:

- Il protagonismo dello studente per uno sviluppo delle abilità cognitive e metacognitive.
- L'inversione epistemologica che sostituisce l'imparare per usare con l'usare per imparare.

- Rivalutazione del pensiero operatorio concreto su quello formale logico-deduttivo in quanto sono le esperienze concrete e casuali che danno origine alle teorizzazioni scientifiche.
- Apprendimento sintonico, apprendimento significativo che coinvolge il soggetto a livello corporeo nell'apprendimento.
- I "micromondi", cosiddette palestre cognitive, sono gli ambienti in cui si ricercano di continuo i problemi e le soluzioni attraverso il *problem finding* e il *problem solving*.
- Epistemologia "dell'indeterminatezza gestita", che si basa su un procedere in modo analogico, per aggiustamenti continui, a discapito della logica dicotomica del digitale che si basa sul vero o falso, giusto o sbagliato.
- "Apologia dell'errore" come modo di procedere, che si basa sull'apprendere riflettendo sui propri errori.

Papert, rispetto al Costruttivismo di Piaget, è convinto che la costruzione da parte del bambino sia più efficace e padroneggiata se non è solo rappresentazione mentale, ma se parallelamente è supportata da una costruzione reale, in questo modo perde la caratteristica di dottrina puramente mentalista. La programmazione, cardine del pensiero costruzionista, è quindi uno strumento che promuove un apprendimento significativo, consapevole, centrato sull'allievo.

1.4 Le influenze di Dewey e Piaget

John Dewey (1859-1952), è stato un filosofo dell'educazione e pedagogista. Per Dewey la scuola non deve essere separata dalla comunità in cui opera ed anzi, il suo compito è quello di formare membri capaci di contribuire a sviluppare e migliorare la società. Secondo lo studioso la scuola non può fornire contenuti nozionistici, astratti e separati dall'esperienza reale; l'educazione deve essere vista come una preparazione alla vita civile attraverso un coinvolgimento attivo degli studenti. Come scopo primario dell'educazione, egli suggerisce quello di ampliare lo sviluppo intellettuale attraverso attività volte alla soluzione di problemi,

all'espansione del pensiero critico e all'accrescimento delle capacità di cooperazione con gli altri. Dewey si contrappone alla tradizionale distinzione tra teoria e pratica attraverso la promozione del *learning by doing*, l'attività concreta come momento privilegiato per favorire l'arricchimento dell'individuo in tutti i suoi aspetti (fisico, psichico, intellettuale, sociale). Con Dewey, Papert condivide l'idea di un ambiente in cui si realizzi un apprendimento per imitazione, dei più grandi o comunque dei più esperti, criticando quel modello di scuola che non offre una partecipazione attiva e significativa.

L'altra importante influenza di Papert è stata la teoria costruttivista e il pensiero di Piaget. Jean Piaget (1896-1980), epistemologo, ha affrontato sistematicamente lo studio dei processi di pensiero dalla nascita all'età adulta. I suoi lavori principali sottolineano come l'attività cognitiva si sviluppi in una dimensione relativamente autonoma dai due condizionamenti di base che agiscono nella crescita, rappresentati dallo sviluppo biologico e dall'ambiente. La mente si sviluppa attraverso un equilibrio tra assimilazione (adattamento degli schemi interni alla realtà esterna) e accomodamento (ristrutturazione degli schemi interni posseduti). Alla base del suo ragionamento c'è il concetto che l'intelligenza derivi dall'azione: il soggetto costruisce attivamente le strutture della mente verso le forme dell'intelligenza logica e sperimentale. Secondo Piaget, lo sviluppo intellettuale del bambino si può dividere in fasi distinte, che definisce i quattro stadi di sviluppo cognitivo: il primo è lo stato senso-motorio (dalla nascita ai due anni) durante il quale il bambino esplora l'ambiente attraverso i sensi e il movimento e passa da avere azioni di riflesso ad azioni intenzionali, che hanno uno scopo; il secondo stadio è quello pre-operatorio (dai due ai sei anni circa), fase in cui c'è uno sviluppo del linguaggio e dell'immaginazione e prevale nel bambino l'egocentrismo intellettuale, per cui non comprende punti di vista altrui, ma esiste solo il proprio; il terzo è lo stadio delle operazioni concrete (dai sette ai dodici anni circa), il bambino è in grado di compiere operazioni mentali reversibili e sviluppa la logica, riuscendo a compiere operazioni aritmetiche di base su piano concreto; il quarto stadio è quello delle operazioni formali (dai dodici anni in poi) in cui si sviluppa il pensiero astratto ed il bambino è in grado di pensare in modo ipotetico e deduttivo anche senza aver fatto esperienza reale.

Se per tutti i pedagogisti seguaci di questa teoria è importante consolidare quest'ultimo passaggio, Papert invece vuole portare avanti la sua strategia, che consiste nel perpetuare il processo concreto il più a lungo possibile, anche fino all'età adulta e afferma: «Invece di spingere i bambini a pensare come gli adulti, faremmo meglio a ricordare che abbiamo a che fare con individui che imparano rapidamente e dovremmo fare noi ogni sforzo per assomigliare a loro» [Papert, 1993, p. 168]. Il contributo di Piaget al lavoro di Papert è stato filosofico e teoretico. Ciò che interessa al matematico non riguarda le teorie degli stadi di sviluppo, ma le idee che hanno influenzato la teoria dell'apprendimento, un Piaget come definisce lui stesso «rivoluzionario». Il Piaget epistemologo, parlando di sviluppo infantile, parla allo stesso momento dell'evoluzione della conoscenza. Egli afferma che è un errore separare ciò che viene appreso dal processo di apprendimento. Piaget si può definire come uno dei precursori del Costruttivismo. Secondo le posizioni costruttiviste, il mondo in cui viviamo è il risultato della nostra attività di costruzione di significati, questo implica che anche la conoscenza sia considerata un prodotto costruito. Questo passaggio ha spostato l'idea dell'insegnamento come trasmissione di conoscenze e saperi a un'idea invece centrata sull'apprendimento. Papert condivide gli atteggiamenti critici, tipici dei costruttivisti, nei confronti della scuola che considera il bambino come un fruitore passivo del sapere.

1.5 Pensiero computazionale e Competenza Digitale

Nello stesso anno in cui Jeannette Wing ha scritto il suo articolo riguardo al pensiero computazionale, il Parlamento Europeo pubblicava la Raccomandazione⁶ riguardante le Competenze Chiave⁷ per l'apprendimento permanente [Recommendation 2006/962/EC on key competences for lifelong learning]. Il

⁶ Raccomandazione del Parlamento Europeo e del Consiglio del 18 dicembre 2006 relativa a competenze chiave per l'apprendimento permanente.

⁷ Le otto competenze chiave sono: 1. Comunicazione nella madrelingua; 2. Comunicazione nelle lingue straniere; 3. Competenza matematica e competenze di base in scienza e tecnologia; 4. Competenza digitale; 5. Imparare a imparare; 6. Competenze sociali e civiche; 7. Spirito di iniziativa e imprenditorialità; 8. Consapevolezza ed espressione culturale.

documento, pubblicato a livello comunitario, invita gli stati membri a promuovere lo sviluppo delle otto competenze durante tutto l'arco della vita, per la realizzazione di una cittadinanza attiva. Tra queste appare anche la Competenza Digitale:

"La competenza digitale 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 è 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." (Legge 394/10, Gazzetta ufficiale dell'Unione Europea, 2006)

Il documento è stato rinnovato nel 2018 con una nuova raccomandazione, così motivata: "In risposta ai cambiamenti intervenuti nella società e nell'economia, sulla base delle discussioni sul futuro del lavoro e in seguito alla consultazione pubblica sulla revisione della raccomandazione del 2006 relativa a competenze chiave, è necessario rivedere e aggiornare sia la raccomandazione relativa a competenze chiave per l'apprendimento permanente, sia il pertinente quadro di riferimento". Nelle nuove raccomandazioni si legge:

"La competenza digitale presuppone l'interesse per le tecnologie digitali e il loro utilizzo con dimestichezza e spirito critico e responsabile per apprendere, lavorare e partecipare alla società. Essa comprende l'alfabetizzazione informatica e digitale, la comunicazione e la collaborazione, l'alfabetizzazione mediatica, la creazione di contenuti digitali (inclusa la programmazione), la sicurezza (compreso l'essere a proprio agio nel mondo digitale e possedere competenze relative alla cibersicurezza), le questioni legate alla proprietà intellettuale, la risoluzione di problemi e il pensiero critico."

La competenza digitale oggi risulta di fondamentale importanza, dato l'ingresso delle ICT in ogni ambito dell'esistenza umana, è quindi dovere da parte delle istituzioni scolastiche promuoverne lo sviluppo in senso globale. Il rapporto tra

competenza digitale e pensiero computazionale si fa sempre più importante da capire, anzi necessario per la tendenza di oggi di integrare il pensiero computazionale nell'educazione obbligatoria. È ormai chiaro che, come ha spiegato Wing, sviluppare il pensiero computazionale significa pensare e comportarsi come un informatico, non dal punto di vista dell'uso pratico delle tecnologie, ma focalizzando l'attenzione sui concetti di base che caratterizzano questo modo di pensare. Nonostante la definizione di competenza digitale presupponga l'utilizzo critico e consapevole delle tecnologie e allo stesso tempo comprenda la risoluzione dei problemi, le ricerche in questo campo evidenziano molti riferimenti alla alfabetizzazione digitale e tecnologica. Un'ampia riflessione sull'introduzione delle abilità del pensiero computazionale nell'istruzione, sottolinea invece come esso porti gli studenti oltre le capacità operative e tecniche, incoraggiando *problem-solving* e creatività (Yadav, 2014). Sviluppare il pensiero computazionale quindi significa molto di più che essere capaci di programmare un computer. Allora perché è importante il coding? La programmazione non implica solamente la scrittura di comandi, ma anche essere capaci di eseguire mentalmente ciò che si scrive. Per questo il coding può rendere concreti i concetti che stanno alla base del pensiero computazionale e può essere un utile strumento per l'apprendimento. Per riprendere le parole di Bogliolo, il coding usa tecniche e strumenti di programmazione visuale solo per sviluppare il pensiero computazionale, competenza che ha a che fare con la capacità di risolvere i problemi, di elaborare procedimenti costruttivi, di esprimere le proprie idee e la propria creatività, quindi

"Il pensiero computazionale viene prima di qualsiasi competenza digitale propriamente detta, in quanto offre gli strumenti per acquisire ulteriori competenze in modo pienamente consapevole". (Bogliolo, 2016)

1.6 Il coding come attività trasversale: Indicazioni Nazionali e Nuovi Scenari

Si può affermare chiaramente che quando si parla di coding non si parla solamente di una competenza digitale. Questo perché la competenza che consente di sviluppare è una competenza trasversale. Le attività legate al pensiero computazionale sono previste nelle *Indicazioni Nazionali per il curriculum della scuola dell'infanzia e del primo ciclo di istruzione*, emanate con il Decreto Ministeriale n. 254 del 16 Novembre 2012, documento che espone i traguardi per lo sviluppo delle competenze e gli obiettivi di apprendimento per ogni disciplina o campo di esperienza. Nell'ambito della Tecnologia è specificato: "*è precisamente attraverso la progettazione e la simulazione, tipici metodi della tecnologia, che le conoscenze teoriche e quelle pratiche si combinano e concorrono alla comprensione di sistemi complessi*" [Ministero dell'Istruzione, dell'Università e della Ricerca, 2012, p.66]. Nello stesso documento, per quanto riguarda gli obiettivi da raggiungere al termine della classe terza della scuola secondaria di primo grado si legge: "*Programmare ambienti informatici e elaborare semplici istruzioni per controllare il comportamento di un robot*". Il pensiero computazionale inteso nella complessità della sua definizione, viene inserito successivamente nel nuovo documento *Indicazioni Nazionali e Nuovi Scenari*⁸, documento a cura del Comitato Scientifico Nazionale per l'attuazione delle Indicazioni Nazionali e per il miglioramento dell'insegnamento. Nei "*Nuovi Scenari*" al punto 5 del documento si legge: "*L'esercizio della cittadinanza attiva necessita di strumenti culturali e di sicure abilità e competenze di base, cui concorrono tutte le discipline*". Tra gli strumenti culturali per la cittadinanza appare il pensiero computazionale, descritto come "*processo mentale che consente di risolvere problemi di varia natura seguendo metodi e strumenti specifici pianificando una strategia*" (MIUR, 2018). Il documento afferma che alla base di esso ci sono la lingua e la matematica e che attività legate al pensiero computazionale possono essere previste in ogni ambito del sapere. Questo perché è un processo logico e creativo allo stesso tempo, che viene messo in atto più o meno consapevolmente

⁸Nel nuovo documento del 2018, istituito con DM 254/12, il MIUR evidenzia, in risposta alla sperimentazione quinquennale delle Indicazioni Nazionali del 2012, le abilità e competenze di base che possono contribuire allo sviluppo delle competenze chiave e necessarie per l'esercizio di una cittadinanza attiva.

nella vita quotidiana, per affrontare e risolvere problemi. Le attività che possono essere messe in atto non necessitano obbligatoriamente di macchine, perché ogni situazione che implichi una costruzione di procedure e una sequenza di operazioni si colloca in questo ambito, *"a patto che le procedure e gli algoritmi siano accompagnati da riflessione, ricostruzione metacognitiva, esplicitazione e giustificazione delle scelte operate"*. Il coding rimane l'attività privilegiata che consente l'utilizzo di tali abilità, in quanto l'alunno ne può constatare la concretezza applicativa. Non si parla qui solamente di programmazione, ma di uno strumento metodologico e di una forma di arricchimento personale che non hanno a che fare con la tecnologia e riguardano invece la creatività e la capacità di espressione e di autorealizzazione; esso non contribuisce solo alla costruzione di competenze matematiche, scientifiche e tecnologiche, ma sviluppa il pensiero logico, lo spirito di iniziativa e l'affinamento di competenze linguistiche. Agli informatici competono le conoscenze specifiche e professionali della programmazione, ma tutti devono avere la possibilità di usare il coding per sviluppare il pensiero computazionale,

"in questo modo metteremo coding e programmazione nel giusto rapporto, formando nuove generazioni che, avendo sviluppato capacità di pensiero computazionale in modo naturale, potranno scegliere consapevolmente se intraprendere carriere nel campo dell'informatica o dedicarsi a qualsiasi altra disciplina con una carta in più da giocare per affrontare i problemi e realizzare le proprie idee." (Bogliolo, 2016)

1.7 L'importanza del coding a scuola

Il coding si identifica come l'attività per eccellenza per sviluppare il pensiero computazionale e le capacità di problem solving. La sua applicazione nelle classi è importante anche per le possibilità che offre di imparare in modo critico, intuitivo e creativo, grazie ai metodi e agli strumenti che vengono utilizzati. In un editoriale della rivista *Scuola Italiana Moderna* (SIM) del 2016, Rivoltella ha elencato alcune motivazioni del perché fare coding a scuola. La prima motivazione è funzionalistica, in quanto sperimentare questo tipo di attività fin dai primi anni di scolarizzazione, data la diffusione e l'impatto delle tecnologie

dell'informazione, permette ai bambini un'alfabetizzazione precoce nel linguaggio informatico, che li renderà preparati, in futuro, per le professioni che necessitano di questa competenza. La seconda motivazione è espressiva, legata principalmente allo sviluppo della creatività. La terza motivazione è legata alla funzione interpretativa, in quanto permette di sviluppare la consapevolezza, aiutando a capire che per ogni azione, apparentemente semplice, dietro ce ne sono molte e molto più complesse, che fanno sì che essa avvenga. Questo è possibile grazie alla funzionalità pratica della programmazione, che consiste nella scrittura di un codice per ottenere l'azione che vogliamo. La quarta motivazione si lega al significato emancipatorio: come ha affermato lo stesso Papert con l'introduzione della programmazione nell'ambito dell'istruzione, nel fare coding è il bambino che ha il controllo sul calcolatore, non ne diventa succube, ma è lui che decide, che analizza il problema e scrive un codice che sa modificare, smontare e rimontare; questo aiuta a rafforzare il proprio pensiero critico, a saper pensare con la propria testa. La creazione attraverso il coding è un importante contesto in cui sviluppare competenze che vadano oltre ciò che si sta imparando e come si sta imparando, concentrandosi sui processi di pensiero. Brennan e Resnick (2012), osservando bambini e ragazzi alle prese con la programmazione, hanno individuato una serie di pratiche computazionali, cioè nuove modi di lavorare, di pensare e di approcciarsi ai problemi [Marchignoli, Lodi, 2016] e delle nuove prospettive, nuovi modi di vedere il mondo e noi stessi, che sono utili non solo nelle attività di coding, ma in generale per ogni attività di progettazione:

Pratiche:

- *Essere incrementali e iterativi:* con la programmazione è quasi impossibile o comunque molto difficile realizzare un progetto perfetto al primo tentativo, questa richiede controlli successivi che ne migliorino le funzionalità e ne aggiungano nuove.
- *Testing e debugging:* la pratica di testare e correggere, procedere per tentativi ed errori è fondamentale nel coding, in quanto per sapere se un

programma sia corretto o meno è necessario provare, individuare i *bug*⁹ e cercare di risolvere i problemi.

- *Riuso e Remixing*: Utilizzare lavori già scritti di altre persone, anziché creare un programma da zero, non è solo più conveniente in termini di tempo e fatica; costruire qualcosa di nuovo sulla base di qualcosa che è già stato creato consente lo sviluppo di capacità critiche di lettura del codice, per capire cosa è opportuno utilizzare, modificare o perfezionare dei progetti altrui e per arrivare a creare cose molto più complesse di quelle che si potrebbero fare da soli.
- *Astrazione e decomposizione*: pratiche importanti nella progettazione e nel problem solving, attività che consentono di costruire qualcosa concettualizzando il problema, per tradurlo in codice nel modo più essenziale possibile, tralasciando dettagli inutili e scomponendo e lavorando su singole parti più semplici da riunire in seguito per ottenere la soluzione completa.

Prospettive:

- *Esprimere se stessi*: fare uso dei media interattivi semplicemente come consumatori, permette di imparare ad usare le tecnologie, ma non promuove lo sviluppo di tutte quelle abilità legate al pensiero computazionale; questo modo di pensare consente di creare, di esprimere se stessi e le proprie idee attraverso un nuovo mezzo, con il quale ognuno può creare qualcosa di unico e personale-
- *Connettersi*: progettare insieme, cosa molto comune e proficua in questo tipo di attività, aiuta a comprendere come collaborare con gli altri e i benefici e le opportunità che questo offre. Collaborare consente di creare con gli altri, ma anche di creare per gli altri, quindi oltre alla possibilità di creare di più di quanto si potrebbe fare da soli, offre anche la possibilità di

⁹ In informatica errore di progettazione o di programmazione di un computer. Il termine bug è entrato a far parte della terminologia informatica quando una, Grace Hopper, trovò dentro ad uno dei primi grandi computer un *baco*, un piccolo insetto tra le meccaniche, che interferiva con il programma, facendo produrre ad esso risultati inattesi.

sperimentare il valore di avere un pubblico autentico, da intrattenere, coinvolgere, arricchire e educare.

- *Farsi domande*: per programmare un computer in modo da risolvere un problema che viene dalla realtà è necessario porsi domande sul mondo che ci circonda e sul suo funzionamento, interrogarsi su ciò che diamo per scontato e, attraverso la programmazione e la progettazione, cercare di trovare delle risposte.

Uno dei rischi principali a cui è facile andare incontro quando si applica il coding a scuola, è quello di creare situazioni episodiche di apprendimento, quindi è necessario utilizzare un metodo che renda l'apprendimento significativo. L'attività laboratoriale è indispensabile e inevitabile, quando si vuole lavorare in classe facendo coding: il laboratorio prevede una didattica del fare, la risoluzione di problemi concreti. Nelle Indicazioni Nazionali è scritto che l'esperimento e la manipolazione, il gioco e tutte le esperienze pratiche sono occasioni privilegiate per apprendere ciò che successivamente dovrà essere approfondito con conoscenze teoriche. Per i teorici dell'apprendimento come Dewey e Piaget, questi sono i presupposti fondamentali dell'apprendimento significativo, che rimane tale solamente se frutto della scoperta personale di chi apprende. Scegliere di attivare un laboratorio di coding in classe diventa così un ottimo modo per mettere al centro dell'episodio di apprendimento il bambino, perché diventi protagonista e costruttore della propria conoscenza e del proprio pensiero.

1.8 Pensiero computazionale e coding nella realtà scolastica italiana

Il primo documento ufficiale che ha inserito il pensiero computazionale nella scuola italiana è la legge de "La Buona Scuola" (legge 107/2015). Il comma 7, lettera h enuncia, tra gli obiettivi formativi individuati come prioritari: "*sviluppo delle competenze digitali degli studenti, con particolare riguardo al pensiero computazionale, all'utilizzo critico e consapevole dei social network e dei media nonché alla produzione e ai legami con il mondo del lavoro*". Per sviluppare e migliorare la competenza digitale a scuola e promuovere la tecnologia come

strumento di costruzione delle competenze, la Legge 107 ha istituito il Piano Nazionale per la Scuola Digitale (PNSD), documento del Ministero dell'Istruzione, dell'Università e della Ricerca, volto a un rinnovamento della scuola italiana nell'era digitale. Il Piano nazionale era stato istituito già nel 2007, con l'obiettivo di modificare l'ambiente di apprendimento in funzione dei cambiamenti tecnologici. Le innovazioni principali sono state: Azione LIM, volta a promuovere la diffusione nelle classi della Lavagna Interattiva Multimediale come ausilio nella didattica; l'Azione Cl@ssi 2.0, con l'obiettivo di rinnovare la classe in un ambiente di apprendimento innovativo, un laboratorio tecnologico; l'Azione Scuol@ 2.0, che ha consentito a 14 istituti un cambiamento non solo nella programmazione didattica, ma anche dell'organizzazione delle risorse umane e infrastrutturali; Azione centri Scolastici Digitali (CSD), nata per soddisfare le esigenze delle scuole situate in territori particolarmente disagiati dal punto di vista geografico. Inoltre sono stati promossi altri progetti come l'Azione Wi-Fi che ha consentito la connettività wireless nelle scuole e sono stati organizzati corsi di formazione sul digitale rivolti ai docenti. Gli obiettivi del Piano Nazionale del 2015 tendono a un miglioramento complessivo del sistema scolastico attraverso strategie organizzate in quattro passaggi principali. Il primo passo è quello degli strumenti, per garantire condizioni di accesso alle scuole, qualità di spazi e ambienti adatti a praticare tutte le opportunità della società dell'informazione. Il secondo passo è quello delle competenze e dei contenuti, con lo scopo di identificare nuove traiettorie guardando al futuro. Il terzo passo riguarda la formazione del personale, indispensabile per migliorare le competenze digitali dei docenti. Infine l'ultimo passo è quello dell'accompagnamento. Per ogni passaggio sono spiegati gli obiettivi e le azioni specifiche per raggiungerli, oltre alle risorse necessarie. Il Piano Nazionale promuove lo sviluppo del pensiero computazionale in modo specifico nell'azione n. 17, "*Portare il pensiero logico-computazionale a tutta la scuola primaria*", definendo come obiettivo quello di far praticare un'esperienza di pensiero computazionale a tutti gli studenti. Viene ritenuto importante partire fin da giovanissimi nella pratica per preparare gli studenti allo sviluppo di competenze che sono al centro del nostro tempo e della vita di ogni cittadino, ma anche per anticipare la comprensione della logica che implica questo

tipo di pensiero. Il piano promuove l'iniziativa "*Programma il Futuro*" come offerta di base per tutte le scuole del territorio, ma tende ad allargare l'orizzonte a cui si rivolge la sperimentazione coinvolgendo anche le scuole dell'infanzia.

1.8.1 Programma il futuro

Il progetto Programma il Futuro è stato avviato dal CINI, il Consorzio Interuniversitario Nazionale per l'Informatica, in collaborazione con il MIUR, nell'anno scolastico 2014-2015., per portare nelle scuole italiane le attività del progetto americano *Code.org*¹⁰. Attraverso il sito dedicato¹¹, è fornito un supporto per gli insegnanti e sono resi accessibili strumenti utilizzabili in classe da tutti i docenti, senza il bisogno di una specifica conoscenza tecnica o di programmazione. Il progetto prevede due tipi di percorsi diversi: il primo, la modalità di base, consiste ne "*L'ora del Codice*", ovvero lo svolgimento di un'ora di coding da parte di tutti gli studenti; la modalità avanzata prevede, oltre alla modalità di base, dei percorsi più approfonditi per lo sviluppo del pensiero computazionale. Per partecipare è necessaria l'iscrizione dell'insegnante sul sito di fruizione¹².

1.8.2 CodeMOOC

CodeMOOC¹³ è una campagna di alfabetizzazione online avviata nel 2016 da Alessandro Bogliolo. Si tratta di un corso aperto e gratuito rivolto agli insegnanti, che aiuta a introdurre il pensiero computazionale in classe con strumenti immediati e intuitivi. Il corso consiste in video-lezioni in diretta, in cui i partecipanti sono coinvolti attivamente attraverso sondaggi il cui esito condiziona

¹⁰ Per una descrizione dell'iniziativa vedere sottoparagrafo 1.9.2.

¹¹ Consultabile al link: <https://programmailfuturo.it>.

¹² Il sito su cui si appoggia il progetto di Programma il futuro è raggiungibile con il seguente link: <https://studio.code.org/courses>.

¹³ MOOC è l'acronimo di *Massive Open Online Course* (corsi online aperti alla fruizione di massa).

e può indirizzare la lezione verso altre strade. La caratteristica del MOOC è quella di offrire attività asincrone che ciascuno può seguire quando vuole, ma nonostante il corso dia disponibile *on demand* e senza scadenze, cerca di mantenere un contatto dal vivo, sia online che in presenza.

1.9 Le proposte internazionali

Nel 2018, la Commissione Europea ha adottato il Digital Education Action Plan, il Piano d'Azione per l'Istruzione Digitale, per supportare l'uso delle tecnologie e lo sviluppo delle competenze digitali nell'educazione. Il piano d'azione stabilisce tre priorità, suddivise in 11 azioni concrete, predisponendo delle misure per aiutare gli stati membri ad accogliere i cambiamenti e le opportunità educative dell'era digitale. Le priorità sono:

1. Migliorare l'utilizzo della tecnologia digitale per l'insegnamento e l'apprendimento: l'Europa si pone come obiettivi quelli di far fronte al divario di connettività tra gli stati membri, sostenendo le zone svantaggiate e rurali per una diffusione della banda larga in tutte le scuole europee, sostenere la preparazione digitale delle scuole e fornire un quadro di riferimento per le qualifiche.
2. Sviluppare le competenze e le capacità digitali pertinenti ai fini della trasformazione digitale: le azioni prefissate sono la creazione di una piattaforma europea per l'istruzione digitale superiore, rafforzare la scienza aperta e dei cittadini tramite formazione, introdurre classi di programmazione in tutte le scuole, promuovere campagne di sensibilizzazione a livello comunitario e informazione sulla cyber-sicurezza, ridurre il divario di genere favorendo le competenze digitali tra le ragazze.
3. Migliorare l'istruzione mediante un'analisi dei dati e una previsione migliori: per questo aspetto l'UE si impegna a raccogliere dati concreti relativi all'utilizzo delle ICT e delle competenze digitali tramite studi di riferimento che valutino i progressi conseguiti; inoltre avvia, a partire dal

2018, progetti relativi all'intelligenza artificiale e all'analitica dell'apprendimento a scopo di monitoraggio e risoluzione di problemi; infine si impegna a avviare una previsione strategica sulle tendenze fondamentali derivanti dalla trasformazione digitale per il futuro dell'istruzione.

Nell'azione numero sei la commissione incoraggia tutte le scuole ad aderire all'iniziativa *CodeWeek* (Settimana Europea della Programmazione), per dare la possibilità agli studenti di esplorare la programmazione. L'obiettivo dell'Unione Europea è quello di raggiungere il 50% di tutte le scuole dell'Unione entro il 2020. Dal 2015 è stata promossa da SAP¹⁴ una iniziativa di sviluppo delle competenze digitali in Africa, l'*Africa Code Week*, per promuovere lo sviluppo sostenibile del continente. Il progetto di alfabetizzazione digitale e di coding è rivolto ai giovani e finora ha coinvolto circa 4 milioni di Africani. Un progetto esteso a livello mondiale è invece *Code.org*, sostenuto dal governo americano e finanziato da aziende informatiche di tutto il mondo, con lo scopo di diffondere coding e pensiero computazionale garantendo l'accesso anche alle categorie più svantaggiate.

1.9.1 CodeWeek

La Settimana Europea della Programmazione è un'iniziativa che ha lo scopo di portare la programmazione e l'alfabetizzazione digitale a tutti in modo divertente. CodeWeek promuove l'organizzazione di eventi, accessibili e replicabili, che offrano la possibilità di sperimentare il coding. Consiste nell'offrire e condividere esperienze che possono essere organizzate da chiunque, senza specifiche regole da seguire, purché la proposta sia condivisa, quindi per raggiungere più persone possibile; per questo si può definire una campagna di alfabetizzazione. La gestione è coordinata da volontari, chiamati "ambasciatori", ma la partecipazione è libera e sono invitate soprattutto le scuole a prenderne parte. Per condividere una

¹⁴ SAP è un'azienda mondiale di software gestionale nata in Germania nel 1972. Per informazioni consultare il sito: <https://www.sap.com/index.html>.

lezione, un *open day* o un evento organizzato basta registrare la data e l'attività nella mappa, esplorabile sul sito¹⁵, sulla quale sono visibili altre attività, in Europa e non solo. Il progetto offre anche corsi di formazione online per chi non ha mai avuto esperienze nel campo della programmazione. CodeWeek richiede che siano rispettate alcune linee guida per prendere parte all'iniziativa, come l'utilizzo di materiali open-source disponibili gratuitamente, la pubblicizzazione dell'evento e soprattutto la condivisione sulla mappa.

1.9.2 Code.org

Organizzazione senza scopo di lucro, Code.org nasce negli Stati Uniti, fondata da Hadi Partovi, con l'idea di collezionare i corsi di programmazione delle scuole americane. Il portale offre oggi attività interattive strutturate in corsi gradualmente adatte per ogni fascia di età, con obiettivi definiti e fornendo supporto per gli insegnanti. La registrazione è gratuita e dà la possibilità di salvare i progressi di ogni utente; le attività di coding utilizzano linguaggi testuali e visuali, ma alcune non richiedono l'uso del calcolatore e possono essere svolte negli spazi scolastici o all'aperto; alcune lezioni sono libere, per consentire agli utenti di sperimentare con i concetti imparati. Code.org promuove l'evento "*Hour of Code*": attività online che consiste nel far sì che ogni studente, durante la seconda settimana di dicembre, svolga almeno un'ora di programmazione. Proposta per la prima volta nel 2013, l'attività consisteva in un gioco a 20 livelli in cui un personaggio deve superare un labirinto (una scacchiera quadrata) per raggiungere un obiettivo. Il personaggio si muove con l'esecuzione di un programma che il giocatore deve costruire in un editor, usando dei blocchi messi a disposizione. Dal 2013 ad oggi sono state sviluppate molte versioni diverse, sia online che *unplugged*¹⁶. Code.org è l'iniziativa alla quale si ispira il progetto italiano Programma il Futuro.

¹⁵ Pagina web al link: <https://codeweek.eu/>.

¹⁶ Sono giochi motori o da fare con carta e penna o altri materiali, senza l'uso di un calcolatore o dispositivi elettronici.

1.9.3 CoderDojo, una esperienza extra-scolastica

Movimento internazionale volontario e no-profit, CoderDojo nasce nel 2011 a Cork, in Irlanda, quando il diciottenne James Whelton fonda un club per alcuni ragazzi suoi compagni, ai quali aveva iniziato ad insegnare a programmare. Questo catturò l'attenzione di Bill Liao, un benefattore che, interessato al progetto, ne finanziò la crescita, contribuendo alla fondazione del movimento di club gratuiti che in poco tempo si diffuse nel mondo. Il termine Coder significa programmatore, mentre Dojo è il luogo in cui si praticano le arti marziali, quindi il termine significa palestra per programmatori. Ogni club è indipendente e può essere creato da chiunque, ma ci sono dei principi da rispettare tra cui la tutela dei ragazzi, la gratuità dei laboratori, la condivisione delle risorse; il motto del movimento è: *"Above All, Be Cool: bullying, lying, wasting people's time and so on is uncool"* (Prima di tutto sii in gamba: fare il bullo, mentire e far perdere tempo agli altri non è da persone in gamba). Per creare un club non è necessario disporre di molte risorse, basta uno spazio fisico che possa ospitare le attività e dei volontari che si dedichino nel seguire i bambini, non come insegnanti, ma come facilitatori, perché essi possano sperimentare e scoprire da soli. In Italia sono attivi circa 150 CoderDojo. Gli eventi, della durata in genere di tre ore, sono divisi in tre parti: un tutorial, la merenda, una parte di tempo per la sperimentazione libera dei ragazzi.

1.10 Strumenti per il coding

Le proposte di programmazione per bambini e ragazzi in tempi recenti hanno avuto una ampia diffusione, con proposte diverse e diversi approcci. Un approccio è quello del *problem solving*, che propone al bambino problemi di difficoltà crescente, da risolvere scrivendo un programma. Per questo vengono utilizzati software specifici adatti all'età dello studente. L'altro approccio mette al centro la creatività lasciando completa libertà ai bambini, che possono scegliere il progetto da realizzare utilizzando i linguaggi di programmazione. Sono disponibili inoltre schede elettroniche e robot didattici programmabili, con i quali invece

dell'animazione sullo schermo si può osservare il movimento nel mondo fisico; anche se i concetti e le pratiche sono le stesse, il vantaggio è l'oggetto fisico tangibile, quindi la possibilità di toccare e vedere concretamente gli effetti del codice. Di seguito sono descritti alcuni dei principali e più usati strumenti di programmazione.

- **Scratch:** Progetto nato nel 2006 del Lifelong Kindergarten Group, al Media Lab del MIT, a Boston. L'ideatore è Mitch Resnick, un ex studente di Seymour Papert. Si tratta di un software di programmazione usufruibile in modo gratuito online, con il quale è possibile creare storie interattive, giochi, animazioni. Scratch utilizza un linguaggio visuale a blocchi, è molto intuitivo e facile da utilizzare. Progettato specificamente per i ragazzi dagli otto ai sedici anni, è utilizzato invece da persone di tutte le età, è usato in più di 150 paesi e tradotto in 40 lingue diverse. Sul sito è possibile creare progetti, salvarli e condividerli, facendo così in modo che altri utenti possano vedere il codice con cui sono stati creati, per modificarli o commentarli. È possibile inoltre registrare un account da educatore per aprire classi virtuali a cui possono accedere più utenti, questo rende Scratch molto adatto per essere utilizzato a scuola. Per i più piccoli, è disponibile una versione di ScratchJR, semplificata per chi sta appena imparando a leggere.
- **LOGO:** Software di programmazione di tipo testuale ideato da Seymour Papert negli anni '60 al MIT, con lo scopo di rendere la pratica della programmazione facile e accessibile a tutti, anche ai bambini. LOGO nasce come un grande robot a forma di tartaruga che, comandato tramite il computer, poteva disegnare grazie ad un pennarello. Successivamente si sono sviluppate le versioni su schermo, dove il cursore aveva la forma di un piccolo triangolo. Oggi è disponibile una implementazione di LOGO gratuita sul free software LibreOffice, chiamata LibreLogo, con la quale basta scrivere il codice su una pagina del word processor Writer e premere il pulsante play sul menù per vedere muoversi e disegnare una piccola tartaruga.

- **Bee-Bot:** Si tratta di un robot a forma di ape programmabile per eseguire percorsi. Possiede dei tasti sul dorso che consentono la programmazione e memorizza fino a 40 comandi. Può spostarsi in avanti, indietro e ruotare di 90°. Ogni passo misura 15 cm. Il robot è adatto per i bambini della scuola dell'infanzia e dei primi anni di scuola primaria.



Figura 1 - Bee Bot¹⁷

- **Blue-Bot:** È la versione di Bee-Bot che, oltre ai tasti sul dorso, è fornita di tecnologia Bluetooth ed è programmabile anche con smartphone o tablet, tramite l'applicazione dedicata scaricabile per iOS e Android. A differenza dell'altro robot, questo ha un guscio trasparente che consente di vedere come è fatto all'interno.



Figura 2 - Blue Bot¹⁸

¹⁷ Immagine consultabile all'indirizzo: <https://www.campustore.it/robotica-educativa-elettronica-coding/bee-bot-blue-bot.html> (consultato in data 03/08/2019).

- **LEGO Mindstorms:** È un set di costruzioni robotiche programmabili, che permette di creare e comandare i robot tramite un software rilasciato dalla ditta. Il kit consente di costruire ben 17 robot diversi per struttura e per livello di abilità. Attualmente la piattaforma LEGO Mindstorms ha rilasciato il set EV3, dove EV sta per evoluzione. Il mattoncino EV3 è intelligente e programmabile, come la mente del robot controlla i motori e i sensori e lo fa muovere e parlare. Le comunicazioni sono trasmesse in modalità wireless, tramite Wi-Fi e Bluetooth.

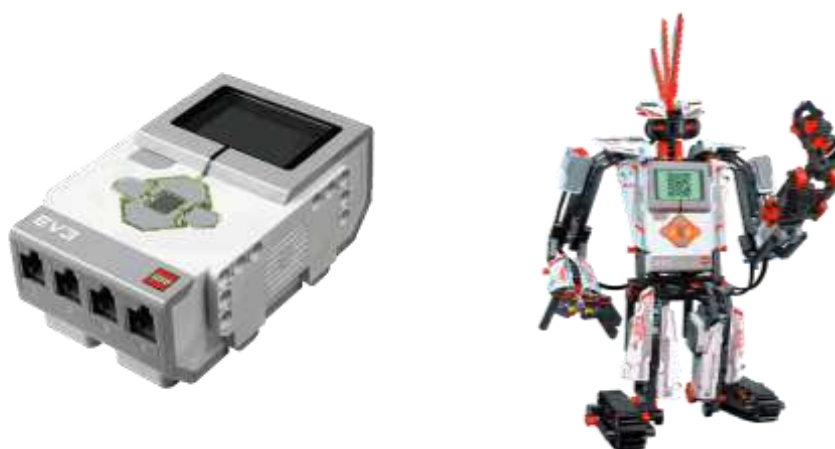


Figura 3 - Mattoncino intelligente di LEGO Mindstorms EV3 e esempio di robot costruito con il kit¹⁹

- **Cubetto:** Robot di legno che si muove su ruote e controllabile via Bluetooth tramite un'apposita tavoletta forata in cui è possibile inserire dei tasselli di 4 colori diversi. Ad ogni colore corrisponde un comportamento, avanti, destra, sinistra e posizionandoli uno dopo l'altro si può programmare il robot per fargli svolgere dei percorsi. Il kit comprende anche una mappa in tessuto sulla quale è possibile far muovere Cubetto. Il vantaggio di questo strumento è che dà la possibilità di sperimentare il coding anche a bambini ipovedenti grazie alla componente tattile, oltre ai colori, infatti, i tasselli sono distinguibili anche per la forma e per i segni incisi sui lati.

¹⁸ Immagine consultabile all'indirizzo : <https://www.campustore.it/robotica-educativa-elettronica-coding/bee-bot-blue-bot.html> (consultato in data 03/08/2019).

¹⁹ Immagine consultabile all'indirizzo: <https://www.lego.com/it-it/mindstorms/build-a-robot/ev3rstorm> (consultato in data 03/08/2019).



Figura 4 - Kit completo Cubetto²⁰

- **CodyRoby**²¹: Versione di coding unplugged, è un gioco di carte fai da te in cui i giocatori hanno a disposizione carte comando di tre tipi, con dei simboli che indicano l'azione da svolgere (avanti, gira a destra, gira a sinistra) per muovere delle pedine su una scacchiera. Il nome è formato dalle due componenti del gioco: Cody è il programmatore mentre Roby è il "robot" che esegue le istruzioni.

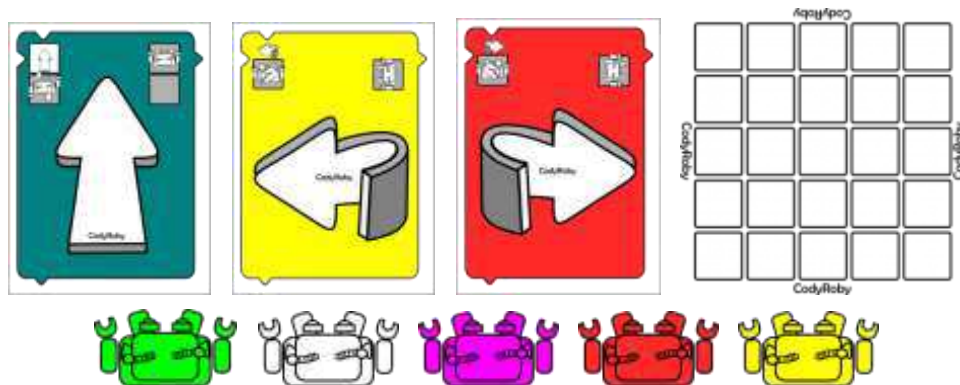


Figura 5 - Kit CodyRoby: comprende le carte azione, la scacchiera e le pedine²²

²⁰ Immagine consultabile all'indirizzo: <https://www.campustore.it/robotica-educativa-elettronica-coding/cubetto.html> (consultato in data 03/08/2019).

²¹ Il kit CodyRoby è scaricabile al seguente link: <http://codemooc.org/codyroby/> dove sono presenti anche dei video tutorial per il montaggio.

²² Immagini consultabili all'indirizzo: <http://codeweek.it/cody-roby/> (consultato in data 03/08/2019)

- **CodyWay**: Altra versione di coding unplugged, offre istruzioni visuali da comporre per formare dei codici che rappresentano le azioni da eseguire per raggiungere un luogo o una stanza. I programmi possono essere stampati per essere usati come cartelli segnaletici (per esempio a scuola per spiegare la strada dell'uscita antincendio) o come guide.



Figura 6 - Blocchi ritagliabili di CodyWay²³

²³ Immagini tratte dalla presentazione sul sito di CodeWeek, consultabili e scaricabili all'indirizzo: <https://docs.google.com/presentation/d/1wgibNptuspaiLjwNtpszQaBOPM5TMeUUYNJfQOp-wjA/edit#slide=id.p4> (consultato in data 03/08/2019).

Capitolo 2

Linguaggi di programmazione

2.1 Linguaggio di programmazione: una definizione

Fare coding, nel senso di programmare, implica la scrittura di un programma, di una sequenza di azioni che devono essere eseguite per raggiungere un determinato obiettivo. Nell'informatica, il programmatore comunica alla macchina cosa deve fare, attraverso un codice che questa è in grado di comprendere. Questo tipo di comunicazione può avvenire grazie al microprocessore²⁴, un componente in grado di leggere, interpretare ed eseguire istruzioni [Bogliolo, 2016]. Ma che tipo di linguaggio viene usato per programmare? "Occorre imparare a dialogare con la macchina attraverso un linguaggio condiviso sia dall'uomo che dalla macchina" [Capponi, 2009, pp. 87-88]. In informatica, il linguaggio di programmazione è un insieme di parole e di regole che consentono di programmare un elaboratore affinché esegua compiti predeterminati [dall'enciclopedia online Treccani²⁵]. È importante specificare che questo linguaggio è "formale²⁶", governato quindi da regole precise che definiscono la forma delle frasi (la sintassi) e il loro significato (la semantica). Questo sistema formale è tipico dei linguaggi con i quali si possono scrivere gli algoritmi, per questo non consentono ambiguità, tipiche invece del linguaggio umano. La sintassi del linguaggio indica come le istruzioni devono essere scritte. La correttezza sintattica è necessaria perché un'azione sia eseguita, in quanto i comandi scritti in modo non corretto non vengono riconosciuti. La semantica indica qual è l'istruzione che la macchina deve svolgere, quindi un errore di questo

²⁴ Concepito da Turing e von Neumann nel secolo scorso. Oggi i microprocessori sono talmente piccoli da poter essere nascosti in qualsiasi tipo di apparecchio, per questo gli oggetti programmabili aumentano ogni giorno.

²⁵ <http://www.treccani.it/enciclopedia/linguaggio-di-programmazione/>.

²⁶ Il linguaggio formale si contrappone al linguaggio naturale, il linguaggio solitamente usato nella comunicazione tra individui di un gruppo sociale che lo condivide.

tipo viene fuori solo nell'esecuzione delle istruzioni, se la soluzione ottenuta non risolve il problema posto in partenza. I linguaggi di programmazione possono essere classificati come linguaggi di alto livello o di basso livello, a seconda del grado di astrazione dal linguaggio macchina, cioè il linguaggio del calcolatore, che viene eseguito direttamente da esso. Quelli a basso livello, coincidono con lo stesso linguaggio macchina, o ne differiscono per poco, mentre quelli ad alto livello sono scritti in un linguaggio più comprensibile da chi programma, necessitano quindi di un compilatore che li traduca in un linguaggio macchina per essere eseguite. Questi ultimi si possono suddividere a loro volta in linguaggi interpreti e linguaggi compilatori: l'interprete traduce un'istruzione alla volta, ne verifica la correttezza sintattica e la esegue, questo rende il processo più lento e a rischio di blocco, se nella stringa di comandi si presenta un errore, che però è immediatamente individuabile; il compilatore esegue la traduzione e il controllo delle istruzioni una sola volta per tutto il programma, il codice che ne deriva quindi è sintatticamente corretto, direttamente eseguibile, ma non sempre ha una logica nelle operazioni da compiere [Capponi, 2009].

La storia dei linguaggi di programmazione inizia poco prima di mezzo secolo fa, tempo durante il quale sono state sviluppate molte tipologie di linguaggi diversi, anche e soprattutto a causa dell'enorme espansione dei computer e del loro utilizzo, che è passato dalle mani di esperti a quelle di tutti. La programmazione, prima degli anni '50, consisteva nello scrivere programmi direttamente nel linguaggio macchina, in cui le istruzioni erano rappresentate da codici numerici espressi nel sistema binario²⁷. Successivamente, data la difficoltà di questo tipo di linguaggio, si sono sviluppati i linguaggi assemblativi, che permettevano l'utilizzo di parole al posto dei numeri. Uno dei primi linguaggi ad esercitare una forte influenza fu il *Fortran*, sviluppato tra il 1954 e il 1957 per eseguire calcoli scientifici e numerici. La diffusione dei computer su larga scala, richiese lo sviluppo di un linguaggio semplice, che potesse avvicinare alla programmazione anche persone non esperte nel campo dell'informatica. Venne così diffuso su tutti i

²⁷ Sistema di numerazione posizionale in base 2, utilizzato per rappresentare i numeri in un elaboratore grazie alla sua semplicità di scrittura; usa solamente due simboli, 0 e 1 [dall'enciclopedia online Treccani].

personal computer il *Basic*²⁸, dal quale poi si sono evoluti molti linguaggi diversi. Il Basic includeva un set di istruzioni relativamente ristretto, inoltre era stato progettato rimuovendo tutta la sintassi non necessaria, per supportare tempi brevi tra la composizione e l'esecuzione e tutto questo rendeva il linguaggio più accessibile ai principianti [Weintrop, Wilensky, 2015]. I linguaggi fino ad oggi sviluppati sono diverse migliaia, infatti è diventato piuttosto semplice inventarne uno; le differenze sono sostanziali, in quanto ogni programma deve far fronte a specifici problemi e deve gestire determinate tipologie di dati. Per questo è praticamente impossibile imporre un linguaggio universale [Gian Marco Todesco, 2008, Enciclopedia della Scienza e della Tecnica²⁹]. La maggior parte dei linguaggi utilizzati oggi sono di alto livello e si possono valutare in base ad alcuni criteri (evidenziati e descritti da Todesco): l'espressività è la capacità di esprimere in modo conciso operazioni complicate; la semplicità condiziona la possibilità di apprendere e padroneggiare il linguaggio; la leggibilità influisce sulla capacità di comprendere il funzionamento e la finalità del programma e ne facilita la correzione; l'efficienza misura la qualità del prodotto in termini di velocità e dimensioni; l'affidabilità è data dall'assenza di costrutti che possono portare un alto rischio di commettere errori e dal controllo in esecuzione; la portabilità indica la possibilità di adattamento del programma a diverse piattaforme; la popolarità corrisponde alla diffusione del linguaggio, quindi condiziona la disponibilità di manuali, corsi di formazione, numero di programmatori esperti nel dato linguaggio. I linguaggi che saranno analizzati in questo capitolo sono linguaggi semplici, quindi fondamentali per l'utilizzo a scopo didattico. In una pubblicazione del 2012, gli autori Brennan e Resnick hanno individuato, tra le possibilità di sviluppo del pensiero computazionale attraverso Scratch, sette concetti, comuni a molti linguaggi di programmazione:

- Sequenze: un concetto chiave nella programmazione è quello che ogni attività è espressa come una serie di singoli passi o istruzioni eseguibili dal

²⁸ Sviluppato nel 1964 da John C. Kemeny e Thomas E. Kurtz, acronimo di *Beginner's All-purpose Symbolic Instruction Code*, in italiano "codice di istruzioni simbolico universale per principianti".

²⁹ Versione online al link: [http://www.treccani.it/enciclopedia/linguaggi-di-programmazione_\(Enciclopedia-della-Scienza-e-della-Tecnica\)/](http://www.treccani.it/enciclopedia/linguaggi-di-programmazione_(Enciclopedia-della-Scienza-e-della-Tecnica)/)

computer e ogni sequenza corrisponde ad una azione o ad un comportamento che deve essere prodotto; l'ordine delle istruzioni determina il risultato finale.

- Cicli (Loop): il meccanismo del *loop* consente l'esecuzione multipla di una sequenza, grazie all'utilizzo della ripetizione che consente alle istruzioni di essere eseguite più volte utilizzando il minor numero di comandi.
- Eventi: componenti essenziali nell'utilizzo di media interattivi, gli eventi sono situazioni che si verificano solamente se viene eseguita una determinata azione.
- Parallelismo: questa caratteristica permette l'esecuzione contemporanea di sequenze diverse. Questo può avvenire per istruzioni date ad oggetti diversi, ma anche sullo stesso oggetto.
- Condizioni: altro concetto chiave dei media interattivi è il verificarsi di determinate situazioni solamente in base certe condizioni, che supportano l'espressione di esiti diversi.
- Operatori: consentono al programmatore la manipolazione numerica e delle stringhe, grazie al supporto per espressioni matematiche e logiche.
- Dati: concetto che consente l'archiviazione, il recupero e l'aggiornamento di valori, in quanto una soluzione algoritmica ad un problema richiede la memorizzazione, l'analisi e la rappresentazione di informazioni.

I principali linguaggi utilizzati oggi sono linguaggi di alto livello e orientati agli oggetti (OOP, Object-Oriented Programming), cioè permettono la programmazione di oggetti che interagiscono gli uni con gli altri. Tra i più comuni abbiamo il linguaggio C++, Java, Python, Visual Basic. I linguaggi che verranno affrontati in questo capitolo e rientrano in questa categoria sono il linguaggio Logo e il linguaggio Scratch; le caratteristiche proprie dei due linguaggi rendono l'utilizzo di questo tipo di software molto adatto per scopi educativi.

2.2 LOGO: un linguaggio di programmazione testuale

Il linguaggio LOGO è stato creato nel Laboratorio di Intelligenza Artificiale del MIT da Seymour Papert e i suoi collaboratori, nel 1967. LOGO permetteva di creare grafica manovrando una tartaruga tramite dei comandi; nella sua prima versione, la tartaruga era un vero e proprio robot che disegnava con un pennarello mentre si muoveva e i comandi le venivano trasmessi tramite il computer. Negli anni '80 LOGO divenne un software e negli anni '90 il programma già circolava in una versione installabile da *floppy disk*: una volta lanciato, sullo schermo nero si potevano scrivere le istruzioni in sequenza, una dietro l'altra e, con il comando per eseguire il codice, la tartaruga (diventata un cursore) iniziava a muoversi, tracciando un disegno sullo schermo [Formiconi, 2018].

LOGO è un linguaggio di programmazione testuale: questo significa che il software necessita (o consente) della scrittura manuale, da parte del programmatore, di stringhe di testo, come istruzioni da impartire alla tartaruga. La sua sintassi è molto semplice e intuitiva: i comandi sono espressi in lingua inglese; quelli principali sono FORWARD e BACK, rispettivamente "vai avanti" e "torna indietro" e LEFT e RIGHT, "ruota a sinistra" e "ruota a destra"; tutti devono essere seguiti da un numero che rappresenta nei primi due casi il numero di passi da eseguire, nel caso delle rotazioni i gradi, ovvero l'ampiezza dell'angolo da formare.

Lo scopo di LOGO è espressamente e puramente quello di essere usato per fini didattici: secondo il suo creatore, tutti devono poter sperimentare la programmazione, soprattutto i bambini. Inoltre, attraverso l'utilizzo del computer, questa serve specificamente per facilitare l'apprendimento della matematica. L'ambiente di lavoro di LOGO è definito da Papert come *Geometria della Tartaruga*. La tartaruga è una entità geometrica, caratterizzata dalla posizione nel piano e dall'orientamento [Capponi, 2009]. Il bambino può identificarsi in essa, sperimentando in prima persona i movimenti che può fare; utilizza così il proprio corpo come punto di partenza per raggiungere la geometria formale [Papert, 1984]. La Geometria della tartaruga si basa su tre principi, descritti da Papert in *Mindstorms*: il *principio di Continuità*, secondo il quale è possibile integrare i nuovi concetti appresi con le conoscenze precedenti; il *principio di Potenza*,

secondo il quale l'ambiente di apprendimento deve consentire la creazione di progetti significativi, che non sarebbero mai stati creati; il *principio di Risonanza Culturale*, secondo cui ciò che viene appreso deve avere un senso anche dal punto di vista sociale. Il linguaggio LOGO è un linguaggio di programmazione interprete. Alcune sue caratteristiche sono definite da Andreoli (1996):

- È procedurale: permette di suddividere un problema in parti più semplici, in questo modo è possibile gestire idee complesse, tramite una struttura modulare.
- È interattivo: l'esecuzione dei comandi è immediata e anche il riscontro del loro funzionamento.
- È ricorsivo: le procedure possono richiamare se stesse.
- È espandibile: il ridotto vocabolario iniziale è espandibile con l'invenzione di parole nuove, grazie alle procedure.
- È facile: per usare LOGO non sono richieste conoscenze di informatica né di programmazione.
- È potente: consente la scrittura di procedure anche molto complesse.

Grazie a queste caratteristiche, LOGO consente di esercitare il processo di analisi e correzione dei problemi suddividendoli in piccole parti e di esercitare la sintesi nel costruire procedure complesse, combinando sottoprocedure relativamente elementari [Reggini, 1982]. Tutto questo permette di allenare la logica, affinando le capacità di ragionamento e di risoluzione dei problemi.

2.2.1 LibreLogo, programmazione con software libero

Una versione di LOGO accessibile in modo semplice e gratuito è quella che si trova nel word processor Writer di LibreOffice. LibreOffice è un software libero, quindi è ideale per essere utilizzato in qualsiasi contesto, anche formativo. Il concetto di Free Software è stato creato nel 1983 da Richard Matthew Stallman, fisico e programmatore presso il Laboratorio di Intelligenza Artificiale del MIT. Il pensiero di Stallman era che il software, in generale, dovesse essere liberamente

condivisibile tra tutti gli esseri umani. Dopo aver lasciato la sua posizione come ricercatore ha fondato la Free Software Foundation, con lo scopo di promuovere il software libero in tutto il mondo, tra tutti gli utenti di computer; inoltre ha creato il Progetto GNU, un sistema operativo libero e la GNU General Public License e il concetto di *copyleft*, base giuridica per mezzo della quale gli autori possono liberamente rinunciare a alcuni diritti d'autore. La Free Software Foundation specifica che "*Per 'software libero' si intende un software che rispetta la libertà e la comunità degli utenti. In linea di massima, significa che gli utenti hanno la libertà di eseguire, copiare, distribuire, studiare, cambiare e migliorare il software*³⁰". Il messaggio di natura etica che comporta l'utilizzo di questo tipo di programmi è riassumibile in quattro libertà:

1. Libertà di eseguire il programma come si desidera, per qualsiasi scopo (libertà 0).
2. Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità (libertà 1). L'accesso al codice sorgente ne è un prerequisito.
3. Libertà di ridistribuire copie in modo da aiutare gli altri (libertà 2).
4. Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti apportati (e le versioni modificate), in modo tale che tutta la comunità ne tragga beneficio (libertà 3). L'accesso al codice sorgente ne è un prerequisito.

LibreOffice include tutte le applicazioni tipiche delle suite per ufficio, per scrivere, creare presentazioni, organizzare i dati ed altre funzioni. Disponibile dal 2014 su LibreOffice Writer è il *plugin* LibreLogo, una versione di LOGO abbastanza completa alla quale si accede semplicemente aggiungendo la barra degli strumenti omonima. Questa idea è stata di Németh László, che ha riprodotto le funzionalità del programma originale scrivendole in linguaggio Python. Per vederlo in esecuzione basta scrivere il codice sulla pagina e premere il pulsante *play*: se il codice è corretto apparirà nel documento una tartaruga che si muoverà in base ai comandi dati. [Formiconi, 2018].

³⁰ <https://www.gnu.org/philosophy/free-sw.en.html>

2.3 Scratch e il linguaggio di programmazione visuale

Gli sforzi per sviluppare linguaggi più accessibili sono stati incentivati dopo che alcuni studi hanno riconosciuto che la progettazione stessa di un linguaggio di programmazione può supportare o ostacolare gli studenti nel loro tentativo di programmare. Stefik e Siebert (2013) hanno riscontrato che le caratteristiche della sintassi influenzano direttamente l'apprendimento del linguaggio di programmazione; una soluzione a questo problema è stata la creazione di strumenti di programmazione visiva, che rappresentano quindi visivamente le informazioni sui comandi, facilitandone la composizione senza il rischio di commettere errori sintattici. L'approccio basato sui blocchi si è diffuso molto in tempi recenti, questo tipo di programmazione fornisce segnali visivi all'utente su come e dove possono essere utilizzati i comandi. Programmare in un ambiente del genere consiste nel trascinare i blocchi in un'area di composizione e nel montarli, farli "scattare" insieme come se fossero dei pezzi di un puzzle, per costruire un testo. Se due blocchi non possono essere uniti per problemi di sintassi, non è possibile neanche incastrarli tra loro, in questo modo si eliminano le difficoltà sintattiche che possono emergere costruendo un codice, mantenendo comunque la pratica di assemblaggio del programma, istruzione dopo istruzione [Weintrop, Wilensky, 2015].

Scratch è un progetto nato da un gruppo di ricerca del MIT Media Lab di Boston, il Lifelong Kindergarten Group. L'ideatore, Mitch Resnick, è stato uno studente di Papert. Scratch è un ambiente di programmazione visuale online (ma è disponibile anche un editor Scratch per Desktop³¹) totalmente gratuito, semplice e intuitivo, utilizzabile anche dai bambini. Progettato in modo specifico per la fascia d'età che va dagli otto anni ai sedici, è effettivamente utilizzato da persone di tutte le età. Scratch consente di creare programmi per dare vita a storie interattive, animazioni e giochi. La sua struttura a blocchi semplifica molto il lavoro di programmazione: essi sono divisi per funzioni, ognuna delle quali è caratterizzata da un colore. I mattoncini sono trascrinabili e impilabili uno sopra l'altro, non c'è bisogno di scrivere nessun tipo di comando, a parte i valori che indicano come o

³¹ Scaricabile dal sito al link: <https://scratch.mit.edu/download>.

di quanto spostarsi, ruotare, o indicazioni di altro tipo. La grafica di Scratch è molto accattivante, il programmatore può giocare con gli Sprite, ovvero gli oggetti programmabili, scambiandoli o modificandone l'aspetto, può creare sfondi personalizzati, aggiungere suoni, musica e tanto altro. Scratch non è solamente un ambiente di programmazione, ma anche di condivisione: tutti gli utenti, dopo essersi registrati, possono salvare e condividere le proprie creazioni, e tutti i progetti condivisi sono consultabili; in questo modo è possibile commentare i lavori degli altri, modificarli e dare consigli. La *community* è sicura e controllata, adatta anche per i bambini, in quanto le informazioni degli account non sono rese disponibili a nessuno³². Per gli educatori, è possibile creare delle classi virtuali alle quali accedere tramite un apposito link creato dal programma, questo permette di visionare le attività e i lavori di chi ne fa parte, in un ambiente di condivisione a totale disposizione dei bambini.

2.4 La metodologia progettuale del bricolage e i micromondi

Due ragioni fondamentali sono state definite per affermare che imparare a programmare può essere un esercizio intellettuale rilevante per gli allievi: la prima ragione è data dal fatto che viene dato loro pieno controllo della macchina e potere su di essa, la seconda è relativa a un apprendimento secondario, che avviene durante la programmazione [Noss, 1988]. Essa infatti è una attività progettuale molto importante a livello di apprendimento, in quanto il programmatore si deve cimentare in processi mentali quali fare ipotesi, risolvere problemi, generalizzare. Chi programma procede per tentativi. Papert usa il termine *bricolage* per definire l'atteggiamento e il modo di lavorare del programmatore. Esso è il tipico metodo di lavoro dei *Bricoleurs*, che si contrappone al metodo dei *Planners*, cioè chi pianifica passo dopo passo il lavoro che deve svolgere [Papert, 1996]. I principi fondamentali di questa metodologia che l'autore ha definito sono: usare le cose di cui si dispone, improvvisare, adattarsi. Il modo di apprendere del Bricoleur è molto simile a quello del

³² Per maggiori informazioni, consultare il sito di Scratch al seguente link: <https://scratch.mit.edu/about>.

bambino, in quanto consiste nell'avvicinarsi gradualmente alla meta. Il concetto del bricolage come metodo progettuale era stato già introdotto da Lévi-Strauss, ne *"Il pensiero selvaggio"*; per l'autore il bricolage riguarda essenzialmente tre aspetti dell'agire umano: l'uso di materiali di seconda mano per costruire qualcosa; l'uso di strumenti che dovrebbero servire per qualcos'altro, ma che sono funzionali per quello che si sta facendo; la combinazione e il riutilizzo di vecchi elementi [Lévi-Strauss, 1962]. Lanzara definisce questa modalità di costruzione del mondo come una forma di progettazione basata su principi molto diversi rispetto ai classici principi costruttivi dell'ingegneria e dell'architettura [Lanzara, 2000]. Nel bricolage infatti non c'è un ordine prestabilito, ma esso emerge a posteriori, seguendo a una serie di azioni e interventi d'improvvisazione [Longo, 2000]. Questo modo di lavorare è tipicamente quello dei bambini che, trovandosi a dover risolvere problemi con il computer, formano le proprie abilità e le proprie conoscenze attraverso pratiche molto diverse rispetto a quelle standard che dominano all'interno della scuola. Tutto questo va oltre all'idea che imparare significhi semplicemente eseguire correttamente alcuni passaggi e operazioni, mettendo in evidenza l'importanza di sapersi adattare alle situazioni, saper scegliere strategie, provare soluzioni alternative e intraprendere iniziative per far fronte a situazioni complesse [Capponi, 2009]. Per fare una analogia tra l'ambiente di programmazione e il bricolage, possiamo prendere da esempio le costruzioni: giocando con esse, i pezzi vengono messi insieme secondo un'idea non troppo o per niente definita di quello che si vuole costruire; piano piano l'idea prende una forma sempre più chiara ed è possibile modellarla a proprio piacimento [Papert, 1996]. Anche nella programmazione si possono iniziare a costruire procedure con uno scopo generico, per poi smontarle, ricrearle o modificarle nel caso di eventuali inconvenienti, consentendo così di imparare a prendere atto dei propri ragionamenti. Il bricolage è un'attività progettuale e costruttiva basata sull'improvvisazione e sull'utilizzo dei materiali che sono a disposizione. In linea con l'idea costruzionista, è un apprendimento che avviene costruendo le proprie conoscenze, in un percorso di scoperta a volte casuale e imprevedibile, senza la progettazione di un percorso prestabilito. Programmando,

il bambino, allo stesso modo, non segue un piano preciso, ma procede per tentativi, secondo la tecnica del "bricolage intellettuale".

Nella programmazione, l'attività di costruzione del sapere è data dalla necessità di risoluzione di problemi reali in una simulazione digitale della realtà, un piccolo universo, un *micromondo* delimitato e sicuro. In esso si possono esplorare alternative diverse di risoluzione di un problema, sperimentare azioni per verificare quali hanno una migliore riuscita. Il micromondo incoraggia un apprendimento per scoperta, in cui le regole sono definite passo dopo passo da chi programma, che si accorge allo stesso tempo che esso funziona come il mondo reale. Costruendo un micromondo si capisce come un oggetto sia caratterizzato da proprietà sue proprie, statiche o dinamiche. Per questo anche i bambini più piccoli padroneggiano istintivamente questi strumenti [Marcianò³³]. Il processo di apprendimento quindi rimane più semplice quando si comprende che gli oggetti del micromondo si comportano come gli oggetti reali, ma che è possibile crearli e plasmarli a proprio piacimento.

2.5 L'epistemologia dell'errore

Caratteristica di un programma è quella di venire costruito passo dopo passo, modificando il codice quando la sua esecuzione non dà la soluzione che ci soddisfa o ricercando eventuali errori. Per definire il meccanismo di ricerca e individuazione degli errori viene usato il termine *debugging*. Cimentarsi nella programmazione di un computer intesa come momento di costruzione della conoscenza, mette in evidenza il significato positivo dell'errore e dell'importanza che assume nei processi di apprendimento, di comprensione e di soluzione dei problemi [Capponi, 2009]. I bambini, spesso, quando eseguono il programma e non sono soddisfatti, tendono a cancellare tutto il codice, per riscriverlo da capo, oppure abbandonare il progetto, che viene considerato sbagliato. Ma è la scuola che insegna che gli errori sono un male, elementi negativi. Papert sostiene che la dicotomia giusto/sbagliato in questo senso è controproducente: secondo questo

³³ <http://www.robocupjr.it/margi/pubblicazioni/1203.pdf> (consultato il 19 Agosto 2019).

modello di apprendimento, infatti, quando un bambino non capisce qualcosa viene bloccato nell'atto dell'imparare [Papert, 1984]. La paura di sbagliare può portare a stati d'ansia che fanno in modo che gli alunni sviluppino stati affettivi di negazione verso l'apprendimento e barriere psicologiche difficili da abbattere. Whitehead diceva: "Il panico dell'errore è la morte del progresso" [Capponi, 2009]. Evidenziare l'errore come un aspetto negativo porta infatti a una riduzione dell'energia e dell'interesse dei bambini, che non devono essere rimproverati per lo sbaglio commesso, ma vanno messi nella condizione di controllare l'errore, stabilendo un rapporto con esso [Montessori, 1966]. Il bug è protagonista dell'epistemologia di Papert, il quale riprende il pensiero di Popper, sostenendo che la conoscenza si accresce nella misura in cui impariamo dai nostri errori [Popper, 1959]; la correzione degli errori fa parte del processo di comprensione del programma e il bambino è incoraggiato a studiare il bug, non a cancellarlo [Papert, 1984]. Questa è la filosofia del debugging: studiare l'errore per capire cosa c'è che non va, cosa abbiamo sbagliato e perché, per riuscire a sistemare le cose.

L'ambiente di programmazione attribuisce all'errore un ruolo di notevole importanza nel processo di comprensione e soluzione dei problemi e nella costruzione della conoscenza. La ricerca e individuazione degli errori, nella maggior parte dei linguaggi di programmazione, avviene in una particolare struttura, quella tipica della programmazione, che rende più semplice e veloce il processo di debugging: il problema è suddiviso in parti più semplici in cui gli errori possono essere cercati e rettificati separatamente. Avere la possibilità di identificare e eliminare i bag passo dopo passo consente anche di prendere consapevolezza dei propri progressi. Inoltre nella programmazione, procedendo per tentativi, utilizzando la tecnica del bricolage intellettuale, non ci si aspetta che tutto funzioni al primo tentativo, non si considera sbagliato un programma mal riuscito, ma si affronta con l'idea di capire che cosa è successo e come poterlo migliorare [Andronico, 1996]. Questo processo di verifica acquista significato nella strutturazione del pensiero, un primo passo verso il miglioramento [Capponi, 2009].

2.6 L'apprendimento della geometria attraverso la programmazione

Molte ricerche si sono sviluppate in indagini su come l'uso della programmazione informatica influisce sulle conoscenze geometriche di base dei bambini. Precedentemente, abbiamo visto come Papert aveva sviluppato LOGO, in base alla cosiddetta Geometria della Tartaruga, per sottolineare il fatto che la geometria affrontata in un ambiente di programmazione del genere (con funzionalità grafiche di disegno) è una geometria "fisica", applicabile sul proprio corpo, sperimentabile in prima persona e che quindi è più facile da comprendere. Numerosi studi successivi hanno indagato su quale sia la relazione tra programmazione e apprendimento della geometria, in particolare il concetto di angolo. Nel 1990 Simmons e Cope hanno confermato che la programmazione rende la prospettiva di approccio alla geometria diversa rispetto alla prospettiva convenzionale della geometria piana [Simmons, Cope, 1990]. È stato dimostrato che gli ambienti della programmazione influenzano positivamente le abilità metacognitive di problem solving [Clements, Battista, 1989], ma ricerche recenti hanno indicato anche che, comunque, i giovani programmatori non si impegnano in un alto livello di costruzione e astrazione di idee matematiche in modo indipendente [Kieran, 1986]. Secondo la teoria dei due coniugi e matematici olandesi Van Hiele, l'apprendimento degli schemi geometrici passa attraverso determinati livelli di pensiero, che scandiscono la progressione verso l'astrazione. I livelli sono cinque, numerati da 0 a 4 e descrivono come gli studenti avanzano nel processo di apprendimento. Essi non dipendono strettamente dall'età: alcuni adulti possono non superare mai il primo livello, mentre sperimentando esperienze molto formative, un bambino può raggiungere un livello centrale già dalla scuola primaria. La gerarchia dei livelli è così organizzata³⁴:

- Livello 0: Visualizzazione. Le figure geometriche sono riconosciute come un tutto, ma non per le loro proprietà.
- Livello 1: Analisi. Si riconoscono le caratteristiche e le proprietà specifiche delle figure, che non sono però messe in relazione tra loro.

³⁴ In aggiunta a questi Clements e Sarama, 2000, hanno proposto un ulteriore livello che si sviluppa in modo precoce rispetto agli altri, il livello del pre-riconoscimento, che gli autori definiscono livello 0, numerando gli altri da 1 a 5 [Giofrè, Mammarella, Lucangeli, 2009].

- Livello 2: Deduzione informale. Si comprendono le relazioni logiche tra le proprietà, sia nelle figure che tra le figure.
- Livello 3: Deduzione formale. Si possono sviluppare successioni di affermazioni per dedurre un'affermazione dall'altra, si comprende il ruolo di assiomi, definizioni e teoremi.
- Livello 4: Rigore. A questo livello l'astrazione è massima, si studiano i sistemi geometrici deduttivi.

Un ambiente di programmazione come quello di LOGO, può facilitare il passaggio dal livello 0 della visualizzazione a quello 1, analitico-descrittivo, del pensiero geometrico. Infatti, per scrivere il codice di costruzione di una figura riconosciuta visivamente, chi programma deve anche analizzare gli aspetti visivi della figura e come sono unite le parti che la compongono e questa attività stimola il passaggio al livello 1 [Clements, Battista, 1989]. Molte attività possono inoltre facilitare lo sviluppo di specifici concetti geometrici come gli angoli, la loro misura e altre idee matematiche relative, considerando il ruolo che la rotazione (sia della tartaruga che di qualsiasi altro elemento programmabile) gioca nella costruzione di figure geometriche: nella formazione dei poligoni essa ha un ruolo centrale e programmando l'attenzione si focalizza su questa, consentendo a chi scrive il codice di capire meglio il ruolo della rotazione nella costruzione di un angolo, quindi di comprendere meglio il concetto stesso di angolo. Una ricerca effettuata da Simmons e Cope si è focalizzata sul verificare in che modo i bambini che hanno sperimentato LOGO sono in grado di definire un angolo tramite un comando del programma e di stimare la misura degli angoli. Il campione, composto da 59 bambini provenienti da due scuole diverse, ha avuto esperienze con LOGO per almeno tre mesi. Dai risultati è emerso che tutti i bambini sanno scrivere il codice per formare un quadrato, ma che solamente un quarto di essi sa fare lo stesso per il triangolo, questo perché è necessaria una conoscenza più profonda su come si costruiscono gli angoli nell'ambiente di LOGO. Essi hanno imparato che un certo numero scritto nel codice produce un certo effetto sullo schermo, ma non è chiaro se associno questo effetto all'angolo interno o a quello esterno. Gli autori sostengono comunque che le idee sbagliate di questo tipo

possono essere considerate un passaggio legittimo nell'apprendimento del concetto di angolo.

Clements e Battista, nella loro indagine sui possibili effetti positivi di LOGO, si focalizzano su diverse ipotesi, tra cui valutare come si sviluppa la concettualizzazione degli angoli e della loro misura nei bambini che programmano e quante *misconcezioni*³⁵ relative agli angoli riescono ad eliminare dopo aver sperimentato LOGO. In questo studio hanno preso come campione 48 bambini di 8 anni circa, dividendoli in due gruppi, uno di programmazione informatica con LOGO e un gruppo di controllo. Il gruppo sperimentale ha affrontato tre sessioni da 45 minuti per settimana, per un totale di 78 sessioni, costruendo e manipolando figure geometriche nell'ambiente di programmazione; il gruppo di controllo ha lavorato solamente per 26 sessioni con programmi di composizione, disegno e musica. I risultati sono stati valutati grazie a dei pre-test, per valutare le conoscenze di partenza e dei colloqui individuali. Da questi è emerso che, anche se non ci sono state differenze sostanziali tra i due gruppi per quanto riguarda le forme in generale, i bambini del gruppo sperimentale hanno avuto prestazioni migliori nel compito di identificazione degli angoli; essi hanno superato leggermente l'altro gruppo anche per quanto riguarda il confronto tra gli angoli e la loro misura. Inoltre hanno riscontrato che le *misconcezioni* presenti nei bambini del gruppo di controllo erano più numerose.

In un'altra indagine, Noss ha portato avanti uno studio per valutare gli effetti di LOGO sulla comprensione dei concetti di lunghezza dei lati e di angolo. L'approccio adottato dall'autore prevedeva di confrontare la performance su specifiche abilità geometriche degli alunni che hanno studiato il linguaggio LOGO con quella degli alunni che non lo hanno utilizzato. Il test è stato effettuato in quattro classi in cui è avvenuta la sperimentazione e in quattro classi di controllo, dopo un anno di lavoro. Per quanto riguarda gli angoli, la ricerca si è focalizzata sul concetto di conservazione dell'angolo, nello specifico dell'angolo retto e della sua misurazione, cioè il riconoscimento di angoli più grandi e più

³⁵ D'Amore (1999) ha dato la seguente definizione: «Una *misconcezione* è un concetto errato e dunque costituisce genericamente un evento da evitare; essa però non va vista sempre come una situazione del tutto o certamente negativa: non è escluso che per poter raggiungere la costruzione di un concetto, si renda necessario passare attraverso una *misconcezione* momentanea, ma in corso di sistemazione».

piccoli. Nelle conclusioni Noss specifica che i risultati riguardanti gli angoli sono più marcati rispetto a quelli relativi alle lunghezze e che c'è stato un significativo orientamento a favore del lavoro con LOGO relativamente agli aspetti della conservazione e della misura degli angoli: i bambini che hanno sperimentato la programmazione si sono rivelati migliori a ignorare l'orientamento e la lunghezza dei lati nel valutare la misura di due angoli a confronto.

I risultati, in generale, indicano che l'ambiente di programmazione arricchisce la concettualizzazione geometrica dei bambini e rende più sofisticato il pensiero geometrico, se affrontato in un contesto adeguato in cui l'insegnante prende il ruolo di guida e in un clima di collaborazione e interazione tra pari. Questi risultati supportano l'uso della programmazione come quadro concettuale per l'apprendimento degli aspetti della geometria finora descritti, dato che l'ambiente educativo della programmazione è strutturato in modo specifico per guidare l'esperienza dei bambini in questo senso.

2.7 Linguaggi visuali e testuali a confronto

Alcuni studi sono stati portati avanti per rispondere alla domanda su quale linguaggio testuale sia migliore per i principianti, ma da quando è emersa una alternativa ai linguaggi convenzionali, che vede entrare nelle aule informatiche il nuovo linguaggio grafico basato su modalità visuale, la ricerca si è posta sull'argomento con un nuovo punto di vista. Gli ambienti di programmazione basati sui linguaggi a blocchi stanno diventando sempre più comuni nei contesti informatici sia formali che informali, fornendo nuove funzionalità e vantaggi di tipo pedagogico. Il vantaggio di un linguaggio di programmazione visuale di eliminare gli errori dal punto di vista sintattico fa di questa modalità quella favorita nei contesti educativi in cui è introdotta questa tipologia di sviluppo del pensiero computazionale. La progettazione di ambienti didattici adeguati per la programmazione e le pratiche di insegnamento in classe necessitano quindi di indagini specifiche per valutarne gli effetti non solo dal punto di vista didattico. La questione affrontata oggi e già da qualche tempo dagli educatori informatici è

se i nuovi sistemi grafici di rappresentazione dei linguaggi di programmazione comportino qualche differenza significativa rispetto a quelli testuali. Alcune ricerche si sono concentrate su sperimentazioni e analisi per rivelare le differenze tra linguaggio testuale e visuale.

2.7.1 Lo studio di David Weintrop e Uri Wilensky

Uno studio dei due autori Weintrop e Wilensky (2015), si è basato sul metodo di indagine definito *Commutative Assessment*: un metodo per valutare se e come le modalità di programmazione influenzano l'apprendimento, progettato nello specifico per misurare la comprensione dei concetti della programmazione nelle due modalità, a blocchi e testuale. Lo studio ha coinvolto novanta studenti di tre classi di scuola superiore, per la durata di 10 settimane in cui è avvenuto il corso introduttivo alla programmazione; ogni classe ha passato le prime cinque settimane lavorando con un linguaggio a blocchi, per poi passare al linguaggio testuale; le valutazioni sono state effettuate all'inizio, a metà e alla fine del percorso. Un primo articolo si è incentrato sul confronto della comprensione degli studenti nelle due diverse modalità di scrittura del codice e sul valutare se queste influenzano significativamente la comprensione dei concetti di base della programmazione. Alla prima domanda risponde la modalità stessa di indagine; infatti il *Commutative Assessment* dà la possibilità di confrontare direttamente le risposte alle domande nelle due modalità e in momenti diversi della sperimentazione. I risultati mostrano un significativo vantaggio della modalità grafica basata sui blocchi relativamente alla comprensione dei concetti di base della programmazione; per quanto riguarda le domande relative alla comprensione generale di un programma, i dati invece non mostrano differenze tra le due modalità.

Il secondo articolo pubblicato da Weintrop e Wilensky (2015a) mira a conoscere quale sia la percezione degli studenti delle due diverse modalità di programmazione. Basandosi sul medesimo studio descritto sopra, gli autori si sono posti tre domande: se la programmazione a blocchi sia più facile per gli

studenti e perché; quali sono le differenze principali rilevate; quali sono gli svantaggi percepiti nella programmazione a blocchi. Per rispondere a queste domande gli autori si sono basati sulle risposte a delle interviste, svolte durante lo studio, divise anch'esse in tre tempi distinti (all'inizio, a metà e alla fine del percorso). I dati emersi indicano che gli studenti trovano l'approccio di programmazione basato sui blocchi più semplice di quello testuale e dalle interviste sono emerse quattro principali motivazioni. Il primo aspetto evidenziato è stata la maggiore facilità di lettura delle etichette descrittive sui blocchi e la loro comprensione. La seconda caratteristica è identificata nella stessa natura visiva dei blocchi, nei segnali grafici che essi danno per essere utilizzati nel modo giusto; gli studenti hanno espressamente affermato che la forma stessa dei blocchi consiglia in che modo essi possono essere utilizzati, indicando chiaramente se e come possono essere incastrati tra loro. Il terzo aspetto è la facilità di composizione del codice, data dalla possibilità di scegliere e trascinare i blocchi nel dato spazio, al posto di doverlo scrivere di proprio pugno. L'ultima caratteristica evidenziata dagli studenti è la capacità del linguaggio visuale di rendere meno necessaria la memorizzazione di ciò che è necessario fare, dato che le indicazioni date dalle etichette sui blocchi aiutano a ricordare o a capire come è possibile costruire il programma per avere determinate soluzioni. Le maggiori differenze identificate dallo studio di Weintrop e Wilensky tra il linguaggio visuale e quello testuale sono state valutate in base a domande presentate a metà e alla fine dello studio. Oltre alle categorie esplicitate per rispondere alla prima domanda, sono state evidenziate dagli studenti differenze come la presenza di comandi preimpostati, la facilità di programmare per tentativi ed errori e i differenti tipi di programmi possibili con il linguaggio a blocchi rispetto a quello testuale. Nelle risposte date a metà percorso, senza ancora aver lavorato con il linguaggio testuale, le affermazioni degli alunni sono state relative al fatto che con questo secondo tipo di linguaggio sarebbe stato necessario scrivere il codice senza l'aiuto dei blocchi, e questo avrebbe voluto dire «*pensare di più*», anche per quanto riguarda la ricerca e correzione degli errori. Alcuni studenti hanno addirittura considerato il passaggio al linguaggio testuale e l'adattamento ad esso come un momento difficile. Alla fine dello studio, gli autori notano invece che la differenza relativa

alla rappresentazione visiva dei comandi perde di significato, e acquisisce invece importanza la semplicità del linguaggio dei blocchi (in questo caso l'inglese) che supporta il raggiungimento degli obiettivi preposti.

Per quanto riguarda l'ultima domanda postasi dagli autori, relativa agli svantaggi percepiti nella programmazione a blocchi, sono emerse delle risposte molto precise degli studenti che mettono in discussione il fatto che questo linguaggio sia privo di problematiche. Le principali critiche rilevate sono tre. La prima è la sensazione che la programmazione grafica sia meno potente: per gli studenti con il linguaggio testuale si possono fare «*molte più cose*» e creare programmi più complessi rispetto a quelli costruibili con i blocchi. Gli studenti hanno percepito la programmazione visuale come una semplificazione di quella testuale. Il secondo svantaggio è relativo alla lentezza e al maggior numero di blocchi necessari per costruire un codice rispetto alla scrittura manuale di un programma; per alcuni studenti è risultato più facile da comprendere un programma complesso se testuale, che risulta meno confuso. L'ultima caratteristica che è stata evidenziata è il fatto che gli studenti hanno percepito e descritto la modalità visuale come non autentica, meno "vera" rispetto a quella testuale. Questa affermazione è risultata la più compromettente per quanto riguarda l'efficacia dell'utilizzo degli strumenti basati sui blocchi, nell'introduzione alla programmazione a scuola.

2.7.2 Lo studio di Lewis: Logo vs. Scratch

Colleen Lewis, nel suo studio a riguardo dei metodi visuali e testuali, mette a confronto due dei linguaggi più utilizzati oggi, Logo e Scratch. La ricerca in questo caso è volta a valutare la valenza educativa dell'ambiente Scratch rispetto allo strumento più conosciuto e studiato, l'ambiente Logo. L'autrice vuole confutare le seguenti ipotesi:

- La maggiore facilità della programmazione con Scratch.
- La maggiore sicurezza nelle proprie competenze di scrivere programmi e l'intenzione di continuare lo studio della programmazione.

- La maggiore capacità di comprendere e controllare costrutti come i cicli e le istruzioni condizionali.

Il suo studio ha coinvolto due gruppi di studenti tra i dieci e i dodici anni, in un corso estivo di programmazione della durata di 12 giorni. I gruppi hanno lavorato con i due software in modo alternato: un gruppo (*Logo-First*) è stato introdotto alla programmazione con Logo mentre l'altro (*Scratch-First*) con Scratch; dopo sei giorni i gruppi si sono invertiti. Per raccogliere i dati, sono state fornite delle prove di valutazione scritte durante il corso, per valutare la comprensione dei concetti e delle tecniche apprese, inoltre è stata fatta un'indagine rivolta agli studenti riguardante la valutazione dell'esperienza svolta, i loro interessi e i loro obiettivi. Per analizzare i risultati, l'autrice ha riunito le risposte degli alunni in istogrammi che mettono a confronto il due gruppi. Prendendo in considerazione le domande relative alla difficoltà di apprendimento delle funzionalità del programma, l'autrice ha notato che gli studenti che hanno utilizzato Logo hanno trovato maggiore difficoltà. Le domande rappresentative della ricerca riguardano l'abilità nell'utilizzo del computer. Il confronto tra i due gruppi non mostra differenze rilevanti, anche se le risposte positive nel gruppo Logo-First sono superiori di uno o due punti rispetto al gruppo Scratch-First. Allo stesso modo, è risultata significativamente superiore la percezione degli studenti del gruppo Logo-First di aver acquisito più competenze di programmazione, trovando anche più facile la scrittura di un programma. Al contrario dell'ipotesi di partenza, quindi la ricerca dell'autrice ha rivelato che gli alunni che hanno seguito il percorso con Logo hanno in media una maggiore fiducia nelle proprie capacità di programmare e che imparare con il linguaggio testuale, piuttosto che con Scratch, non incide sulla scelta di continuare a programmare anche in futuro.

Il confronto tra Logo e Scratch ha dato il via a uno studio di ricerca che si sta portando avanti in diversi contesti, permettendo di allargare il campione e di avere risultati sempre più precisi e significativi. Il capitolo successivo, riprendendo questa metodologia, descrive nel dettaglio la sperimentazione dei due linguaggi in una scuola primaria, sperimentazione che si basa sugli studi finora descritti.

Capitolo 3

Un laboratorio di coding alla scuola primaria

3.1 Il progetto

Questa attività si inserisce in un progetto di sperimentazione più ampio, iniziato in precedenza da alcune colleghe, ex studentesse di Scienze della Formazione Primaria, che lo hanno applicato in diverse scuole primarie della Toscana. La scelta di lavorare sul coding è dettata da diversi fattori: in generale, dalla opportunità importante di poter fare didattica utilizzando strumenti che, anche se usati quotidianamente, sono in realtà poco conosciuti dal punto di vista del loro funzionamento e non vengono considerati per le quasi infinite possibilità di utilizzo che possono avere nei processi di insegnamento-apprendimento; nello specifico, dal fatto che i linguaggi di programmazione, come spiegato precedentemente, sono un valido aiuto all'insegnamento, se affrontati in un contesto di scoperta, di sperimentazione e di condivisione. Infatti, la possibilità di poter andare avanti per tentativi, in un'ottica in cui l'errore non mortifica, ma stimola la ricerca della soluzione, costituisce un'ottima opportunità di sviluppo di competenze, ma anche di costruzione dell'autostima e della consapevolezza delle proprie capacità.

3.2 Finalità

Lo scopo del progetto è osservare i risultati dell'uso di due software di programmazione diversi, per valutarne gli effetti sull'apprendimento della geometria. Più precisamente, si cerca di capire in che misura l'utilizzo di determinati programmi può aiutare ad eliminare alcune misconcezioni che si consolidano nei bambini riguardo agli angoli; inoltre il progetto si struttura in

modo da poter confrontare i risultati dell'efficacia di due tipologie diverse di programmazione. I programmi utilizzati sono LibreLogo, estensione di LibreOffice, dove il linguaggio di programmazione è testuale e Scratch nella versione online, che usa invece un linguaggio visuale a blocchi. Inoltre Logo richiede la scrittura di comandi in lingua inglese.

3.3 Contesto

La sperimentazione ha coinvolto le due classi quinte (sezione A e B) della Scuola Primaria San Lino, plesso dell'Istituto Comprensivo Volterra, in cui ho svolto anche il tirocinio. La classe 5A è composta da 20 alunni, tra cui sono presenti due bambini con DSA e tre bambini certificati dalla legge 104, nello specifico un bambino con sindrome di Tourette, una bambina con sindrome di Asperger e un bambino con disturbi del linguaggio, quindi è presente una insegnante di sostegno. La classe 5B è composta da 23 alunni tra cui è presente un solo bambino certificato dalla legge 104, un solo bambino con DSA e due bambini stranieri arrivati nell'anno scolastico corrente. Uno dei due bambini stranieri ha già acquisito una corretta conoscenza della lingua italiana, mentre l'altro, arrivato successivamente, presenta ancora grandi difficoltà di comprensione e comunicazione. Anche in questa classe è presente l'insegnante di sostegno. Il totale del campione quindi è composto da 43 alunni. Durante le attività le due classi sono rimaste nelle rispettive aule, le quali sono state preparate all'occorrenza con dei computer portatili di proprietà della scuola, uno per ogni coppia di bambini. La classe 5B ha iniziato l'attività lavorando con LibreLogo (chiamerò la classe *Gruppo Logo*), mentre la 5A ha iniziato con Scratch (chiamerò la classe *Gruppo Scratch*).

3.4 Descrizione delle attività

Il progetto si struttura in 12 lezioni laboratoriali con l'uso del computer, mentre 6 ore circa sono dedicate alla compilazione dei test. Ogni classe ha sperimentato l'uso di LibreLogo e Scratch nell'arco di 6 lezioni della durata di un'ora, lavorando per tre incontri con un programma e per i successivi tre con l'altro. Inoltre sono stati somministrati dei test che hanno richiesto ognuno la durata di circa un'ora: un pre-test (test 1) all'inizio, un test intermedio (test 2) a metà e un test finale (test 3) alla fine del percorso. Per concludere è stato consegnato un questionario di valutazione dell'esperienza e sull'uso delle tecnologie. La durata del progetto è stata la stessa per entrambe le classi e il progetto si è svolto in parallelo, settimanalmente, per un totale di 9 ore per sezione, nei mesi di aprile e maggio 2019.

3.5 Progettazione delle attività

Le discipline coinvolte nel progetto sono Matematica/Geometria, Tecnologia e Inglese. Per ogni disciplina, i rispettivi traguardi per lo sviluppo delle competenze e gli obiettivi di apprendimento sono ricavati dalle *Indicazioni Nazionali per il curriculum della scuola dell'infanzia e del primo ciclo d'istruzione* del 2012.

Matematica

Traguardi per lo sviluppo delle competenze:

- L'alunno riconosce e rappresenta forme del piano e dello spazio.
- L'alunno descrive, denomina e classifica figure in base a caratteristiche geometriche, ne determina misure, progetta e costruisce modelli concreti.
- L'alunno sviluppa un atteggiamento positivo rispetto alla matematica, attraverso esperienze significative

Obiettivi di apprendimento:

- Descrivere, denominare e classificare figure geometriche, identificando elementi significativi.
- Riprodurre una figura in base a una descrizione, utilizzando gli strumenti opportuni (software informatici).
- Confrontare e misurare angoli.

Inglese

Traguardi per lo sviluppo delle competenze:

- L'alunno comunica con espressioni memorizzate

Obiettivi di apprendimento:

- Comprendere e scrivere i vocaboli: "forward", "right", "left", "repeat", "home", "clearscreen".

Tecnologia

Traguardi per lo sviluppo delle competenze:

- L'alunno si orienta tra i diversi mezzi di comunicazione ed è in grado di farne un uso adeguato a seconda delle diverse situazioni.
- L'alunno produce semplici modelli o rappresentazioni grafiche del proprio operato utilizzando strumenti multimediali.

Obiettivi di apprendimento:

- Cercare, selezionare sul computer un comune programma di utilità.
- Riconoscere e documentare le funzioni principali di una nuova applicazione informatica.

Tempi: Le attività sono suddivise in diverse lezioni della durata di circa 1 ora, una volta alla settimana, da svolgersi in giorni stabiliti con le insegnanti nei mesi di

aprile e maggio. Anche la durata dei test è di un'ora circa, mentre il questionario è stato somministrato insieme al test 3 e ha richiesto circa 30 minuti per classe.

- Pretest
- Attività 1 - Gruppo Logo --> Scopriamo Logo; comandi principali
- Attività 2 - Gruppo Scratch --> Scopriamo Scratch; comandi principali
- Attività 3 - Gruppo Logo --> Comando REPEAT
- Attività 4 - Gruppo Scratch --> Blocco delle ripetizioni
- Attività 5 - Gruppo Logo ---> Costruiamo figure con Logo
- Attività 6 - Gruppo Scratch --> Costruiamo figure con Scratch
- Test intermedio
- Attività 7 - Gruppo Logo --> Scopriamo Scratch; comandi principali
- Attività 8 - Gruppo Scratch --> Scopriamo Logo; comandi principali
- Attività 9 - Gruppo Logo --> Blocco delle ripetizioni
- Attività 10 - Gruppo Scratch --> Comando REPEAT
- Attività 11 - Gruppo Logo --> Costruiamo figure con Scratch
- Attività 12 - Gruppo Scratch --> Costruiamo figure con Logo
- Test finale
- Questionario (valutazione dell'esperienza)

Strategie: La strategia utile ai fini del progetto è quella della didattica laboratoriale; inoltre sono state utilizzate strategie come *cooperative learning*, *peer tutoring* e *problem solving*.

Spazi e Strumenti: L'attività si è svolta in classe, con i banchi sistemati a coppie o gruppi di 3. Gli strumenti necessari per la realizzazione dell'attività sono stati i PC portatili con su installato LibreOffice, la LIM e la connessione internet del plesso. Inoltre i bambini potevano utilizzare fogli e penna per appuntare tutto ciò che ritenevano utile.

Modalità di verifica: Essendo un laboratorio, la verifica si è svolta principalmente in itinere, osservando la partecipazione e l'impegno dei bambini durante lo svolgimento delle attività.

Anche i test svolti durante e alla fine dell'attività sono stati strumento di valutazione, oltre ad essere tenuti in considerazione ai fini della sperimentazione.

Attività 1 – Gruppo Logo: Scopriamo Logo; comandi principali

Prima di presentare ai bambini il programma è stato opportuno riattivare le preconcoscenze. Poiché entrambe le classi avevano in precedenza già sperimentato il coding e la programmazione dei robot li aveva molto appassionati, è stato un ripasso piuttosto semplice e veloce e hanno dimostrato di avere dimestichezza in materia. Ho chiesto loro di darmi delle indicazioni per spostarmi nell'aula e arrivare alla porta e subito mi hanno suggerito di andare “avanti di 10 passi, no aspetta, torna indietro di due, poi gira a destra!” a questo punto ho iniziato a girare su me stessa e quando mi hanno fermato ho chiesto loro di quanto dovevo girare, così mi hanno risposto che dovevo girare a destra di 90°. In questo modo mi hanno fatto raggiungere la destinazione. Dopo aver chiesto cosa è il coding ho ricevuto risposte diverse, ma la più frequente e sulla quale erano tutti d'accordo è stata “è quando si dice al robot dove andare”. Questo mi ha fatto capire che dovevo spiegare loro cosa si intende per programmazione, o meglio, per programmazione informatica e così ho detto loro che avremmo imparato a programmare in modo semplice usando il computer per scrivere dei codici, ovvero le indicazioni che avevano usato per farmi muovere. I bambini erano già divisi per coppie e ogni coppia aveva a disposizione il computer portatile, così lo hanno acceso e ho chiesto di cercare nella barra di ricerca di Windows LibreOffice, quindi di aprire LibreOffice Writer. Tutti sono stati in grado di trovarlo così ho fatto subito aggiungere la barra degli strumenti di Logo, aiutando chi non riusciva a vederla intera a spostarla. Per prima cosa ho insegnato il comando `SHOWTURTLE`” così che tutti potessero vedere la tartaruga sullo schermo, poi ho spiegato che questa tartaruga sa disegnare e avremmo potuto dirle noi cosa e come disegnare, ma ho anche detto che capisce solo l'inglese. I bambini si sono entusiasmatisi dicendo che conoscevano bene l'inglese. Quindi ho presentato i primi comandi: `FORWARD`,

BACK, RIGHT, LEFT, ma ho specificato che ad ogni comando dovevano anche indicare di “quanti passi” andare avanti e quanti gradi ruotare. Le tartarughe si muovevano e tutti hanno iniziato a sperimentare i comandi; qualcuno inserendo troppi passi ha fatto in modo che la tartaruga uscisse dallo schermo, così ne ho approfittato per spiegare anche altri comandi, come HOME, per far tornare la tartaruga nel centro del foglio e CLEARSCREEN. I bambini hanno subito annotato i comandi, così ho chiesto loro di scrivere anche i codici di quello che avrebbero disegnato per usarlo in futuro. A questo punto, vedendo gli alunni a loro agio con il programma, ho chiesto loro di provare a disegnare un quadrato. Non hanno trovato grandi difficoltà e anche se sullo schermo appariva il segnale di un errore, sono riusciti a capire da soli cosa avevano sbagliato e quindi a correggerlo. Quasi tutti i gruppi sono riusciti a costruire il quadrato già durante questa prima lezione.

Attività 2 – Gruppo Scratch: Scopriamo Scratch; comandi principali

Per svolgere questa lezione è stato necessario creare un account docente sul sito di Scratch e creare un link che poi ho stampato e consegnato ad ogni coppia di bambini, così sono potuti entrare creando un account studente, che permette di salvare le creazioni per poterle rivedere in futuro. L’attività si è strutturata nello stesso modo del gruppo Logo, quindi siamo partiti con un ripasso sul coding e su una piccola spiegazione sulla programmazione e su cosa avremmo imparato. Poi i bambini hanno acceso i computer e sul browser hanno inserito il link che avevo consegnato loro. Grazie alla presenza della LIM in classe ho potuto mostrare loro passo dopo passo ciò che dovevano fare. Questa parte di preparazione ha richiesto qualche minuto, ma quasi tutti i bambini sono riusciti ad entrare sul sito al primo tentativo e a registrarsi. Quando tutti sono arrivati sulla pagina di editor del codice ho spiegato che avremmo imparato a far muovere il gattino quindi ho fatto notare che sulla sinistra dello schermo c’erano dei blocchetti con su scritto delle azioni: trascinando i blocchetti sulla pagina bianca e facendo doppio clic su di essi il gatto avrebbe compiuto quella azione. Inoltre la loro forma suggeriva la possibilità di

combinarli tra loro come un puzzle. Il primo blocco che ho mostrato è stato il “fai ... passi”, poi, visto che qualcuno ha inserito un numero troppo grande di passi e il gattino è uscito dalla visuale, ho spiegato che aggiungendo il blocco “vai a x: ... y: ...” sopra quello dei passi e inserendo i valori $x=0$ e $y=0$ il gattino sarebbe tornato al centro del foglio. Ho suggerito subito di usare anche il blocco delle situazioni che permette di avviare il codice cliccando sulla bandierina. Ho lasciato provare ai bambini questi comandi fino a che alcuni mi hanno detto che avevano difficoltà poiché le scritte erano in lingua inglese, così li ho aiutati a modificarla in italiano usando il tasto posto in alto sulla pagina che rappresenta il mappamondo. Successivamente ho spiegato che con questo programma avremmo potuto anche disegnare, quindi abbiamo aggiunto l’estensione penna, il blocco “penna giù” e “fai ... passi”, in questo modo abbiamo tracciato le prime linee. A questo punto il tempo a disposizione era finito quindi ho dovuto concludere l’attività dicendo ai bambini che la volta successiva avremmo imparato a costruire una casetta. Con questo gruppo, a causa delle problematiche interne alla classe, ho dovuto lavorare più lentamente e con diverse modalità, per permettere a tutti di capire e di provare ad usare il programma.

Attività 3 – Gruppo Logo: Il comando REPEAT

Il secondo incontro con Logo è iniziato con un breve ripasso sui comandi principali del programma e sul modo per disegnare il quadrato. In questo modo anche chi non era riuscito a costruirlo nella lezione precedente ha potuto riprovare e capire come fare. Visto che i bambini erano molto coinvolti, ho deciso di suggerire il comando REPEAT, scrivendolo alla lavagna. È bastato far loro notare che avevano scritto le stesse cose 4 volte e in molti hanno provato subito ad usarlo. Le figure che sono venute fuori da questa lezione sono state molte, tutte diverse e molto creative. Poiché alcune coppie, scrivendo il comando REPEAT senza specificare il numero di ripetizioni, si sono accorti che la tartaruga non si fermava più, ma che la figura si avvicinava molto ad un cerchio, mi hanno chiesto come potevano riuscire a farlo così ho voluto accontentarli.

Prendendo come esempio l'esperienza descritta nel "Piccolo Manuale di LibreLogo"³⁶ ho chiesto ai bambini di immaginare di essere la tartaruga e di provare a muoversi come farebbe lei, per capire come comandarla. Così ho fatto camminare i bambini in cerchio, dicendo loro di mettere un piede davanti l'altro ad ogni passo e sono arrivati alla conclusione che la tartaruga fa un passo avanti e gira un poco ad ogni passo. Alla domanda "Quanto gira ad ogni passo?" mi hanno risposto "Un grado!" quindi hanno provato a scriverlo nel codice di LibreLogo. Qualcuno ha usato il comando REPEAT [FORWARD 1 RIGHT 1] senza indicare il numero di ripetizioni e la tartaruga non smetteva più di girare, così ho chiesto loro di ragionare su quante ripetizioni dovevano inserire, per costruire un cerchio, se ad ogni passo girava solo di 1°. Subito sono arrivati alla soluzione, così ho chiesto anche cosa sarebbe successo se al posto di 360 avessero scritto 180. Divertiti mi hanno risposto: "Mezzo cerchio!". Prima che la lezione si concludesse ho chiesto ai bambini di provare a costruire un triangolo equilatero, così da poterci lavorare nella lezione successiva. Alcuni bambini hanno subito notato che facendo ruotare la tartaruga di 60°, non ottenevano il triangolo richiesto, quindi ho promesso loro di spiegarlo nella lezione successiva.

Attività 4 – Gruppo Scratch: Il blocco delle ripetizioni

Nella seconda lezione del gruppo Scratch ho chiesto ai bambini spiegare brevemente cosa avevamo sperimentato la volta precedente. Poi ho chiesto di provare a disegnare un quadrato usando i blocchetti "fai ... passi" e "ruota di ... gradi" a destra o a sinistra e tutti hanno inserito il valore 90. I bambini hanno iniziato a creare le forme geometriche e i più veloci hanno iniziato a sperimentare nuove combinazioni. In molti sono riusciti a far "miagolare" il gatto grazie all'apposito blocco. Durante la lezione ho notato una difficoltà generale: nonostante abbia mostrato come incastrare tra loro i blocchi, molti bambini li hanno posizionati sparsi nella pagina, cliccando ogni volta su quello che volevano attivare. Sono riusciti lo stesso a creare il quadrato e, con mio stupore, hanno

³⁶ A. R. Formiconi, *Piccolo manuale di LibreLogo*, 2018, capitolo 11.

mostrato molta soddisfazione, anche se durante la costruzione si potevano notare l'impegno e la fatica. Un altro aspetto critico che si è manifestato è stato l'uso del "pulisci": diverse coppie hanno aggiunto il blocco in una posizione del codice sbagliata, quindi la figura veniva costruita solo per metà, oppure si cancellava del tutto. In questo modo sono riuscita a far capire a tutti che il codice, se costruito correttamente, viene eseguito nell'ordine in cui vengono posizionati i blocchi e che quindi l'ultima azione che il gatto esegue corrisponde all'ultimo blocco inserito. Ho deciso quindi di concludere la lezione insegnando ad usare il blocco delle ripetizioni, mostrando a tutti come inserirlo e come togliere o aggiungere i comandi al suo interno.

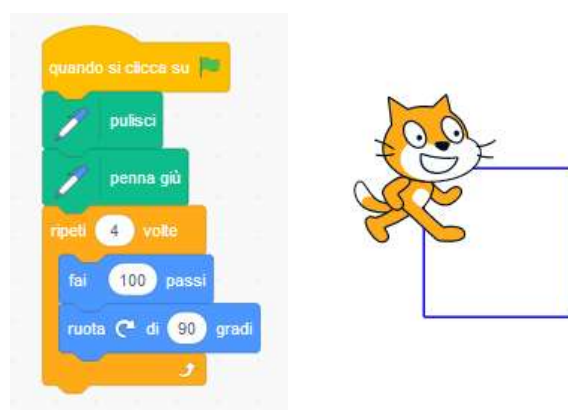


Figura 7 - Un utilizzo del blocco delle ripetizioni per la costruzione del quadrato

Attività 5 – Gruppo Logo: Costruiamo figure con Logo

Nell'ultimo incontro con Logo ho scelto di iniziare l'attività facendo disegnare ai bambini un angolo retto. Dopo ho chiesto di disegnare invece un angolo che fosse più grande o più piccolo. Ovviamente alcuni bambini hanno aumentato i gradi di rotazione mentre altri hanno diminuito il dato, a questo punto ho chiesto ad una coppia di bambini di dirmi il loro codice: FORWARD 40 RIGHT 15 FORWARD 40. Ho chiesto che tipo di angolo avevano disegnato e mi hanno risposto che era un angolo ottuso, allora ho chiesto di quanti gradi avevano fatto ruotare la tartaruga. Quando mi hanno risposto ho fatto notare che l'angolo formato era molto più ampio di 15°, così ho disegnato l'angolo alla lavagna, evidenziando la

rotazione e ho chiesto secondo loro quanto poteva misurare l'angolo formato. Dopo brevi ragionamenti tutti hanno capito che dovevano sottrarre a 180 i gradi di rotazione. Così ho chiesto loro di disegnarli un triangolo equilatero. Alcuni bambini hanno trovato subito la soluzione, altri mi hanno detto che provando con 60° non tornava e quindi hanno riprovato, riuscendoci. Visto che non hanno trovato nessuna difficoltà ho dato un ultimo comando: disegnare il triangolo equilatero con la base orizzontale e costruire una casetta. Più della metà della classe è riuscita da sola a capire come unire il quadrato al triangolo in modo da formare la casetta e alcuni bambini hanno aggiunto anche la porta. Ho voluto inoltre suggerire l'uso del comando PENCOLOR "... " e di aggiungere il colore che preferivano, così da poter cambiare il colore del tratto disegnato dalla tartaruga. Ho lasciato molta libertà di sperimentare i comandi che avevo insegnato, così che anche i bambini che erano rimasti più indietro potessero arrivare a un risultato per loro soddisfacente, senza che provassero l'ansia di non essere riusciti a capire che cosa fare o di non riuscire a fare quello che avevano in mente.

```

Clearscreen
home
pencolor "yellow"
repeat 4 [ forward 100
left 90 ]
forward 100
left 90
forward 100
pencolor "red"
right 120
forward 100
right 120
forward 100

```

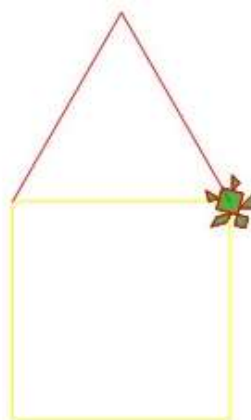


Figura 8 - Casetta costruita usando il "Repeat" e il colore della penna

Attività 6 – gruppo Scratch: Costruiamo figure con Scratch

Nella terza lezione del gruppo Scratch sono tornata a riflettere insieme ai bambini sulla costruzione del quadrato, ripassandone il codice quindi i blocchi da usare. A questo punto tutti i bambini lo avevano costruito e ho chiesto loro di pensare

adesso come potevamo disegnare un triangolo equilatero. Poiché il tempo a disposizione con questa classe era più breve (anche a causa dell'uso della connessione internet) non ho potuto spiegare la costruzione del triangolo come nell'altro gruppo, ma ho comunque chiesto loro di provare a inserire i blocchi del codice e quando mi hanno chiesto di quanti gradi dovevano far ruotare il gatto ho spiegato loro la regola, quindi che dovevano calcolare i gradi sottraendo l'ampiezza da creare a 180. Non tutti i bambini hanno capito subito come fare quindi sono dovuta passare tra i gruppi ad aiutare chi era rimasto indietro, ma alla fine tutti sono riusciti a costruire il triangolo. Quando ho chiesto di provare a costruire una casetta, senza suggerire di unire le due figure che avevamo imparato a disegnare, mi sono molto stupita poiché alcuni bambini sono riusciti a capire da soli come unire il triangolo al quadrato. Avendo avuto meno tempo a disposizione, il gruppo ha avuto poca possibilità di sperimentare autonomamente le funzionalità del programma, a parte alcuni bambini più veloci che hanno scoperto da soli il blocco del colore, come cambiare il tratto della penna o addirittura come aggiungere o sostituire uno *Sprite*. In conclusione posso comunque affermare che durante l'attività i bambini sono stati molto partecipativi e si sono divertiti molto anche se alcuni hanno trovato qualche difficoltà.

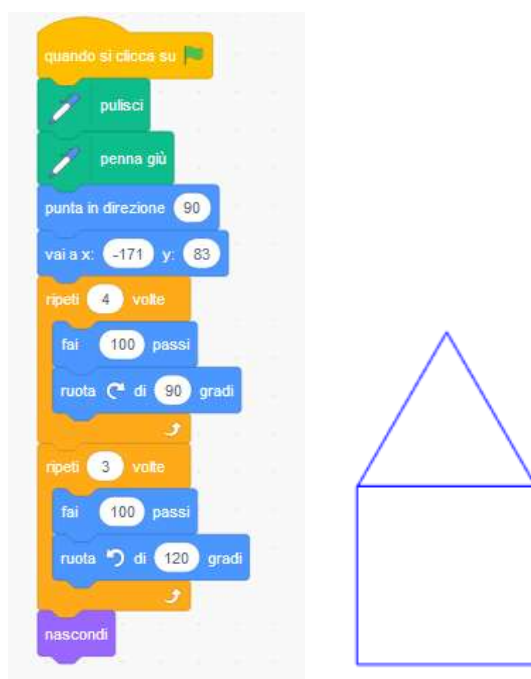


Figura 9 - Costruzione della casetta utilizzando il blocco della posizione e della ripetizione

Attività 7 - Gruppo Logo: Scopriamo Scratch; comandi principali

Dopo aver sperimentato LibreLogo, con questa classe abbiamo iniziato a scoprire il programma Scratch. Allo stesso modo del gruppo precedente ho consegnato ai bambini il link da inserire nella barra degli indirizzi del browser, per iscriversi alla classe che avevo creato precedentemente. Anche in questo caso abbiamo perso qualche minuto della lezione, ma non è stato un problema, visto che i bambini hanno dimostrato molta dimestichezza anche con questo tipo di programmazione. Dopo aver spiegato la struttura del programma i bambini si sono sentiti molto a loro agio nell'usare i blocchi dei comandi. Avendo già imparato a scrivere il codice autonomamente, partendo da zero e per di più in inglese, quasi tutti hanno affermato che per loro è stato più semplice usare i blocchi di Scratch, che sono apparsi come più intuitivi e immediati rispetto al movimento che volevano programmare. Dopo aver spiegato che il codice si avviava facendo doppio click sul blocco, gli alunni si sono accorti che il gattino di Scratch non disegnava, così ho spiegato come aggiungere il comando "penna giù", consigliando di aggiungere insieme anche il blocco "pulisci". Quando ho chiesto secondo loro dove andava posizionato questo blocco e a cosa serviva, subito mi hanno risposto che corrispondeva al "clearscreen" di LibreLogo. Inoltre, ho consigliato anche di inserire il blocco "quando si clicca su bandierina". Ho lasciato i bambini liberi di esplorare tutte le funzionalità, per prendere confidenza con in programma; anche in questa classe hanno subito capito come far miagolare il gattino e come cambiargli il colore. Poiché avevamo ancora del tempo a disposizione, ho chiesto di provare ad usare questi blocchi per costruire la casetta, quindi il quadrato sormontato dal triangolo equilatero. Ho suggerito subito di scrivere un numero di passi non elevato, per evitare che la figura uscisse dallo schermo.

Attività 8 - Gruppo Scratch: Scopriamo Logo; comandi principali

Terminate le lezioni con Scratch il gruppo ha iniziato a conoscere il programma Logo. Allo stesso modo del gruppo precedente, ho fatto cercare ai bambini il programma Libre Office Writer nell'apposito spazio di ricerca di Windows.

Poiché sono stati utilizzati gli stessi computer, la barra degli strumenti di Logo era già visualizzata, così è stato più veloce spiegare le funzionalità del programma. In questa classe ho spiegato che al posto del gattino avremmo conosciuto un nuovo personaggio, la tartaruga, che anche lei sapeva disegnare ma che conosceva solo la lingua inglese. A questa affermazione, al contrario del gruppo Logo, i bambini non sono stati entusiasti e qualcuno si è lamentato di non sapere l'inglese, ma quando si sono messi all'opera, hanno invece dimostrato una buona padronanza della terminologia richiesta. Dopo aver ripassato i comandi che hanno usato su Scratch, ho spiegato i corrispondenti da usare per far muovere la tartaruga, quindi "FORWARD", "LEFT", "RIGHT". Ho spiegato che il valore va inserito dopo il comando e che per le rotazioni non occorre aggiungere il simbolo dei gradi (°), ma solo l'ampiezza dell'angolo in numeri. I bambini hanno iniziato a scrivere i nuovi comandi e a far muovere la tartaruga, poi mi hanno chiesto come "pulire", così ho fatto inserire nel codice anche "CLEARSCREEN" e "HOME". È bastato loro premere su *play* per capire a cosa servivano, dato che avevano già visto le stesse funzionalità su Scratch. La lezione si è conclusa dopo che tutti i bambini sono riusciti a costruire il quadrato.

Attività 9 - Gruppo Logo: Il blocco delle ripetizioni

Il secondo incontro con Scratch del gruppo Logo è servito per conoscere alcune funzionalità, come il blocco delle ripetizioni e il blocco della posizione, che non avevano imparato ad usare nella lezione precedente. Ho spiegato che insieme al blocco pulisci è utile inserire il blocco "vai a x... y..." per fare in modo che il gattino torni alla posizione desiderata ogni volta che si attiva il codice. È stato semplice far notare che posizionando l'indicatore del mouse in un punto dello schermo, le coordinate venivano indicate automaticamente dal programma nel riquadro in basso a sinistra, così potevano scegliere facilmente i valori da inserire nelle ascisse e nelle ordinate. A questo punto ho mostrato loro anche il comando delle ripetizioni, suggerendo di inserire prima il blocco apposito e poi di aggiungere successivamente i blocchi di movimento al loro interno. Abbiamo

visto che per fare il quadrato erano necessarie 4 ripetizioni, ma ho detto loro di sperimentare il comando provando a cambiare il valore sia delle ripetizioni che dell'angolo di rotazione. Sono venuti fuori dei disegni molto creativi e i bambini si sono divertiti a costruire alcune figure geometriche, tra cui il cerchio. In questa lezione ho chiesto di iniziare a pensare ad un semplice disegno, che avrebbero potuto completare nell'ultimo incontro, che poi avremmo stampato.

Attività 10 - Gruppo Scratch: Il comando REPEAT

La seconda lezione con Logo è ripartita da dove eravamo rimasti la volta precedente, ovvero con la costruzione del quadrato, al quale ho chiesto di unire il triangolo equilatero per formare la casetta. A questo punto ho introdotto il comando "REPEAT", spiegando che allo stesso modo del blocco delle ripetizioni, basta aggiungere il valore desiderato, così ho chiesto di modificare il codice della casetta inserendolo al posto giusto. Il problema principale che è emerso è stato il messaggio di errore che appariva sullo schermo di molte coppie, dato dagli spazi mancanti prima e dopo le parentesi quadrate, che i bambini spesso si dimenticavano di inserire.

Attività 11 - Gruppo Logo: Costruiamo figure con Scratch

In questa lezione ho lasciato i bambini completamente liberi di lavorare, aiutandoli, mano a mano che procedevano nel lavoro, a risolvere le problematiche che subentravano. Vedendoli a loro agio nell'uso del programma ho aiutato solo quelle coppie di alunni che erano più in difficoltà, ma tutti alla fine sono riusciti a creare qualcosa di proprio, che fosse una semplice casetta con porta e finestre o un disegno più complesso. Le figure che sono emerse sono state molto carine e creative. Di seguito qualche esempio.

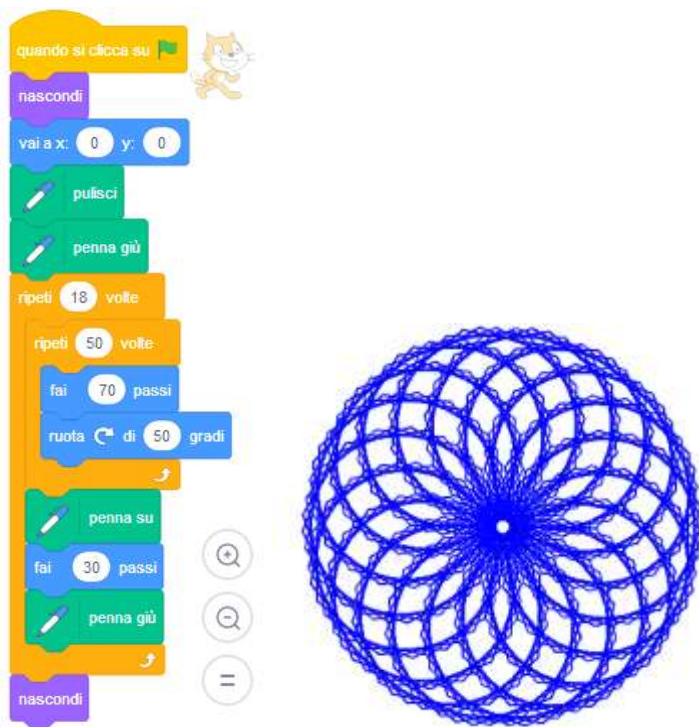


Figura 10 - "Fiore o Sole"

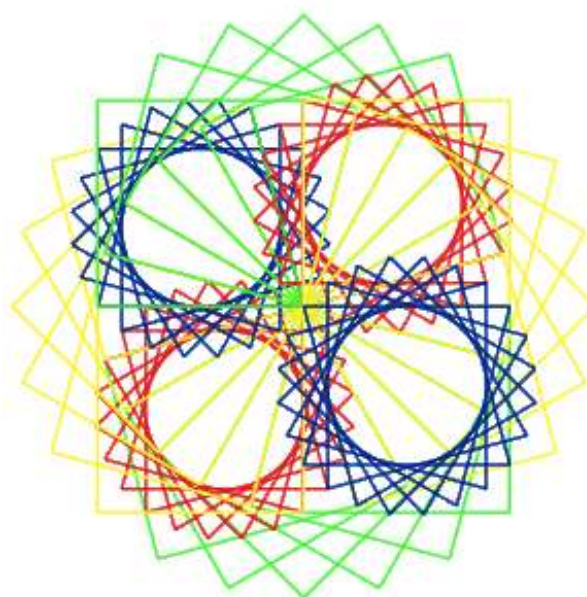
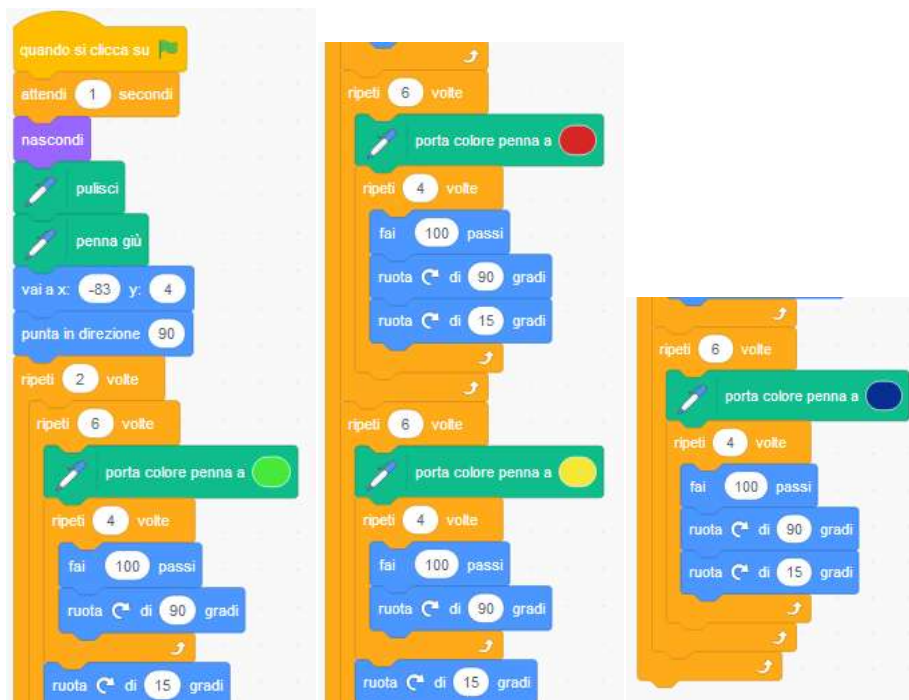


Figura 11 - Immagine astratta

Attività 12 - Gruppo Scratch: Costruiamo figure con Logo

Anche per questo gruppo l'ultima lezione è servita per la produzione di una figura a piacere. I bambini si sono impegnati a lavorare autonomamente, ma a causa delle problematiche presenti nella classe, ho dovuto prestare più attenzione ad alcune coppie, vigilando tra i banchi. Tutti sono riusciti a costruire una immagine, che poi abbiamo salvato su *pendrive* per stamparla. Solamente un gruppo ha avuto un problema con il codice creato perché, senza volerlo, ha modificato un comando interno, che ha cambiato completamente la struttura della figura. Purtroppo non sono riusciti a trovare l'errore per correggerlo, così hanno modificato leggermente il codice dando alla figura un nome "diverso".

```
CLEARSCREEN  
HOME  
FORWARD 90  
LEFT 60  
FORWARD 90  
LEFT 120  
FORWARD 90  
LEFT 60  
FORWARD 90  
LEFT 30  
FORWARD 90  
LEFT 90  
FORWARD 90  
RIGHT 90  
FORWARD 120  
RIGHT 90  
FORWARD 90  
RIGHT 90  
FORWARD 120
```

```
penup  
right 180  
forward 80  
left 90  
forward 50  
pendown  
CIRCLE 10  
penup  
right 180  
forward 50  
pendown  
FORWARD 60  
LEFT 90  
FORWARD 30
```

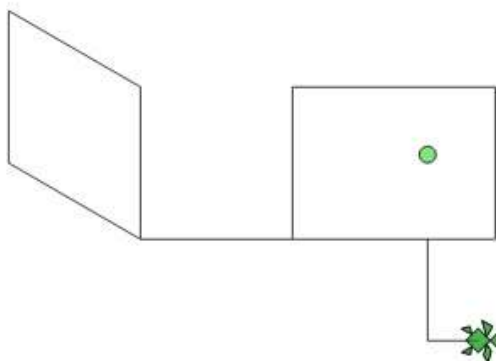


Figura 12 - Pulcino zoppo

```

Showturtle
home
clearscreen
forward 100
left 90
forward 100
left 90
forward 100
left 90
forward 100
left 90
forward 100
left 90
forward 100
left 30
forward 100
left 120
forward 100
left 30
forward 100
left 90
forward 60
left 90
forward 30

```

```

left 90
forward 20
left 90
forward 30
penup
left 180
forward 60
pendown
repeat 4 [ forward 20 left 90 ]
forward 20
right 90
penup
forward 20
pendown
forward 20
right 90
forward 20
right 90
forward 20
right 90
forward 20
right 90
forward 20
hideturtle

```

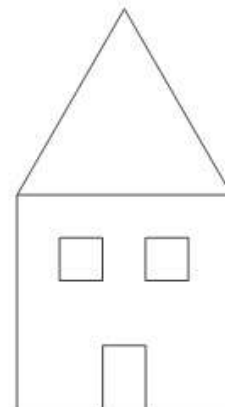


Figura 13 - Casa con porta e finestre

```

Home
clearscreen
pendown
forward 100
left 90
forward 50
right 90
forward 30
right 90
forward 30
left 90
forward 60
left 90
forward 90
left 90
forward 60
left 90
forward 30

```

```

right 90
forward 30
right 90
forward 50
left 90
forward 100
left 30
left 60
forward 30
left 90
forward 80
penup
right 90 forward 70
right 90
pendown
forward 80
left 90
forward 30

```

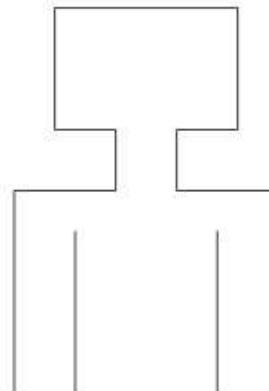


Figura 14 - "Omino"

3.6 Test di geometria

Nel corso dell'attività, sono stati svolti tre test di geometria. Le domande, preparate precedentemente da alcune colleghe, in collaborazione con il Professor Formiconi, sono state ideate specificamente, riprendendo alcuni degli *items* utilizzati nelle ricerche di Clements e Battista (1990) e da Noss (1987), per valutare le conoscenze degli alunni sugli argomenti riguardanti gli angoli, prima, durante e dopo aver sperimentato l'uso dei linguaggi di programmazione proposti. Il Test 1 (pretest) è stato somministrato all'inizio dell'attività, per avere informazioni sulle conoscenze dei bambini prima delle lezioni che avrebbero affrontato. Il Test 2 (test intermedio) è stato svolto a metà del percorso, dopo che ciascun gruppo ha conosciuto e lavorato con uno dei due software di programmazione, mentre il Test 3 (test finale) in conclusione dell'attività, dopo che entrambi i linguaggi sono stati sperimentati. Ogni test ha un ruolo ben preciso in questo progetto: il test iniziale, che serve per avere informazioni sul livello di conoscenza degli alunni, verrà utilizzato per confrontare i dati ricavati dalle prove successive; il test 2 è importante in quanto permette di confrontare i risultati dell'uso dei due linguaggi di programmazione, ovvero per capire in che misura e con quali differenze i linguaggi testuale e visuale influiscono nell'apprendimento; il Test 3 è utile invece per valutare l'efficacia della programmazione nell'acquisizione di concetti geometrici in generale, essendo svolto alla fine di tutta l'attività laboratoriale. I test sono strutturati tutti allo stesso modo, così che ogni domanda e la corrispettiva negli altri test, riguardi lo stesso argomento. Ogni esercizio differisce dal corrispettivo del test precedente perché sono state modificate le immagini o i numeri. Una differenza leggermente maggiore è riscontrabile solo nel test 3, necessaria per fare in modo che i bambini si impegnassero a leggere bene ogni quesito e a cercarne la soluzione. Nel rispondere ad alcune domande, infatti, sono stati fatti degli errori dovuti alla poca attenzione nel leggere i problemi, ritenuti uguali a quelli del test precedente (per esempio, alla richiesta di calcolare quante rotazioni servono ad un robot che ruota di 90° per puntare nella direzione opposta, in molti hanno dato come soluzione il numero di rotazioni necessarie per puntare nella direzione iniziale, richiesta nel test precedente). Questo problema si è presentato durante questa attività come

nelle attività che hanno svolto in precedenza le mie colleghe. Alcune domande dei test sono state formulate specificamente per valutare la presenza e la permanenza o il superamento di misconcezioni, riguardanti per esempio l'angolo retto, rappresentato in posizioni diverse da quella standard a cui sono abituati i bambini.

3.6.1 Descrizione del campione

Tutti e tre i test sono stati svolti in classe. I bambini hanno avuto a disposizione un'ora di tempo, necessaria perché tutti riuscissero a completare i quesiti, anche se in molti hanno consegnato il test in anticipo. Poiché correggendo il primo test ho notato che molte risposte erano state date allo stesso modo, ho deciso di separare i banchi nei successivi, per fare in modo che ognuno ragionasse autonomamente sulle domande. Durante il primo test erano presenti 22 bambini del gruppo Logo e tutti i bambini del gruppo Scratch; il test 2 è stato svolto da 21 bambini del gruppo Logo e 17 del gruppo Scratch, mentre durante il test finale c'erano 21 bambini del gruppo Logo e 20 del gruppo Scratch. Per quanto riguarda gli alunni con bisogni educativi speciali, si sono presentati alcuni problemi di esecuzione e comprensione: uno due bambini stranieri presenti nel gruppo Logo (classe 5B) essendo arrivato in Italia dalla Germania solamente da un mese, non possedeva ancora una conoscenza della lingua tale da permettergli la comprensione di alcuni quesiti, troppo complessi per lui dal punto di vista lessicale, tanto che ha dovuto chiedere aiuto molto spesso sia a me che all'insegnante di matematica; ha mostrato comunque interesse e impegno e di possedere delle solide conoscenze sugli argomenti trattati. Il bambino è stato presente solamente durante il Test 1. Per quanto riguarda l'altra bambina straniera, proveniente dalla Repubblica Dominicana, non ci sono stati problemi di comprensione, essendo già ad un buon livello lessicale/grammaticale e di conoscenze matematiche. L'ultimo caso su cui è importante porre l'attenzione è quello del bambino con ritardo cognitivo, che nonostante fosse perfettamente in grado di svolgere la maggior parte degli esercizi, ha completato quasi del tutto a caso i due test a cui era presente.

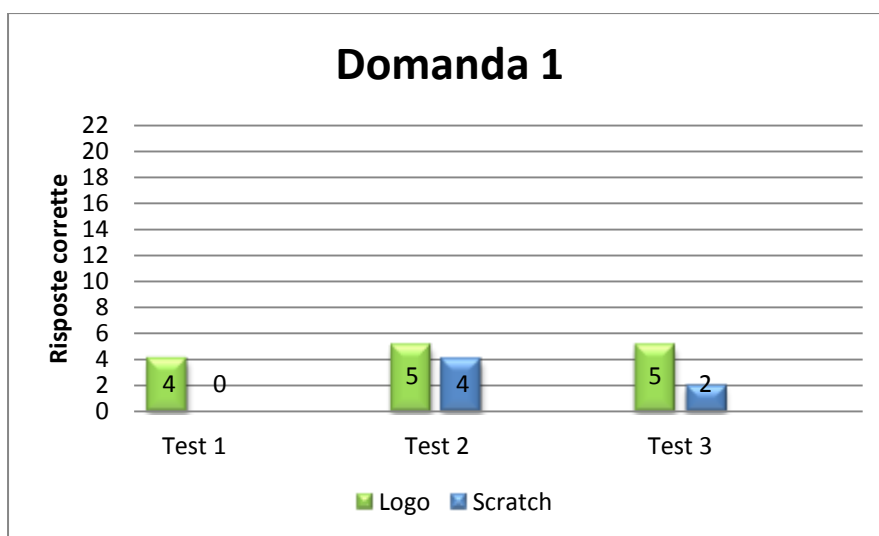
3.7 Risultati dei test

I risultati dei test sono stati analizzati tenendo conto del numero di risposte corrette date nei tre test, e divise per gruppo, per vedere se ci sono differenze rilevanti nell'utilizzo di un linguaggio di programmazione rispetto all'altro. Si vuole evidenziare quindi l'andamento (eventuale aumento o diminuzione di risposte corrette) di ognuna delle due classi dopo aver lavorato prima con un solo programma, poi con entrambi.

Sono riportate di seguito le domande dei tre test e i rispettivi risultati rappresentati in grafici a istogramma dove sono presenti le risposte corrette divise per il gruppo Logo e il gruppo Scratch.

Domanda 1

Test 1, 2, 3: Cos'è un angolo? Disegnane uno.



Le risposte alla domanda "Cos'è un angolo?" sono state le seguenti:

Test 1: *Un angolo è un pezzo di un poligono che può variare dalla sua forma; è il punto di incrocio tra due lati; sono due segmenti che si uniscono a forma di punta e possono essere di diversi tipi; gli angoli sono le cose che formano le forme*

geometriche; un angolo sono due linee che si incrociano formando una cosa un po' appuntita; l'angolo è l'ampiezza di una forma geometrica; è l'ampiezza formata da due linee rette che si incontrano in un punto; un angolo è uno spazio racchiuso da due linee semirette; un angolo è formato da due linee rette che si incontrano nel vertice, l'ampiezza si misura in gradi; un angolo è uno spazio racchiuso da due semirette che si incontrano in un vertice; è uno spazio racchiuso da due linee rette; è un'ampiezza che si crea quando due linee si incrociano; è uno spazio in cui due linee si incontrano; è l'ampiezza di due lati che si incrociano; è l'incrocio di due rette che si incontrano nel vertice e formano l'ampiezza; sono due linee rette che partono dal vertice e dentro ci si forma uno spazio di nome ampiezza, un angolo è un insieme di linee rette che sono unite da un pallino chiamato vertice; è uno spazio di una figura; un angolo è uno spigolo di una figura; un angolo sono due linee che partono da un unico punto; è una figura geometrica che ha 3 lati; è una parte della figura; è composto da un vertice e un'ampiezza che serve a rendere la figura un poligono; sono due semirette che hanno un vertice e un'ampiezza; sono due lati uniti da un vertice; un angolo è l'ampiezza di una figura; l'angolo è la parte interna e stretta di una figura geometrica; è un punto di incontro tra due lati che parte da un vertice e che può essere giro, piatto, retto, acuto e concavo grazie alla sua ampiezza; un angolo è un vertice da cui partono delle linee; un angolo è una figura geometrica che si misura con i gradi e gli angoli possono essere di vari tipi; è il vertice di una figura; un angolo è ciò che si trova vicino al vertice ma non si vede, indica l'ampiezza di una figura; è un incrocio di lati che forma un'ampiezza; è una figura che spesso è appuntita.

Test 2: *Un angolo sono due segmenti che si incontrano in un punto chiamato vertice; è l'ampiezza; un angolo è un'ampiezza che parte dal vertice che è un punto dove inizia l'angolo e poi da quel punto partono due segmenti; un angolo è un'ampiezza formata da due rette che si incontrano; è una parte di una forma che si misura in gradi; è il punto di collegamento da delle linee rette; un angolo sono due linee che incontrandosi formano un angolo; un angolo è il luogo dove due segmenti si incontrano formando una punta; è uno spazio racchiuso tra due semirette; un angolo è l'ampiezza racchiusa da due segmenti che partono da un*

punto chiamato vertice; l'angolo è il punto d'incontro di due segmenti; è l'incontro di due linee rette; un angolo è quando due lati o più si incrociano; un angolo sono due lati uniti che possono avere diverse misure; l'angolo è uno spazio racchiuso da due segmenti uniti dal vertice; un angolo sono due lati che si uniscono in un punto che si chiama vertice e possono essere di vari gradi; è un incidente di due rette che si incontra nel vertice formando un'ampiezza; è uno spazio racchiuso da due linee semirette unite dal vertice; sono due linee che partono da un punto di nome vertice; è una figura composta da due segmenti uniti da un vertice con cui misuriamo l'ampiezza; è una parte di un poligono; è un incontro tra due lati chiamato vertice, può avere diverse forme: giro, piatto, concavo, ottuso...; un angolo è una piccola ampiezza dentro due lati e può essere acuto, retto, ottuso...; è formato da due linee che si incontrano in un vertice e ha un'ampiezza; sono due segmenti che si incrociano e formano il vertice e un'ampiezza cioè lo spazio tra i due segmenti; è un incrocio di linee che forma un'ampiezza; è il punto più stretto della figura; è uno spazio con una punta; è l'ampiezza di una figura; è formato da due linee che a un certo punto si incontrano formando un'ampiezza; un angolo parte dal vertice e da lì spuntano due segmenti che formano un'ampiezza; è l'ampiezza che si trova tra il vertice e i due lati.

Test 3: Un angolo sono due segmenti che si incontrano in un punto chiamato vertice; è uno spazio racchiuso da due linee; è un'ampiezza maggiore o minore o uguale a 90° ; sono segmenti che si incontrano nello stesso punto formando un vertice; è lo spazio tra due segmenti uniti l'uno all'altro; è l'ampiezza; sono due linee dritte che partono da un punto; è un'ampiezza formata da due rette che partono dallo stesso vertice; un angolo è una linea incrociata con un'altra; l'angolo è il punto di incontro fra due linee; è uno spazio racchiuso da due linee semirette; è l'incontro tra due segmenti che creano un'ampiezza fra di loro; è formato da due segmenti uniti, può essere misurato in gradi; un angolo è lo spazio racchiuso da due semirette che si incontrano in un punto chiamato vertice; è l'ampiezza formata da due rette che creano un incidente; è una figura geometrica che ha tre lati che qualche volta sono diversi tra di loro; un angolo è una cosa che ha un'ampiezza e minimo due lati; è una parte della figura che serve

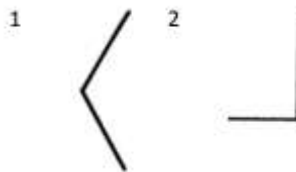
per chiuderla; sono due segmenti che si vanno a incontrare in un punto chiamato vertice e ha un'ampiezza; sono due segmenti che si incontrano in un punto cioè partono da un punto chiamato vertice; sono due semirette che si incontrano in un punto chiamato vertice e formano un'ampiezza; è un'ampiezza creata da due linee che si incontrano; è un'ampiezza che sta in mezzo a due linee; è la parte piegata della figura; è formato dal vertice dall'ampiezza ecc...; è il punto di incontro tra due lati e la sua ampiezza può essere di molti tipi: 90°, 45°, ecc...; sono due segmenti che si uniscono e che formano uno spazio chiamato ampiezza; è una superficie che si trova tra due lati.

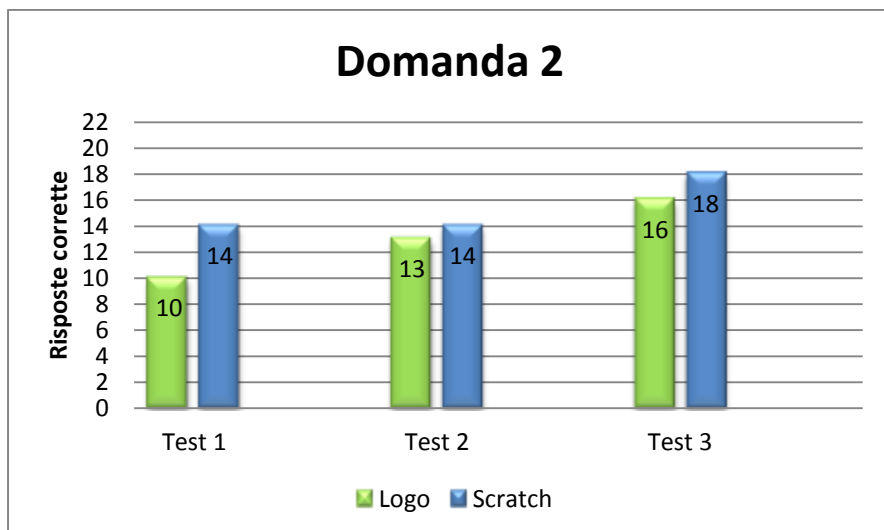
Nonostante tutti i bambini abbiano saputo disegnare l'angolo (a parte l'eccezione di un bambino che ha disegnato un triangolo, nel Test 3), è evidente che pochissimi hanno chiaro il concetto dello spazio contenuto tra le due semirette che hanno origine dallo stesso punto. Dalle risposte si può notare che in generale l'angolo viene identificato nelle due semirette o nel vertice, raramente nell'ampiezza.

Domanda 2

Test 1, 2: Puoi disegnare un angolo maggiore rispetto a quello che hai disegnato prima? Perché è maggiore?

Test 3: Quale tra i due angoli è il maggiore? Perché?





Nei primi due test le risposte non sono state del tutto corrette in quanto molti bambini non hanno saputo dare una spiegazione giusta del perché un angolo si definisca maggiore di un altro, indicando come dato da considerare la lunghezza delle due semirette. Si può notare un miglioramento progressivo del gruppo Logo, dopo aver sperimentato entrambi i programmi e un netto aumento di risposte positive per il gruppo Scratch nel terzo test, nel quale quasi tutti i bambini hanno riconosciuto l'angolo più ampio. Per rispondere alla domanda ho notato che diversi alunni hanno utilizzato il goniometro per specificare l'ampiezza degli angoli in gradi, nonostante avessi spiegato che non ce ne sarebbe stato bisogno.

Domanda 3

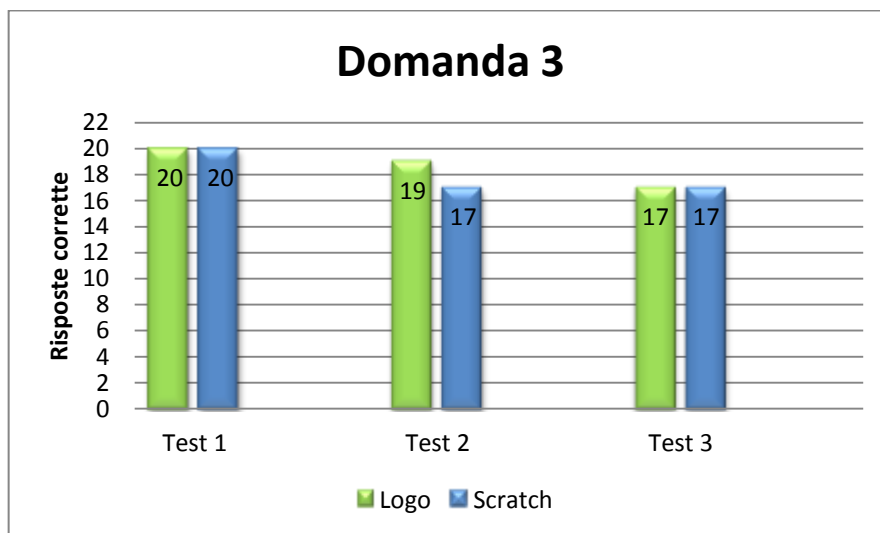
Test 1: Cerchia tra queste figure quelle che rappresentano angoli.



Test 2: Cerchia tra queste figure quelle che rappresentano angoli.



Test 3: In ogni coppia di figure, individua l'angolo e cerchiolo.



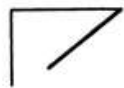
Cerchiando i triangoli, molti bambini hanno riconosciuto nei segmenti l'angolo piatto, disegnando l'arco sulla figura. In molti hanno segnato l'angolo piatto anche nel terzo test, nonostante la domanda richiedesse di riconoscere l'angolo in ogni coppia di figure. Dopo i risultati positivi del primo e del secondo test (tenendo di

conto che durante il Test 2 erano presenti solamente 17 alunni del gruppo Scratch) c'è stato un calo delle risposte corrette per entrambi i gruppi. Alcune risposte sbagliate sono dovute al fatto che qualche bambino ha cerchiato il semicerchio.

Domanda 4

Quanti angoli hanno queste figure? scrivi il numero di angoli sotto ad ogni figura.

Test 1:

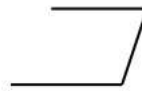


..... Angoli

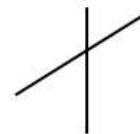


..... Angoli

Test 2:

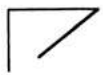


..... Angoli

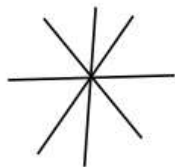


..... Angoli

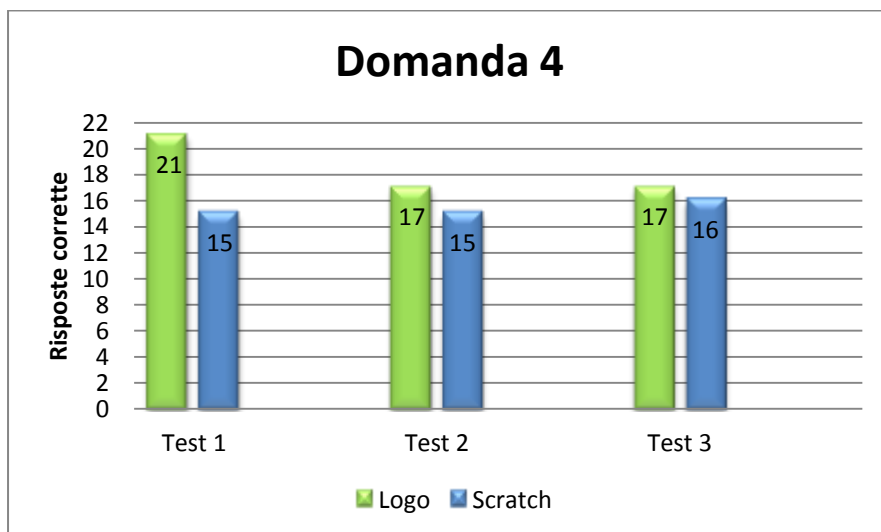
Test 3:



..... Angoli



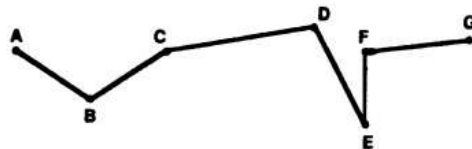
..... Angoli



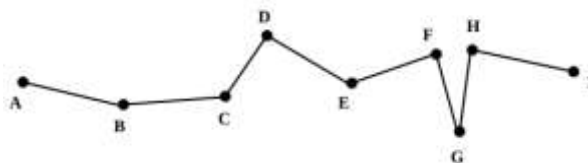
La domanda 4 ha presentato un peggioramento per il gruppo Logo, che nei test 2 e 3 ha avuto un calo di risposte corrette. Il gruppo Scratch invece si è mantenuto più o meno sullo stesso numero. Le risposte dei bambini che hanno contato anche gli angoli esterni nella prima figura di ogni test sono ritenute corrette, in quanto tutti gli angoli sono stati evidenziati con l'archetto corrispondente, mentre sono state valutate sbagliate le risposte dei bambini che hanno evidenziato gli angoli piatti in alcuni segmenti, poiché hanno contato un numero errato di angoli.

Domanda 5

Test 1: Stai camminando su questo sentiero. Inizi dal punto A e finisci nel punto G. In quale punto devi ruotare di più? Indicalo con una freccia di colore blu In quale punto devi ruotare di meno? Indicalo con una freccia di colore rosso



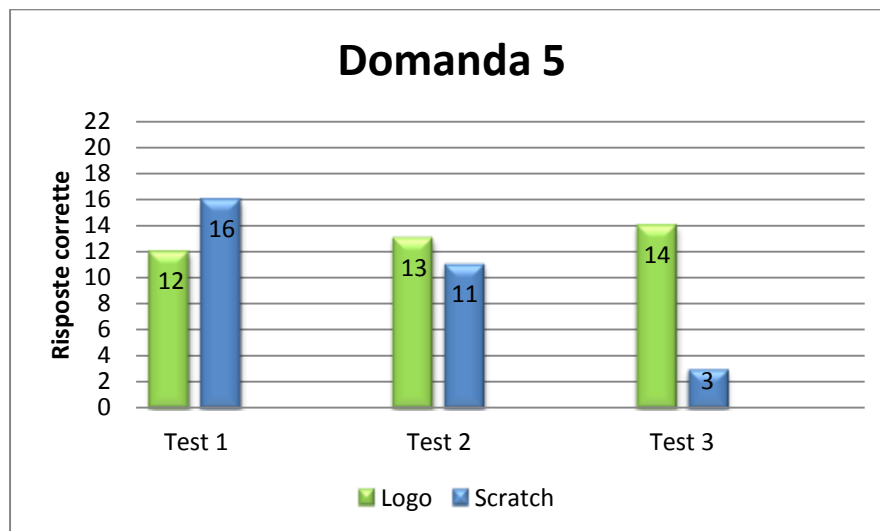
Test 2: Stai camminando su questo sentiero. Inizi dal punto A e finisci nel punto I. In quale punto devi ruotare di più? Indicalo con una freccia di colore blu In quale punto devi ruotare di meno? Indicalo con una freccia di colore rosso



Test 3: Adesso prova tu a disegnare un sentiero che inizi nel punto A e finisca nel punto G. Ad ogni cambio di direzione scrivi una lettera. Indica con una freccia di colore blu il punto in cui devi ruotare di più e con una freccia di colore rosso il punto in cui devi ruotare di meno.

A •

• G



Questa domanda ha messo in difficoltà quasi tutti i bambini. Per il gruppo Logo si può osservare un aumento di due risposte corrette dal primo all'ultimo test, questo significa che è stato compreso il testo del problema dalla maggior parte degli alunni, che hanno saputo disegnarlo autonomamente nel Test 3. Per il gruppo Scratch invece c'è stato un grande peggioramento anche se nel primo test le risposte corrette sono state molte. Osservando i tre test, si possono osservare una moltitudine di risposte diverse: alcuni bambini hanno riconosciuto solamente il punto in cui ruotare di più, molti invece hanno colorato un segmento di blu e un segmento di rosso. Nel terzo test è stato interessante vedere l'impegno che gli

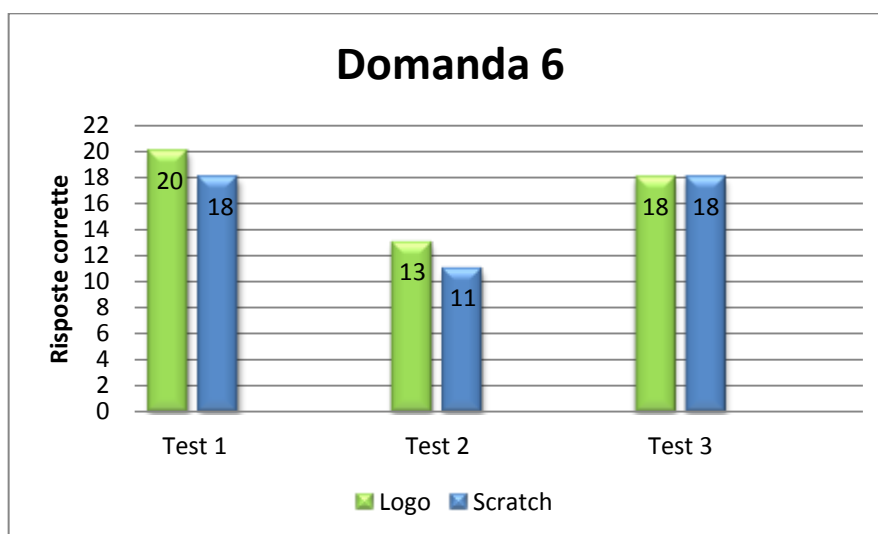
alunni hanno messo nel disegnare il percorso, anche in questo caso però molti hanno individuato correttamente solo il punto in cui ruotare di più. La maggior parte dei bambini del gruppo Scratch non ha risposto a questa domanda.

Domanda 6

Test 1: Un robot ruota di 90 gradi ogni volta che ruota. Quante rotazioni deve fare il robot per puntare nella stessa direzione? Perché?

Test 2: Un robot ruota di 60 gradi ogni volta. Quante rotazioni deve fare prima di tornane nella stessa posizione iniziale? Perché?

Test 3: Un robot ruota di 120 gradi ogni volta che ruota. Quante rotazioni deve fare il robot per puntare nella stessa direzione? Perché?



Osservando il grafico si può notare che nel secondo test, per entrambi i gruppi c'è stato un peggioramento. Questo è dato dal fatto che molti bambini hanno risposto alla domanda del Test 2 come se fosse la stessa del primo test, quindi scrivendo "per fare un giro completo deve fare 4 rotazioni". Nel terzo test le risposte sono state positive in entrambi i gruppi, tornando vicino ai numeri del Test 1. È curioso

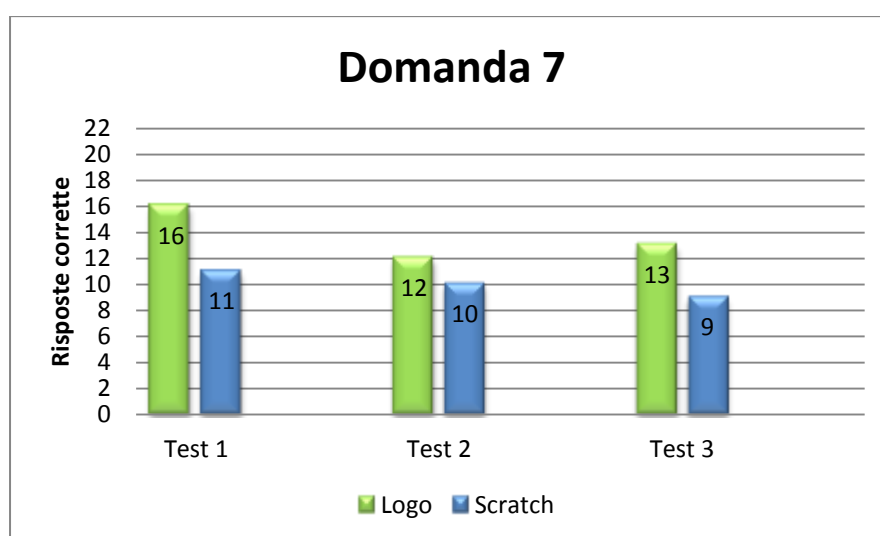
osservare che le risposte corrette sono state più numerose nei Test 1 e 3, nei quali le domande sono formulate nello stesso modo, anche se cambia il valore dei dati.

Domanda 7

Test 1: Un robot ruota di 30 gradi ogni volta. Quante rotazioni deve fare prima di puntare la stessa direzione? Perché?

Test 2: Un robot ruota di 90 gradi ogni volta che ruota. Quante rotazioni deve fare il robot per puntare nella direzione opposta? Perché?

Test 3: Un robot ruota di 45 gradi ogni volta che ruota. Quante rotazioni deve fare il robot per puntare nella direzione opposta?



La domanda 7 ha avuto molte risposte sbagliate e entrambi i gruppi presentano un peggioramento, seppur non elevato. Per quanto riguarda il primo test, è chiaro che sono stati fatti errori di calcolo, in quanto ci sono state risposte come *"Il robot deve ruotare 11 volte per fare il giro completo"* oppure *"Deve ruotare 8 volte"*. Nei Test 2 e 3 invece è stato commesso lo stesso errore da moltissimi bambini, che non hanno letto attentamente il testo. Infatti, quasi tutte le risposte scorrette

sono dovute al fatto che sono state calcolate le rotazioni necessarie per puntare nella direzione iniziale e non nella direzione opposta.

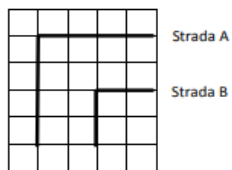
Domanda 8

Nella città chiamata Quadrata tutte le strade sono disposte su uno schema quadrato. Ci sono solo due strade principali, che sulla mappa sono rappresentate da linee più scure.

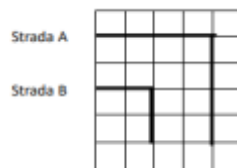
Seleziona la risposta che pensi sia corretta:

- La strada A curva di più della strada B
- La strada B curva di più della strada A
- La strada A e la strada B curvano nello stesso modo
- Non puoi dirlo

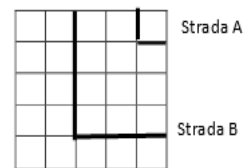
Test 1:

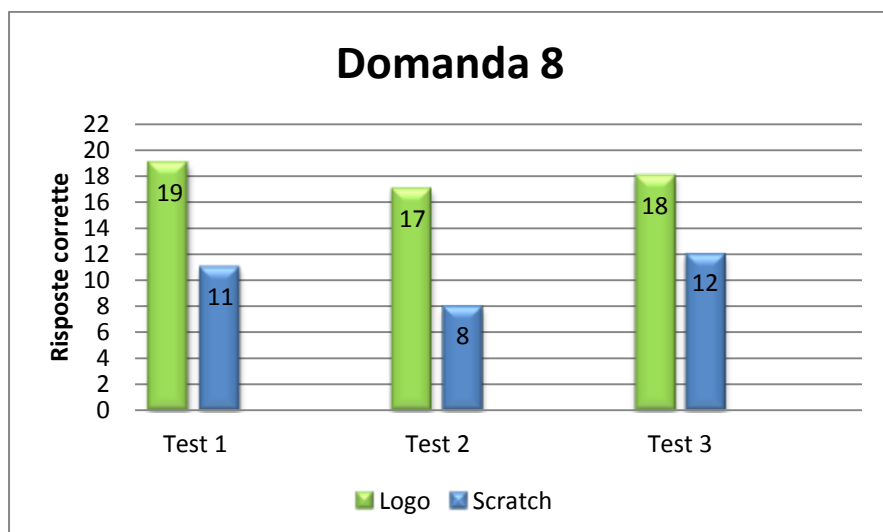


Test 2:



Test 3:





Il grafico della domanda numero 8 ci può dare delle indicazioni per quanto riguarda le misconcezioni che si creano nei bambini riguardo agli angoli retti. Per il gruppo Logo non ci sono stati grossi problemi, anche se nel secondo test ci sono state due risposte corrette in meno rispetto al primo. Nel gruppo Scratch invece è chiaro che molti bambini non hanno compreso il concetto e che per rispondere hanno considerato la lunghezza dei segmenti che formano l'angolo (su molti test si possono osservare i numeri disegnati vicino alle strade per contare i quadratini e misurarne la lunghezza). Nei Test 1 e 2 infatti chi ha risposto erroneamente ha indicato la frase "*La strada A curva di più della strada B*". Nel terzo test, nonostante ci siano state molte risposte scorrette, c'è stato un leggero aumento per questo gruppo, dopo aver sperimentato il programma LibreLogo.

Domanda 9

Nella città di Quadrata ci sono due nuove strade C e D

Seleziona la risposta che pensi sia corretta:

La strada C curva più della strada D

La strada D curva più della strada C

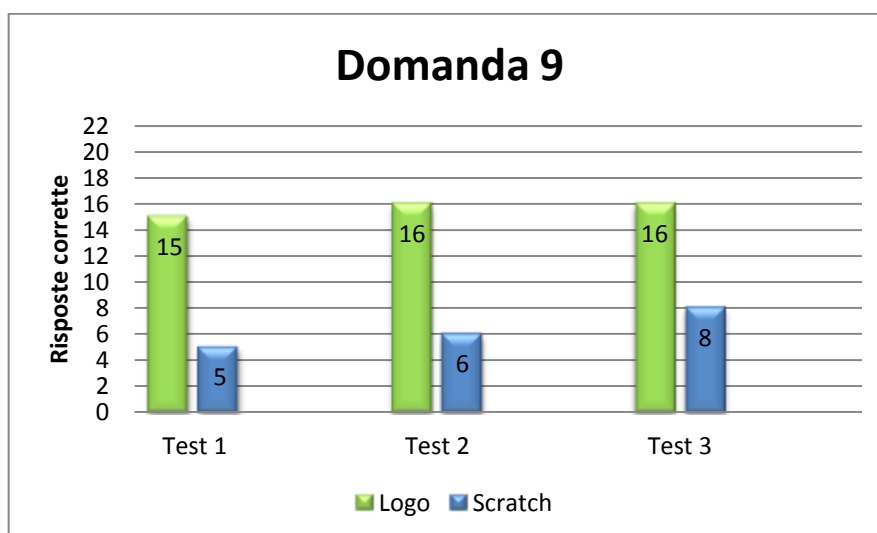
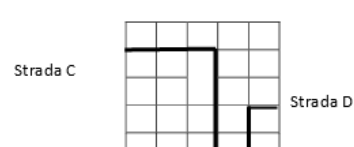
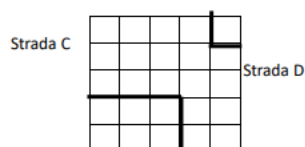
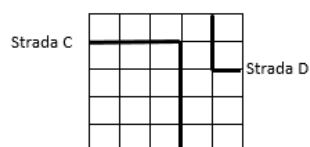
Le strade C e D curvano nello stesso modo

Non puoi dirlo

Test 1:

Test 2:

Test 3:



Il grafico della domanda 9 mostra un miglioramento per entrambi i gruppi; nonostante il gruppo Scratch presenti risultati peggiori rispetto al gruppo Logo, dal Test 1 al Test 3 ci sono 3 risposte corrette in più. Se confrontiamo questi risultati con quelli della Domanda 8 (in cui i due angoli sono posizionati nello stesso modo) possiamo notare che ci sono state difficoltà maggiori nel comprendere il quesito 9, nel quale invece la posizione cambia.

Domanda 10

Seleziona la risposta che pensi sia corretta:

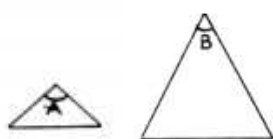
L'angolo A è maggiore dell'angolo B

L'angolo B è maggiore dell'angolo A

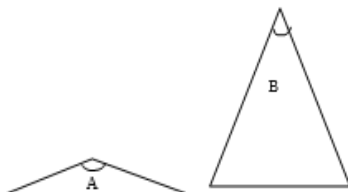
Gli angoli A e B hanno la stessa ampiezza

Non puoi dirlo

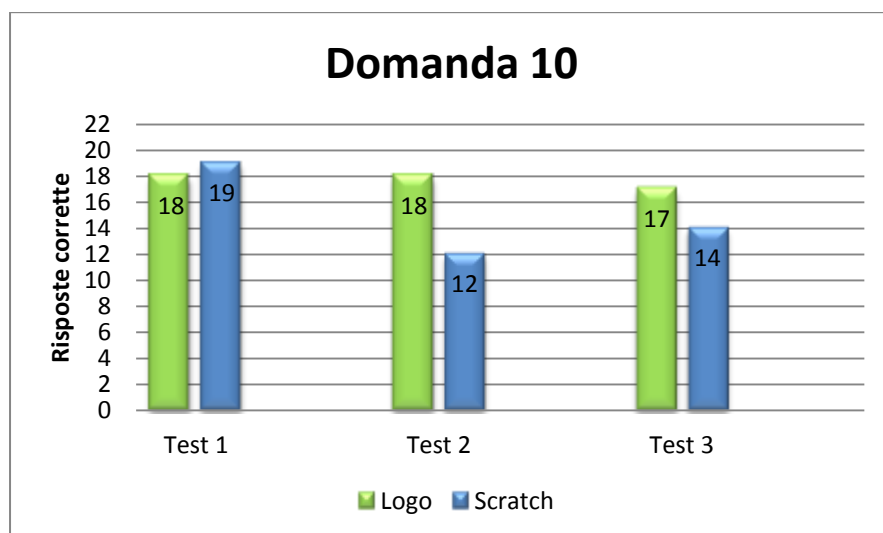
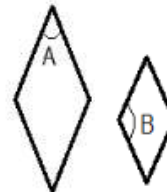
Test 1:



Test 2:



Test 3:



Se per il gruppo Logo non ci sono state grandi variazioni, per il gruppo Scratch si può osservare un notevole calo nel secondo test, per poi migliorare leggermente nel terzo. Come già spiegato per quanto riguarda la Domanda 2, molti bambini considerano maggiore l'angolo compreso tra i due lati più lunghi della figura, infatti nel Test 2 hanno indicato come angolo maggiore l'angolo B, mentre nel Test 3 l'angolo A.

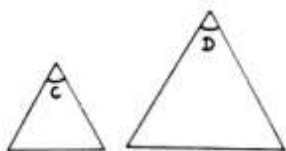
Domanda 11

In ognuno di questi triangoli tutti i lati hanno la stessa lunghezza.

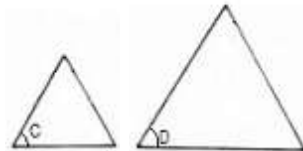
Seleziona la risposta che pensi sia corretta³⁷:

- L'angolo C è maggiore dell'angolo D
- L'angolo D è maggiore dell'angolo C
- Gli angoli C e D hanno la stessa ampiezza
- Non puoi dirlo

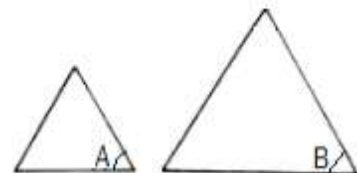
Test 1:



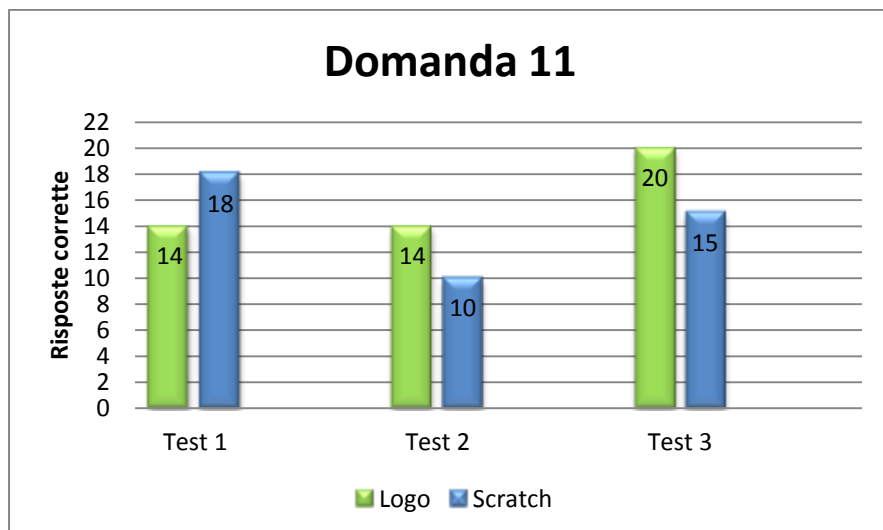
Test 2:



Test 3:



³⁷ Le risposte a scelta multipla sono le stesse in tutti e tre i test, solamente nell'ultimo gli angoli indicati sono l'angolo A e l'angolo B, come mostrato dalla figura.



La Domanda 11 presenta un grande miglioramento per il gruppo Logo per quanto riguarda il Test 3, dopo aver sperimentato il secondo programma; era rimasto invariato il numero di risposte corrette dal Test 1 al Test 2. Per quanto riguarda il gruppo Scratch invece le risposte corrette sono diminuite molto nel secondo test, per poi aumentare di nuovo leggermente dopo aver lavorato con il programma Logo. Ci sono stati alcuni test in cui i bambini non hanno risposto a questa domanda, mentre chi ha sbagliato ha indicato il triangolo più grande al posto dell'angolo.

Domanda 12

Osserva gli angoli A e B

Seleziona la risposta che pensi sia corretta:

L'angolo A è maggiore dell'angolo B

L'angolo B è maggiore dell'angolo A

Gli angoli A e B hanno la stessa ampiezza

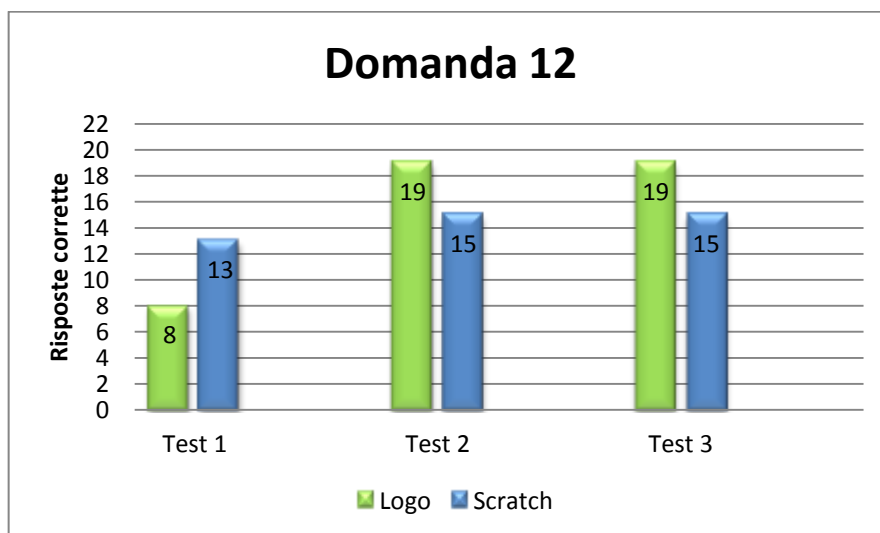
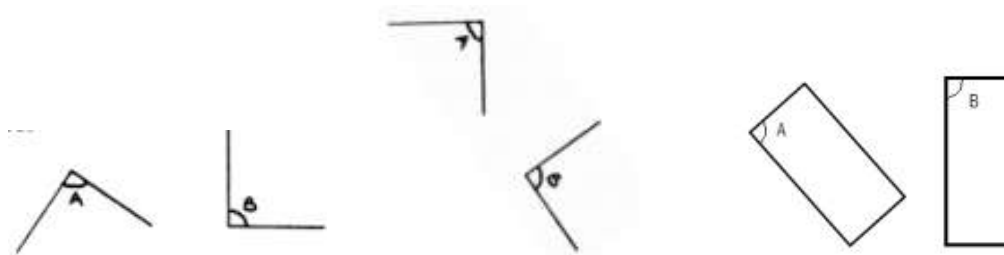
Non puoi dirlo



Test 1:

Test 2:

Test 3:



Per entrambi i gruppi si può osservare un aumento di risposte corrette già dal secondo test, anche se è notevolmente maggiore per il gruppo Logo. Questa domanda è importante per quanto riguarda l'osservazione della presenza di misconcezioni negli alunni rispetto all'angolo retto. Anche per questo quesito, come per alcuni precedenti, alcuni bambini hanno usato il goniometro, che purtroppo li ha fatti sbagliare. Dopo aver spiegato che lo strumento non sarebbe

servito per nessuna domanda in nessuno dei test, hanno iniziato a ragionare da soli sulla risposta.

Domanda 13

Osserva gli angoli

Seleziona la risposta che pensi sia corretta:

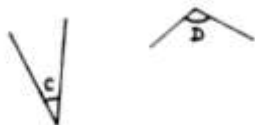
L'angolo C è maggiore dell'angolo D

L'angolo D è maggiore dell'angolo C

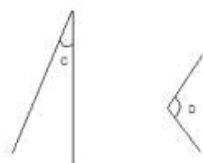
Gli angoli C e D hanno la stessa ampiezza

Non puoi dirlo

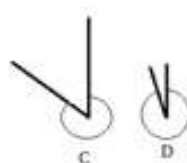
Test 1:

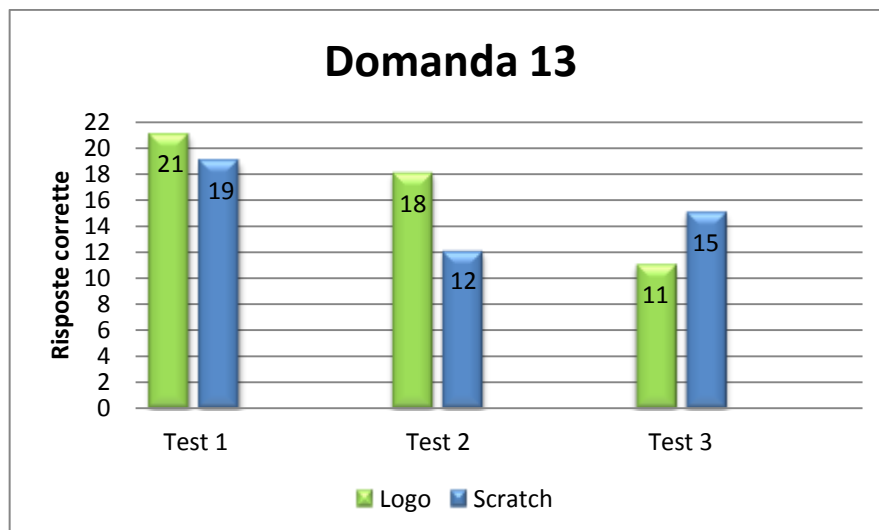


Test 2:



Test 3:





La domanda 13 ha avuto dei buoni risultati nel primo test, ma se per il gruppo Logo c'è stata una variazione di tre risposte corrette nel secondo test, per il gruppo Scratch le risposte errate sono aumentate di sei. È da sottolineare che anche in questo quesito è stato considerato maggiore l'angolo che presenta i segmenti più lunghi. Nel Test 3 c'è stato un miglioramento del gruppo Scratch dopo aver sperimentato Logo. La domanda in questo test è stata interpretata scorrettamente da molti bambini, poiché invece dell'angolo esterno è stato considerato l'angolo interno, quindi è stato indicato l'angolo C.

3.8 Logo e Scratch a confronto: i risultati

Per poter avere un riscontro chiaro sui risultati dell'utilizzo dei linguaggi di programmazione oggetto di questa sperimentazione, è opportuno considerare e confrontare i risultati del Test 1 e del Test 2, poiché permettono di osservare gli effetti diretti dell'utilizzo del linguaggio testuale di LibreLogo e di quello visuale di Scratch. Di seguito è riportata la tabella riassuntiva della variazione di risposte corrette dal primo al secondo test per ciascun gruppo.

N° Domanda	Gruppo Logo			Gruppo Scratch		
	N° Risposte corrette Test 1	N° Risposte corrette Test 2	Variazione Risposte corrette	N° Risposte corrette Test 1	N° Risposte corrette Test 2	Variazione Risposte corrette
1	4	5	+1	0	4	+4
2	10	13	+3	14	14	0
3	20	19	-1	20	17	-3
4	21	17	-4	15	15	0
5	12	13	+1	16	11	-5
6	20	13	-7	18	11	-7
7	16	12	-4	11	10	-1
8	19	17	-2	11	8	-2
9	15	16	+1	5	6	+1
10	18	18	0	19	12	-7
11	14	14	0	18	10	-8
12	8	19	+11	13	15	+2
13	21	18	-3	19	12	-7

Osservando questi risultati si può notare che dal Test 1 al Test 2 per il gruppo Logo le risposte corrette sono aumentate in totale di 17 mentre per il gruppo Scratch solamente di 7. Entrambi i gruppi hanno avuto un aumento di risposte errate, che sono state 21 in più per il gruppo Logo e ben 40 per il gruppo Scratch.

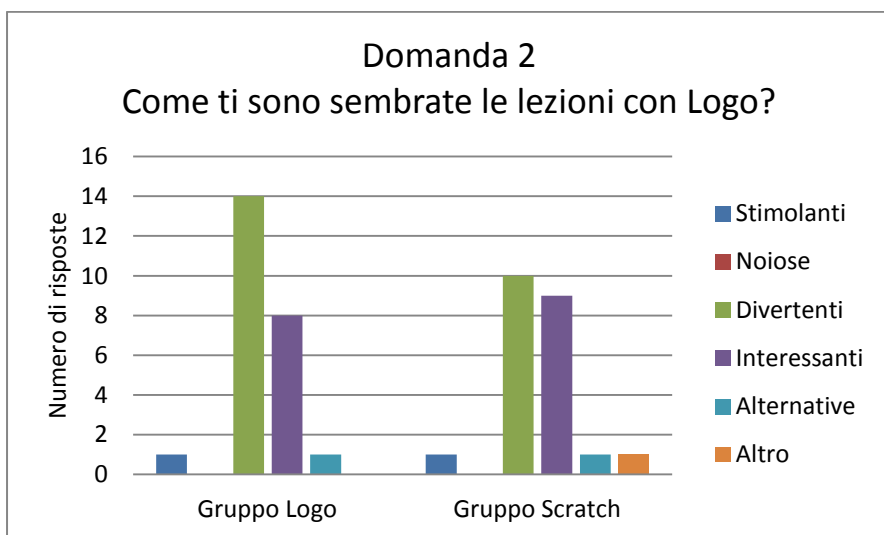
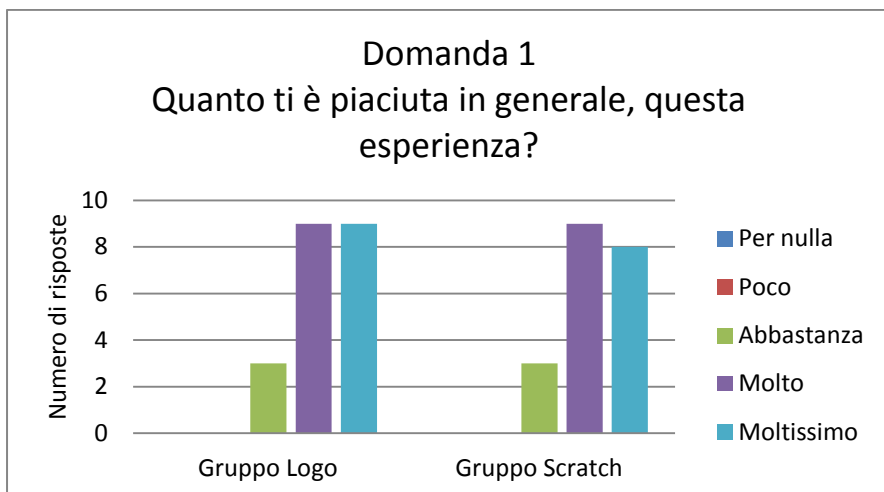
3.9 Presentazione del questionario

Alla fine del progetto è stato somministrato alle classi un questionario, per avere un feedback dai bambini riguardo all'esperienza che hanno svolto e per raccogliere informazioni sull'utilizzo delle tecnologie. Le domande in totale sono

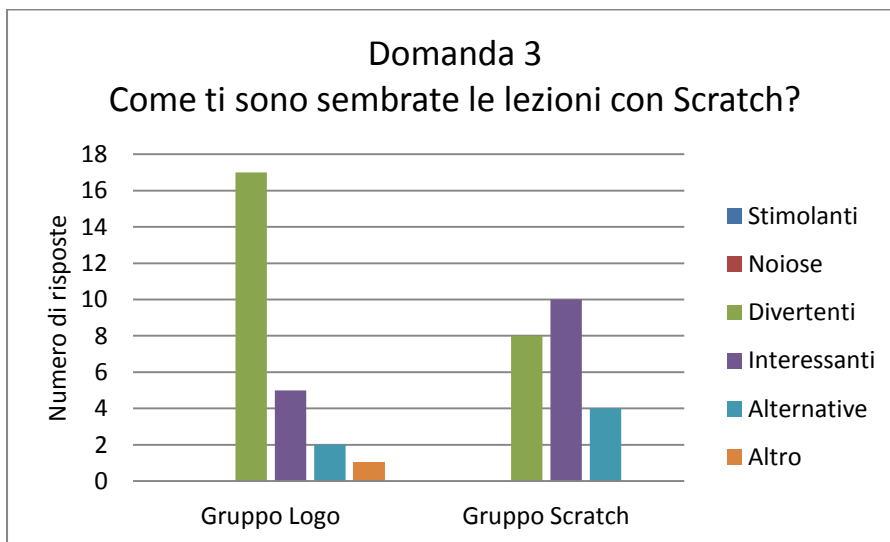
18. Le prime 11 domande si riferiscono all'esperienza svolta, per valutare quanto è stata positiva per i bambini, quali difficoltà hanno incontrato e cosa invece pensano di aver imparato; viene chiesto quale è stato il programma che hanno preferito e per quale motivo e se avrebbero interesse a continuare ad usarli in futuro. Le successive 5 domande riguardano l'utilizzo delle tecnologie a casa, quali vengono utilizzate di più e per quanto tempo. Le ultime domande sono relative al lavoro di coppia, per capire se ci siano stati problemi o difficoltà nel lavorare a piccoli gruppi, o se al contrario sia una modalità vantaggiosa per tutti i bambini. Il questionario è stato svolto nella stessa giornata del Test 3, come ultima attività del progetto; il tempo a disposizione per la compilazione è stato 30 minuti e tutti sono riusciti a completarlo entro il tempo massimo. I bambini del gruppo Logo presenti erano 21, mentre del gruppo Scratch 20, per un totale di 41 bambini; la somma delle risposte supera spesso il numero totale degli alunni, in quanto molte domande a scelta multipla prevedono la scelta di più di una opzione.

3.9.1 Analisi dei risultati del questionario

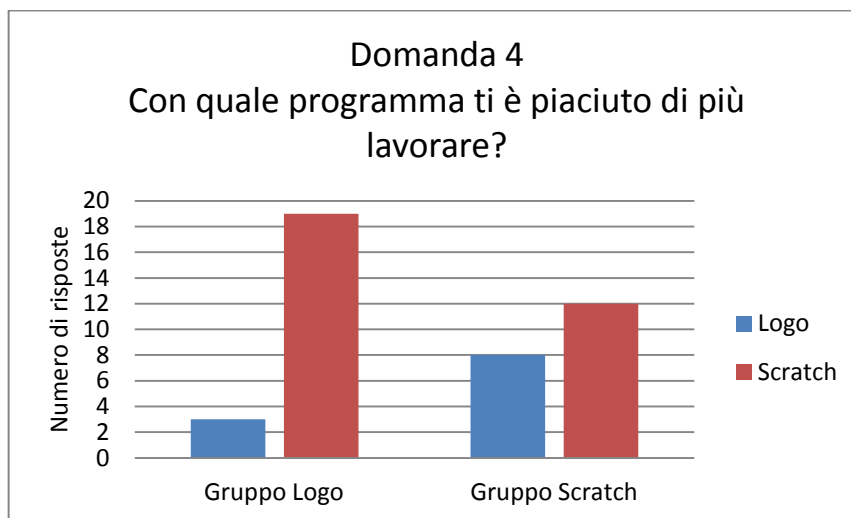
Di seguito sono riportati i risultati dei questionari. Come per i Test, i dati sono divisi per gruppi, per vedere se l'utilizzo dei due programmi possa aver condizionato le scelte dei bambini. Per le domande a risposta aperta sono riportate le parole esatte dei bambini, mentre per quelle a risposta multipla sono stati creati i grafici a istogramma, per poter confrontare in modo immediato i dati.



Un bambino ha risposto con "Altro" specificando che non era presente.



Chi ha risposto "Altro" ha specificato: *Molto divertenti.*



Le risposte alla domanda 4 del gruppo Logo sono in totale 22, poiché un bambino ha segnato entrambi i programmi motivando così la sua scelta: *Perché anche se Logo era in inglese e Scratch in italiano le funzioni sono le stesse cambia solo la lingua.*

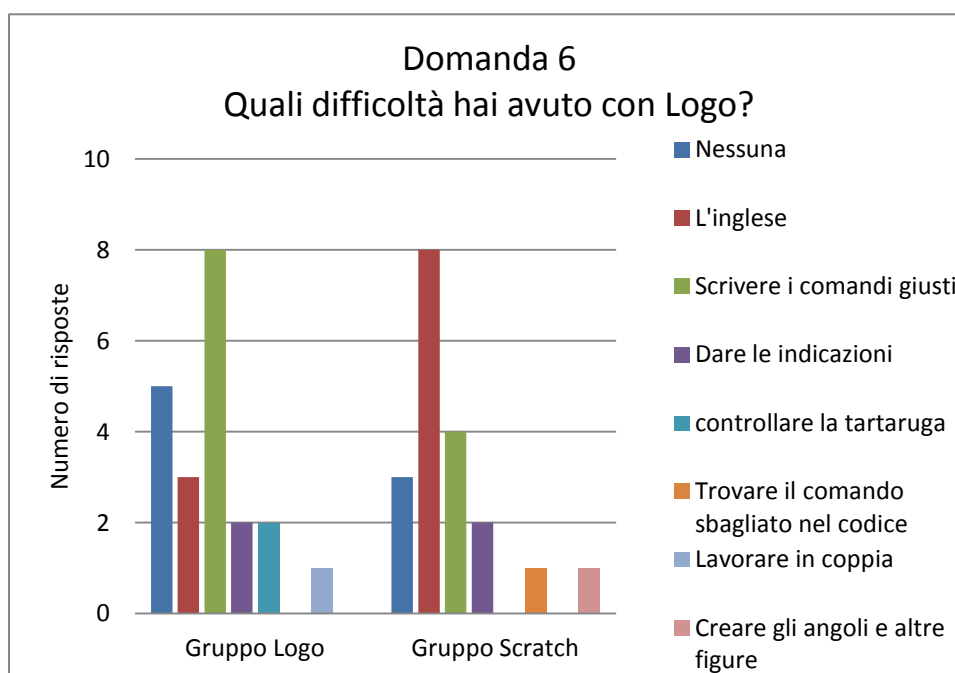
Domanda 5: Perché?

Le risposte di chi ha preferito Logo sono state:

Perché era abbastanza facile e potevi creare cose molto belle; Perché ci siamo stati più tempo e i comandi erano in inglese; Perché su Scratch ho fatto la mia esperienza e non avevo ancora capito come funzionava invece su Logo ho subito capito le meccaniche; Mi piaceva di più scrivere i comandi perché secondo me è più stimolante; A volte Scratch non prendeva i comandi; È stato più facile maneggiarlo fin dall'inizio; In Logo c'erano più cose per fare muovere ma anche per stare fermi; Perché potevamo lavorare di più e anche se ci è voluto più tempo mi sono divertito ugualmente; Era più semplice; Ho imparato un po' l'inglese.

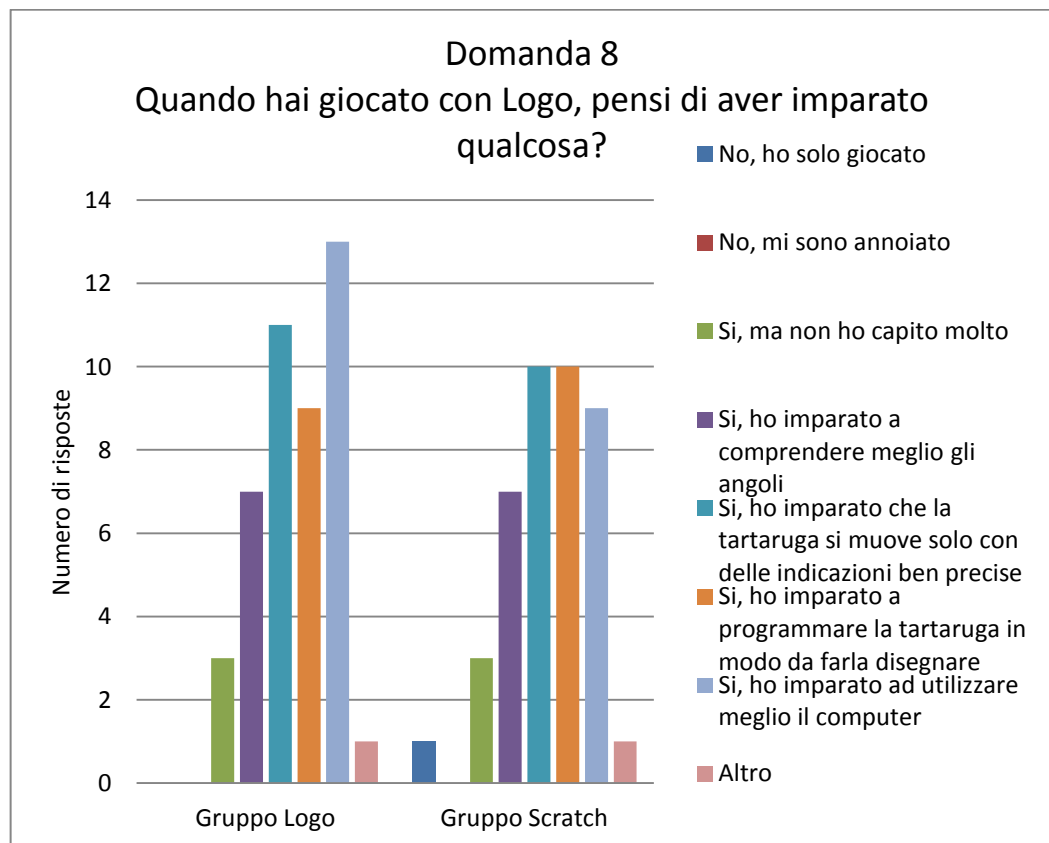
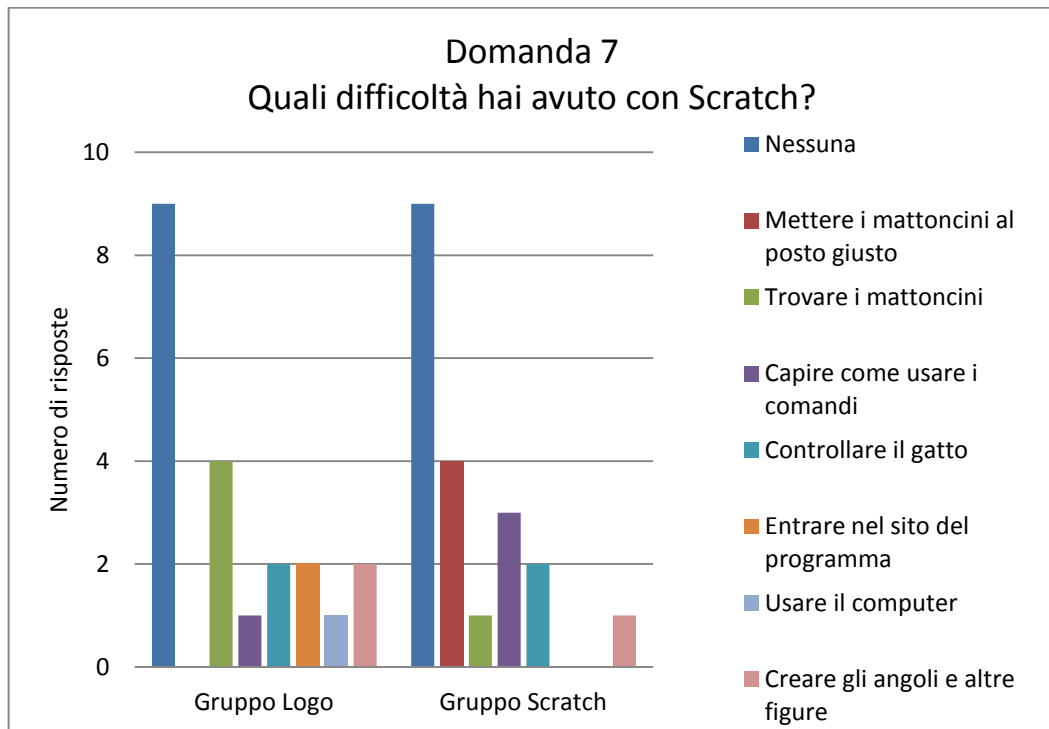
Le risposte di chi ha preferito Scratch sono state:

Perché ha più opzioni a disposizione ed è un programma molto più ampio; Perché non bisognava scrivere le parole e si poteva fare tutto; Fa le stesse cose di Logo però non bisogna scrivere perciò si fa il lavoro più velocemente; Perché mi sembrava più divertente; Perché secondo me era fatto meglio e c'erano più comandi da far eseguire alla figura; Perché ha molti comandi già scritti e sono in italiano quindi ricordi meglio come usarli; Perché con Scratch abbiamo personalizzato il gatto e si poteva far parlare; Perché puoi fare tutto quello che ti pare; Perché avevi già i blocchetti già fatti ed era molto più comodo; Perché avevo imparato dei codici e ho aiutato la mia squadra; Scratch aveva più comandi poi mi piaceva il gattino e poi potevi formare tanti disegni; Perché ci sono le scritte in italiano e sono riuscita a fare tante cose; Perché il robotino aveva molti comandi differenti; Perché mi trovavo meglio con il gatto e ci capivo di più e abbiamo condiviso tante forme; Perché aveva più modi per sperimentare; Perché c'era in tutte le lingue; Ho imparato tantissime cose.



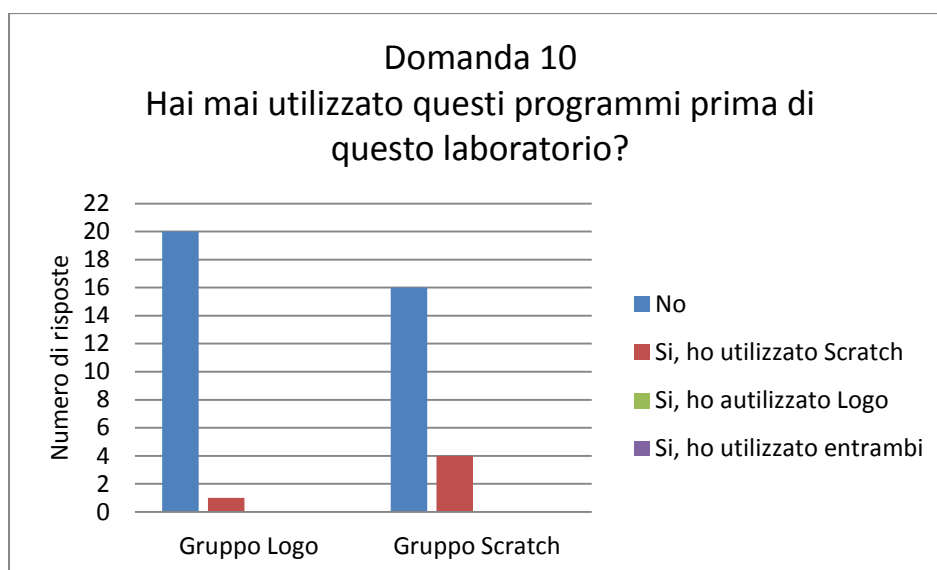
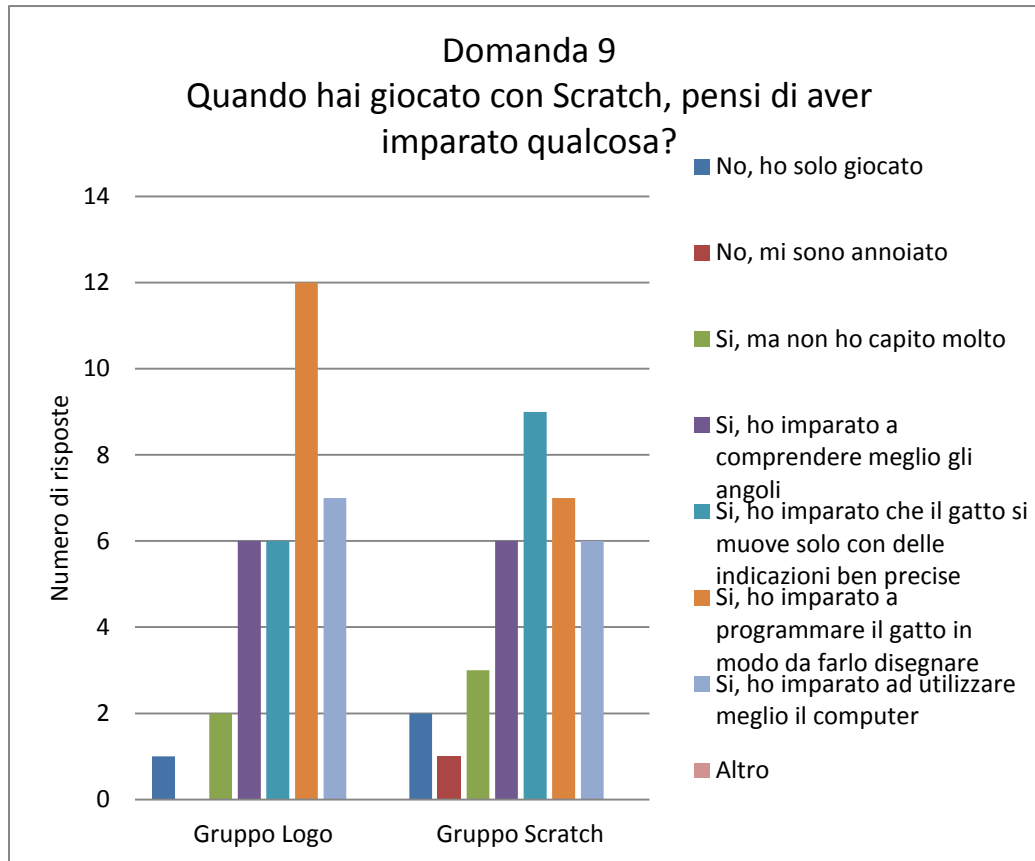
Le risposte alla Domanda 6 (come quelle della Domanda 7) sono state riassunte in un grafico a istogramma per visualizzare meglio in che numero sono state espresse determinate difficoltà. Anche in questo caso sono stati divisi il Gruppo Logo e il Gruppo Scratch per poter analizzare meglio i dati. La somma delle

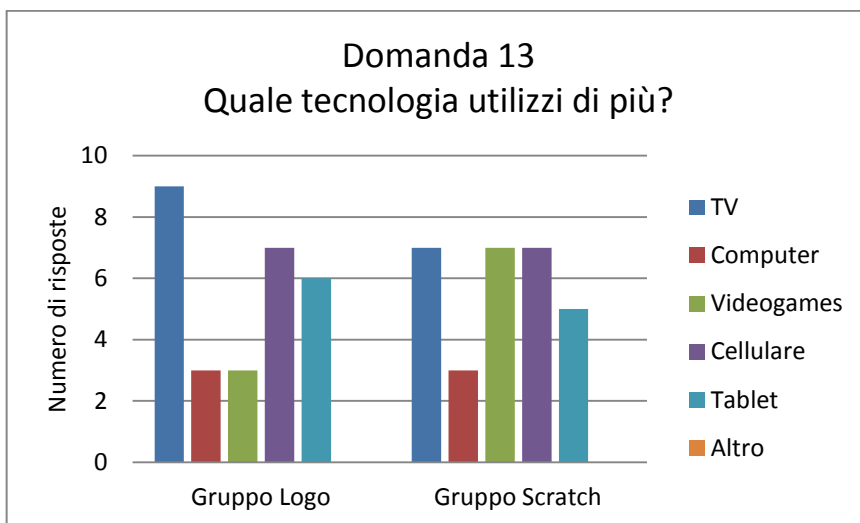
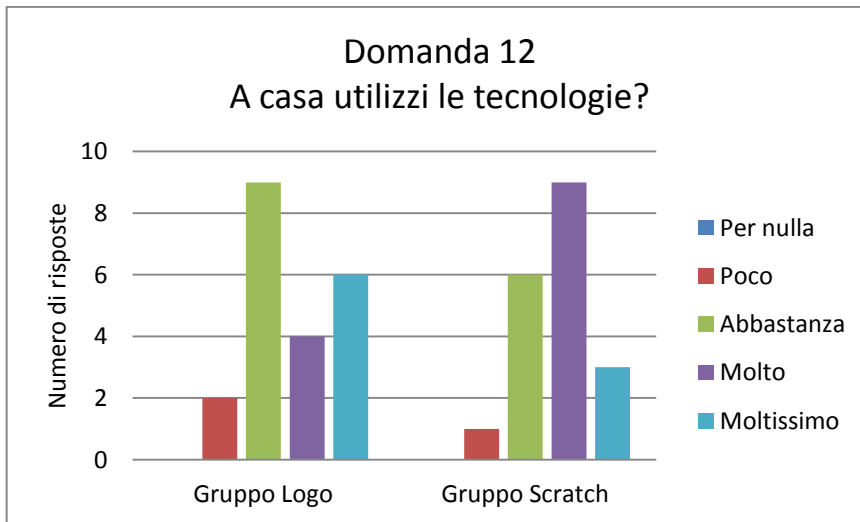
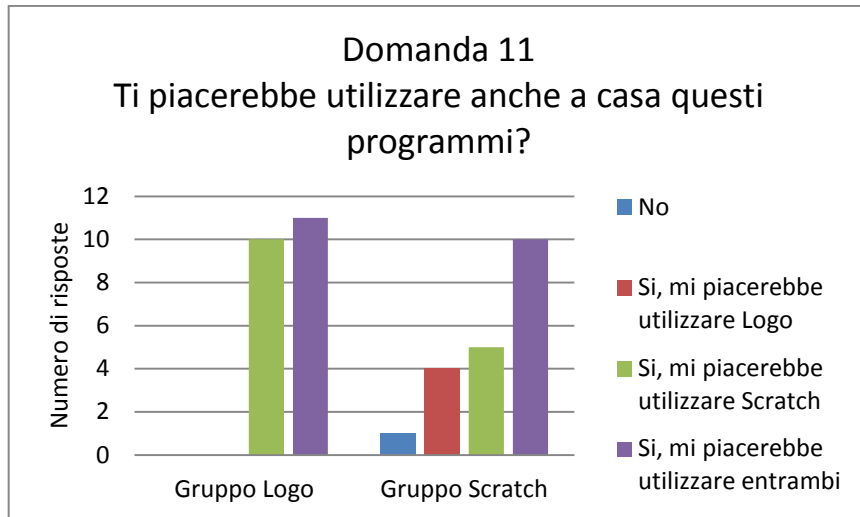
risposte del Gruppo Scratch è pari a 19 poiché una bambina era assente durante le lezioni.

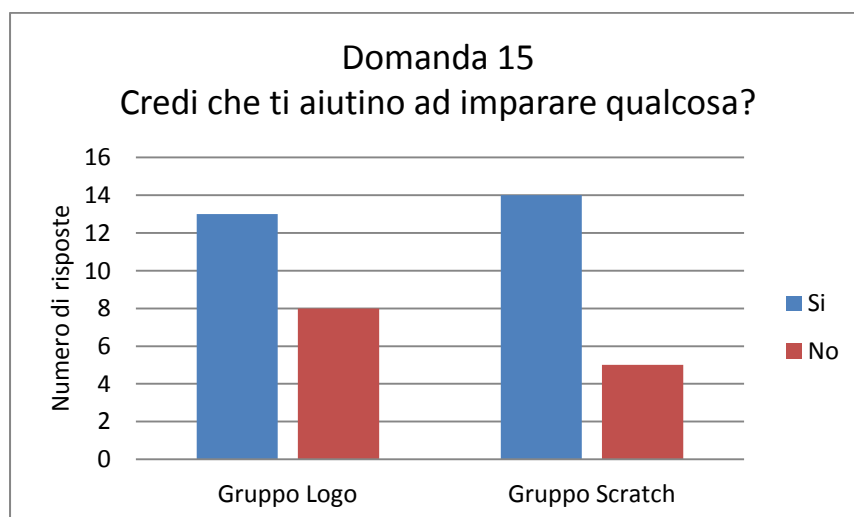
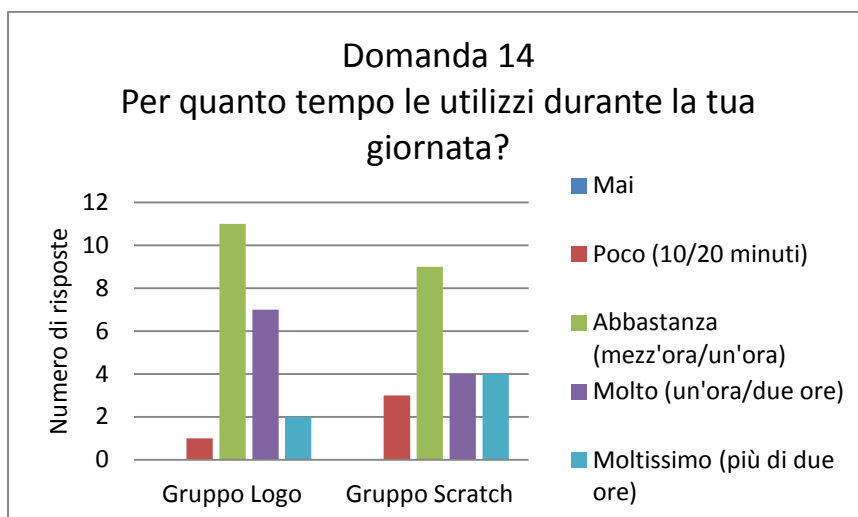


I bambini che hanno risposto "Altro" hanno aggiunto:

Ho imparato a programmare usando l'inglese; Non c'ero.





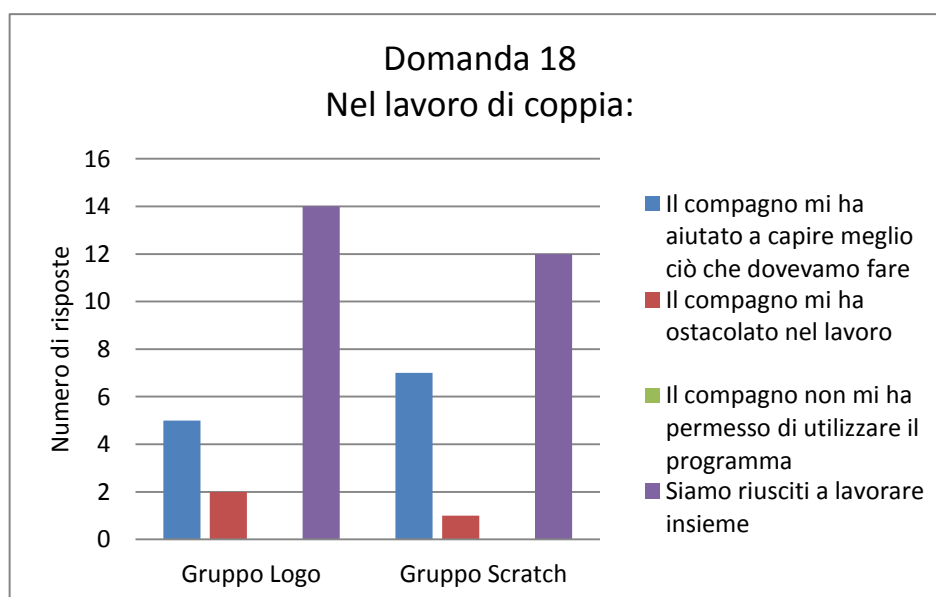
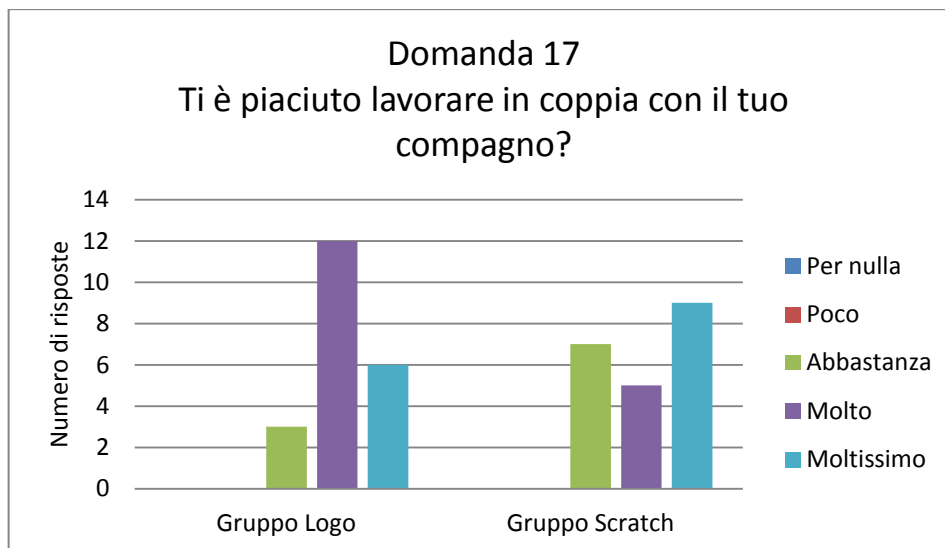


Domanda 16: Se sì, che cosa?

Alcuni bambini hanno frainteso questa domanda ed hanno risposto relativamente ai due programmi utilizzati. Riporto le testuali parole dei bambini:

Mi fanno scoprire molte cose; La creatività e la reattività; Non ci sono solo giochi violenti o video divertenti, ci sono anche giochi istruttivi, documentari, notizie online; Imparare cose nuove e come si utilizza il computer; Con la TV imparo ad ascoltare e a stare a contatto con mondo, invece il telefono non mi insegna tanto; Ad utilizzare il computer; Mi aiutano a capire meglio la geometria e a programmare; Possono insegnare alcune cose sui luoghi reali; Alcuni

programmi mi arricchiscono di informazioni; A comprendere cose nuove, con la TV si ricavano informazioni invece dai videogiochi impari a muoverti con il controller; Cose che non avevo imparato nella vita quotidiana; Dipende da quanto stai a giocare con le tecnologie perché possono sia allenare la mente che rammollirla; I documentari insegnano tante cose sul passato e sul mondo di adesso; Cose sulla natura; L'inglese, la logica; Saper usare meglio il computer; Mi hanno insegnato a non arrendermi; Dipende da cosa si guarda; Ho imparato la matematica e un po' l'inglese; A imparare la vita delle persone, tipo Anna Frank; Le informazioni e le curiosità dal mondo;



3.10 Interpretazione dei risultati

Le prime domande confermano che l'esperienza è stata affrontata in modo positivo o molto positivo dagli alunni; la maggior parte di loro ha trovato le lezioni di Logo e di Scratch divertenti e interessanti, una piccola percentuale le ha trovate alternative, mentre sia per il Gruppo Logo che per il Gruppo Scratch c'è stato un bambino che ha trovato stimolanti le lezioni con Logo. Arrivando alla Domanda 4 possiamo osservare che la maggioranza ha preferito lavorare con Scratch. È curioso comunque che molti bambini del Gruppo Scratch abbiano messo come preferenza il programma Logo, specificando che è stato più facile capire il suo funzionamento. Analizzando le risposte del Gruppo Logo, invece, si osserva che quasi tutti i bambini hanno preferito lavorare con Scratch perché, essendo le funzionalità le stesse, è stato più comodo avere i comandi già pronti per essere inseriti, senza il bisogno di scriverli interamente. Si può notare anche che nelle motivazioni non è rilevante il fatto della lingua, difficoltà invece espressa numerose volte nelle domande 6 e 7. Per il Gruppo Scratch, la difficoltà principale nell'usare Logo è stata proprio l'inglese, mentre per il Gruppo Logo è stata scrivere i comandi nel modo e nell'ordine giusto. Un solo bambino ha riscontrato difficoltà nel lavorare in coppia, mentre ho ritenuto importante la risposta di un alunno che ha espresso difficoltà nel trovare l'errore commesso nel codice scritto. Solo un bambino ha avuto difficoltà a creare gli angoli. Per quanto riguarda le difficoltà avute con Scratch, la maggior parte ha risposto con nessuna difficoltà; nel Gruppo Logo una piccola parte ha riscontrato difficoltà nel trovare i mattoncini e in pochi nell'utilizzo del computer e del web. Per il gruppo Scratch le difficoltà espresse sono state capire come usare i comandi e mettere i mattoncini al posto giusto, in quanto è stato per loro il primo programma sperimentato. In totale tre bambini hanno ritenuto difficile creare angoli e altre figure.

Nell'utilizzo dei due linguaggi di programmazione, quasi tutti i bambini hanno affermato di aver imparato a programmare e che sia il gatto che la tartaruga si muovono solo con indicazioni ben precise; inoltre molti hanno specificato di aver imparato ad utilizzare meglio il computer. Alcuni bambini hanno affermato di aver compreso meglio gli angoli. Le risposte negative sono state rilevanti solamente per quanto riguarda Scratch, infatti ben 3 bambini hanno affermato di

aver solamente giocato ed un bambino di essersi annoiato. Avendo osservato gli alunni durante le attività svolte in classe, posso affermare che entrambi i gruppi hanno ritenuto più facile lavorare con Scratch per motivi ben precisi: il Gruppo Logo, poiché avendo già compreso come costruire un codice scrivendolo da zero, ha trovato più semplice e veloce montarlo attraverso i mattoncini già predisposti; il Gruppo Scratch, poiché avendo imparato ad usare un linguaggio visuale, ha trovato più lento e complesso scrivere il codice di proprio pugno, per di più in inglese. Posso però confermare che per il Gruppo Logo è stato immediato comprendere il funzionamento di Scratch, visto che già conosceva il meccanismo di costruzione di un codice. Allo stesso modo, il Gruppo Scratch ha dimostrato di comprendere il funzionamento della programmazione solamente dopo aver lavorato con Logo; per questo due bambini hanno affermato di aver solamente giocato, mentre utilizzavano il linguaggio di programmazione visuale. Meritano particolare attenzione le domande 8 e 9 anche per quanto riguarda la percezione di cosa hanno appreso i bambini lavorando con i due programmi: se sommiamo le risposte dei due gruppi, notiamo che 21 bambini dichiarano di aver imparato a controllare la tartaruga in Logo, mentre 15 bambini il gatto in Scratch; in più 22 dichiarano di aver imparato ad usare meglio il computer con Logo, mentre 13 hanno avuto questa sensazione con Scratch. Se confrontiamo questi dati con le ricerche effettuate da Weintrop e Wilensky (2015a)³⁸, notiamo subito l'analogia: i risultati dei due studiosi mostrano che in generale viene preferita la programmazione grafica a quella testuale, tuttavia gli studenti hanno evidenziato alcuni svantaggi di questo linguaggio, come il fatto di "poter fare molto di più" con quello testuale. Allo stesso modo, Lewis (2010)³⁹, che aveva già messo a confronto Logo e Scratch per valutare i vantaggi del linguaggio a blocchi, aveva ottenuto risultati imprevisti, in quanto il livello di confidenza nelle proprie

³⁸ Gli autori cercano di mettere in luce il punto di vista degli studenti riguardo alle differenze della programmazione testuale e a blocchi, per capire quale risulti più facile e perché, quali sono le maggiori differenze, quali sono gli svantaggi percepiti nella programmazione a blocchi.

³⁹ Colleen Lewis spiega i risultati del suo studio in cui mette a confronto due gruppi di studenti di 10-12 anni che partecipano ad un corso estivo di programmazione, introducendoli ad essa prima uno con Logo e l'altro con Scratch e successivamente invertendo i ruoli. Questa sperimentazione effettuata nelle scuole primarie riprende esattamente la ricerca dell'autrice, base a cui fa riferimento tutto il lavoro.

capacità di programmazione si è rivelato significativamente superiore nei soggetti che avevano imparato a programmare con Logo.

Dalle successive domande emerge che solamente 5 bambini conoscevano già Scratch, che ad un bambino non interessa utilizzare questi programmi a casa e che molti vorrebbero utilizzarli entrambi. È curioso osservare che il Gruppo Logo presenta un numero elevato nella scelta di utilizzare Scratch e nessuna scelta per l'utilizzo di Logo, mentre nel Gruppo Scratch 4 bambini vorrebbero usare Logo e solo 5 bambini Scratch.

Il questionario pone l'attenzione anche sull'utilizzo delle tecnologie nella quotidianità. Dai risultati emerge che i bambini sono ben consapevoli di utilizzarli in modo a volte eccessivo o irresponsabile, infatti le risposte più frequenti sono state abbastanza, molto e moltissimo. Il dato più scoraggiante è stato quello relativo alle tecnologie più utilizzate: TV e cellulare risultano ai primi posti, contro una bassa percentuale dell'utilizzo del computer, strumento dai molteplici utilizzi, che può aiutare anche nella didattica.

L'ultimo dato rilevato dal questionario riguarda il lavoro di coppia: i bambini hanno dimostrato di essere abituati a lavorare in gruppo e molti hanno affermato di aver apprezzato anche in questo caso questa modalità; solo alcuni hanno sostenuto di essere stati ostacolati dal compagno.

Conclusioni

Giungendo alla fine dell'esperienza, si può affermare che, nonostante le particolarità relative al contesto, i risultati ottenuti sono rimasti in linea con le affermazioni già sostenute negli studi precedenti. Alcune considerazioni importanti possono essere fatte per quanto riguarda gli effetti della programmazione sulla geometria e sull'apprendimento dei concetti riguardanti l'angolo. Osservando i grafici che sono emersi si può notare che, nonostante molte risposte non abbiano evidenziato miglioramenti, alcune delle misconcezioni presenti negli alunni all'inizio del percorso sono state parzialmente corrette, o comunque sono stati appresi i concetti chiave su cui la struttura stessa della sperimentazione voleva porre l'attenzione. I bambini hanno dimostrato, già a metà del percorso ma in modo più significativo al suo termine, di aver compreso che nel confronto tra due angoli diversi, il maggiore è quello che ha la maggiore ampiezza. Nelle prime risposte, infatti, quasi tutti i bambini non hanno saputo dare la giusta spiegazione, tenendo in considerazione la lunghezza dei lati che formano l'angolo per identificare il maggiore; nell'ultimo test, invece, la maggior parte dei bambini ha saputo dare la risposta corretta motivandola in modo più chiaro. Il fatto che le risposte positive siano aumentate fino al terzo test dimostra probabilmente che il concetto è stato appreso e consolidato. Un altro concetto importante che dalle risposte sembra essere stato compreso maggiormente riguarda l'angolo retto. Seppure lievi, ci sono stati dei miglioramenti per quanto riguarda il riconoscimento dell'angolo retto e il confronto tra due angoli retti; un numero maggiore di bambini ha risposto correttamente nel terzo test, dimostrando di aver capito che l'angolo retto rimane tale a prescindere dalla lunghezza dei suoi lati e in qualsiasi posizione esso venga rivolto. Un riscontro molto positivo rispetto a questo concetto è dimostrato dal fatto che la misconcezione relativa alla conservazione dell'angolo è stata eliminata da diversi bambini, dato che le risposte corrette relative all'argomento sono nettamente aumentate, sia per quanto riguarda l'angolo retto, che per gli angoli del triangolo equilatero.

Se ci concentriamo sul secondo test si possono fare delle riflessioni per quanto riguarda le differenze tra i due linguaggi di programmazione. Per alcune domande si possono notare miglioramenti in entrambi i gruppi, questo significa che i concetti sono stati consolidati a prescindere dal tipo di programma con cui i bambini hanno lavorato. Nel gruppo Logo si può notare nella maggior parte delle domande, che il numero di risposte corrette in più nel Test 2 è leggermente superiore rispetto al gruppo Scratch; allo stesso modo che le risposte sbagliate sono presenti in numero minore. Questo indipendentemente dal numero di risposte corrette presenti nel Test 1.

Le considerazioni riguardanti le differenze tra i due linguaggi utilizzati riflettono in modo piuttosto chiaro i risultati già rilevati dalle ricerche precedenti. Il linguaggio preferito rimane quello a blocchi, con il quale i bambini si sono sentiti più a loro agio e con il quale trovano più semplice costruire un codice, ma per quanto riguarda la comprensione dei comandi e di come utilizzarli per creare un programma, le risposte evidenziano che l'esperienza più significativa è avvenuta con Logo.

Per trarre le conclusioni del lavoro svolto è opportuno tenere di conto oltre che dei risultati, anche dell'esperienza diretta svolta nelle classi. Gli alunni (ma anche i docenti) hanno affrontato questo progetto con entusiasmo. L'utilizzo dei computer si è dimostrato una strategia motivante e coinvolgente; i bambini hanno reagito in modo positivo alle sfide che venivano loro poste, con curiosità e spirito di iniziativa. Hanno avuto modo di dare libero sfogo alla loro creatività, grazie all'utilizzo dei due linguaggi di programmazione, creando ciò che volevano, nel modo che preferivano e che risultava per loro più logico e semplice. Sperimentando passo dopo passo le funzionalità dei programmi, hanno avuto l'opportunità di imparare da soli, la possibilità di ragionare per dare una risposta ai dubbi e per trovare le soluzioni più adeguate, tanto che il ruolo dell'insegnante si è limitato al supporto e alla supervisione dall'esterno. In questo modo hanno capito sia che cosa fare che come farlo e i concetti appresi sono rimasti impressi, avendone avuta un'esperienza pratica e diretta. Lavorare in un contesto come questo ha permesso a tutti di mettersi in gioco, anche a quei bambini che, apparentemente, non avevano passione per la matematica. Essi hanno reagito in

modo positivo non solo al tipo di attività richiesta, ma anche alle difficoltà, grazie alla possibilità di poter sbagliare senza nessun tipo di conseguenza, di provare e riprovare fino ad arrivare ad una soluzione accettabile; e sono stati proprio questi bambini che hanno avuto le soddisfazioni più grandi, riuscendo in ciò che dovevano e volevano fare. L'esperienza è stata quindi nella sua globalità molto positiva, in quanto ha permesso di sviluppare e allenare quelle capacità di analisi e risoluzione dei problemi che sono proprie del pensiero computazionale.

Bibliografia

Andreoli, R. A. (1996), *Micromondi linguistici. L'uso di LOGO nella didattica dell'italiano*, Firenze: La Nuova Italia

Andronico, A. (1996), *Non è possibile sostituire l'insegnante: benissimo*, in "TD - Tecnologie Didattiche", 11, pp. 10-16.

Antonietti, A., Cantoia, M. (2001), *Imparare con il computer. Come costruire contesti di apprendimento per il software*, Trento: Erickson.

Bennett, S. & Maton, K. (2010), *Beyond the 'digital natives' debate: towards a more nuanced understanding of students' technology*, in "Journal of Computer Assisted Learning, 26 (5), pp. 321-331.

Bennett, S., Maton, K., Kervin, L. (2008), *The 'digital natives' debate: A critical review of the evidence*, in "British Journal of Educational Technology", 39 (5), pp. 775-786.

Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., Engelhardt, K. (2016) *Developing Computational Thinking in Compulsory Education. Implications for policy and practice*, EUR 28295 EN.

Bogliolo, A. (2016), *Coding in Your Classroom, Now!* Firenze: Giunti.

Bonaiuti, G., Calvani, A., Ranieri, M. (2007), *Fondamenti di didattica. Teoria e prassi dei dispositivi formativi*, Roma: Carocci

Brennan, K., Resnick, M. (2012), *New frameworks for studying and assessing the development of computational thinking*, in "Proceedings of the 2012 annual meeting of the American Educational Research Association", pp. 1-25, Vancouver.

Calvani, A. (1999), *I nuovi media nella scuola. Perché, come, quando avvalersene*. Roma: Carocci.

Calvani, A., Fini, A., Ranieri, M. (2010), *La competenza digitale nella scuola*. Trento: Erickson.

Calvani, A. (2011), *Principi dell'istruzione e strategie per insegnare. Criteri per una didattica efficace*, Roma: Carocci.

Capponi, M. (2003). *Il computer come ambiente di apprendimento*. Perugia: Morlacchi.

Capponi, M. (2009), *Un giocattolo per la mente. L'«informatica cognitiva» di Seymour Papert*. Perugia: Morlacchi.

Clements, D. H., Battista, M. (1989, Novembre), *Learning of Geometric Concepts in a Logo Environment*, in "Journal for Research in Mathematics Education", 20(5), pp. 450-467.

Clements, D. H., Battista, M. (1990, Novembre), *The Effects of Logo on Children's Conceptualizations of Angle and Polygons*, in "Journal for Research in Mathematics Education", 21 (5), pp. 356-371.

Commissione Europea (2018), *Digital Education Action Plan*.

D'Amore, B. (1999), *Elementi di didattica della matematica*, Bologna: Pitagora.

Dewey, J. (1973), *Scuola e società*, Firenze: La nuova Italia.

Formiconi, A. R. (2017, Aprile), *Bibliografia ragionata su coding e pensiero computazionale nel contesto scolastico*, v.0.3.

Formiconi, A. R. (2018), *Piccolo Manuale di LibreLogo. La Geometria della Tartaruga*.

Formiconi, A. R. (Producer), (2018a), *Coding at school with Free Software* [MOOC]. Retrieved from <https://www.federica.eu/>.

Formiconi, A. R., Mancini, F. (2017), *A step back into the future*, University of Florence.

Giofrè, D., Mammarella, I., Lucangeli, D. (2009, Febbraio), *L'apprendimento della geometria in bambini da 4 a 6 anni*, in "Difficoltà in Matematica", 5(2), pp. 21-38.

Harel, I., Papert, S. (a cura di) (1991), *Costructionism*, Norwood (NJ): Ablex.

Kieran, C. (1986), *Turns and angles: What develops in Logo?*, Proceedings of the Eighth Annual Meeting of the North American Chapter of the International Group for the Psychology of Mathematics Education, Lansing: Michigan State University.

Lanzara, G. F. (2000), *La logica del bricolage. Osservazioni sulle pratiche "minori" della progettazione*, presentato al seminario "Ingegneria e scienze umane: mappe mentali e geografie istituzionali", NEHS, Bologna.

Lévi-Strauss, C. (1962), *La pensée sauvage*, Parigi: Plon, tr.it. *Il pensiero selvaggio* (1965), Milano: Il Saggiatore.

Lewis, C. M. (2010), *How programming environment shapes perception, learning and goals: Logo vs. Scratch*, Proceeding SIGCSE '10, Proceedings of the 41st ACM technical symposium on Computer science education, Pages 346-350.

Longo, G. O. (2000), *Il futuro tra incertezza e responsabilità*, atti del convegno "Il futuro: previsione, pronostico e profezia", Istituto Veneto di Scienze Lettere ed Arti.

Marchignoli, R., Lodi, M. (2016), *EAS e pensiero computazionale*, Brescia: La Scuola.

Ministero dell'Istruzione, dell'Università e della Ricerca (2012), *Indicazioni Nazionali per il curriculum della scuola dell'infanzia e del primo ciclo d'istruzione*.

Ministero dell'Istruzione, dell'Università e della Ricerca (2015), *Piano Nazionale Scuola digitale*.

Ministero dell'Istruzione, dell'Università e della Ricerca (2018), *Indicazioni Nazionali e Nuovi Scenari*.

Montessori, M. (1966), *La mente del bambino. Mente assorbente*, Milano: Garzanti.

Noss, R. (1987), *Children's Learning of Geometrical Concepts through Logo*, in "Journal for Research in Mathematics Education, 18(5), pp. 343-362.

Noss, R. (1988), *Imparare a programmare in LOGO. Nuovi approcci alla programmazione del computer*, in Tagg, B., (a cura di), *Il software didattico per insegnanti e genitori*, Roma: Armando.

Papert, S. (1984), *Mindstorms. Bambini computers e creatività*, Torino: Emme Edizioni.

Papert, S. (1993), *I bambini e il computer*, Milano: Rizzoli.

Papert, S. (1996), *The Connected Family*, Atlanta: Longstreet.

Papert, S. (1997, Marzo), *Bambini e adulti a scuola con il computer*, in "Mediamente", Venezia.

Prensky, M. (2001), *Digital Natives, Digital Immigrants*, in "On the Horizon", 9 (5), pp. 1-6.

Ranieri, M. (2011), *Le insidie dell'ovvio. Tecnologie educative e critica della retorica tecnocentrica*, Pisa: ETS.

Reggini, H. C. (1982), *Logo: ali per la mente*, Milano: Mondadori.

Rivoltella, P. C. (2016, Marzo), *Coding Time*, in "Scuola Italiana Moderna, 123 (7), pp. 1-2.

Simmons, M., Cope, P. (1990, Agosto), *Fragile Knowledge of Angle in Turtle Geometry*, in "Educational Studies in Mathematics", 21(4), pp. 375-382.

Stefik, A., Siebert, S. (2013), *An Empirical Investigation into Programming Language Syntax*, in ACM Transactions on Computing Education, 13 (4), pp. 1-40.

Yadav, A., (2014), *Computational Thinking in K-12 Education* [PowerPoint slides], Purdue University, Indiana.

Weintrop, D., Wilensky, U. (2015), *Using commutative assessments to compare conceptual understanding in blocks-based and text-based programs*, Proceeding ICER '15, Proceedings of the eleventh annual International Conference on International Computing Education Research, pp. 101-110.

Weintrop, D., Wilensky, U., (2015a), *To block or not to block, that is the question: students' perceptions of blocks-based programming*, Proceeding IDC '15, Proceedings of the 14th International Conference on Interaction Design and Children, pp. 199-208.

Wing, J. M. (2006), *Computational Thinking*, in "Communications of the ACM", 49 (3), pp. 33-35.

Sitografia

- <https://eur-lex.europa.eu/legal-content/IT/TXT/?uri=celex%3A32006H0962>
(Consultato il 25 luglio 2019)
- <https://www.miur.gov.it/scuola-digitale> (Consultato il 26 luglio 2019)
- https://ec.europa.eu/education/education-in-the-eu/digital-education-action-plan_en (Consultato il 2 agosto 2019)
- https://platform.europeanmoocs.eu/course_coding_in_your_classroom_now
(Consultato il 2 agosto 2019)
- <https://code.org/> (Consultato il 3 agosto 2019)
- <https://africacodeweek.org/about/about1/> (Consultato il 2 agosto 2019)
- <https://hourofcode.com/it> (Consultato il 3 agosto 2019)
- <https://coderdojo.com/movement/> (Consultato il 3 agosto 2019)
- <https://programmaitfuturo.it/> (Consultato il 3 agosto 2019)
- <https://www.libreitalia.org/librelogo-coding-con-libreoffice/> (Consultato il 6 agosto 2019)
- <https://turtleacademy.com/lessons> (Consultato il 5 agosto 2019)
- <http://www.microworlds.com/company/philosophy.pdf> (Consultato il 07 agosto 2019)
- <https://www.python.it/> (Consultato il 12 agosto 2019)
- <http://www.treccani.it/enciclopedia/numerazione> (Consultato il 16 agosto 2019)
- <http://www.robocupjr.it/margi/pubblicazioni/1203.pdf> (Consultato il 16 agosto 2019)

Appendice

Test 1

Nome e Cognome..... Data.....

TEST DI GEOMETRIA N°1

1) Cos'è un angolo? Disegnane uno.

.....
.....

2) Puoi disegnare un angolo maggiore rispetto a quello che hai disegnato prima? Perché è maggiore?

.....
.....

3) Cerchia tra queste figure quelle che rappresentano angoli.



4) Quanti angoli hanno queste figure? Scrivi il numero di angoli sotto ad ogni figura.



..... Angoli



..... Angoli

5) Stai camminando su questo sentiero. Inizi dal punto A e finisci nel punto G. In quale punto devi ruotare di più? Indicalo con una freccia di colore blu. In quale punto devi ruotare di meno? Indicalo con una freccia di colore rosso.



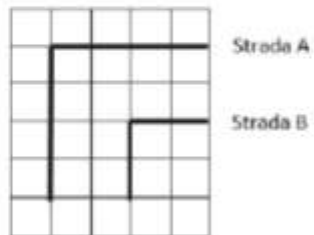
6) Un robot ruota di 90 gradi ogni volta che ruota. Quante rotazioni deve fare il robot per puntare nella stessa direzione iniziale? Perché?

.....

7) Un robot ruota di 30 gradi ogni volta. Quante rotazioni deve fare prima di puntare la stessa direzione iniziale? Perché?

.....

8) Nella città chiamata Quadrata tutte le strade sono disposte su uno schema quadrato. Ci sono solo due strade principali, che sulla mappa sono rappresentate da linee più scure.



Seleziona la risposta che pensi sia corretta:

La strada A curva di più della strada B

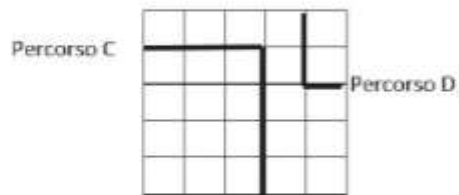
La strada B curva di più della strada A

La strada A e la strada B curvano

nello stesso modo

Non puoi dirlo

9) Nella città di Quadrata ci sono due nuove strade C e D



Seleziona la risposta che pensi sia corretta:

- La strada C curva più della strada D
- La strada D curva più della strada C
- Le strade C e D curvano nello stesso modo
- Non puoi dirlo

10) Osserva gli angoli A e B di questi due triangoli.



Seleziona la risposta che pensi sia corretta:

- L'angolo A è maggiore dell'angolo B
- L'angolo B è maggiore dell'angolo A
- Gli angoli A e B hanno la stessa ampiezza
- Non puoi dirlo

11) In ognuno di questi triangoli tutti i lati hanno la stessa lunghezza.



Seleziona la risposta che pensi sia corretta:

L'angolo C è maggiore dell'angolo D

L'angolo D è maggiore dell'angolo C

Gli angoli C e D hanno la stessa ampiezza

Non puoi dirlo

12) Osserva gli angoli A e B



Seleziona la risposta che pensi sia corretta:

L'angolo A è maggiore dell'angolo B

L'angolo B è maggiore dell'angolo A

Gli angoli A e B hanno la stessa ampiezza

Non puoi dirlo

13) Osserva gli angoli:



Seleziona la risposta che pensi sia corretta:

L'angolo C è maggiore dell'angolo D

L'angolo D è maggiore dell'angolo C

Gli angoli C e D hanno la stessa ampiezza

Non puoi dirlo

Test 2

Nome e Cognome.....Data.....

TEST DI GEOMETRIA N°2

1) Cos'è un angolo? Disegnane uno.

.....
.....

2) Puoi disegnare un angolo maggiore rispetto a quello che hai disegnato prima? Perché è maggiore?

.....
.....

3) Cerchia tra queste figure quelle che rappresentano angoli.



4) Quanti angoli hanno queste figure?
Scrivi il numero di angoli sotto ad ogni figura.

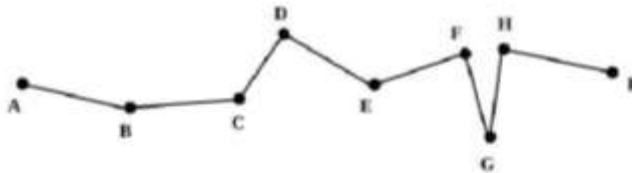


..... Angoli



..... Angoli

5) Stai camminando su questo sentiero. Inizi dal punto A e finisci nel punto I. In quale punto devi ruotare di più? Indicalo con una freccia di colore blu. In quale punto devi ruotare di meno? Indicalo con una freccia di colore rosso.



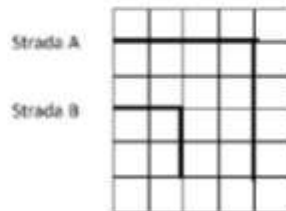
6) Un robot ruota di 60 gradi ogni volta che ruota. Quante rotazioni deve fare il robot per puntare nella stessa direzione iniziale? Perché?

.....

7) Un robot ruota di 90 gradi ogni volta che ruota. Quante rotazioni deve fare il robot per puntare nella direzione opposta? Perché?

.....

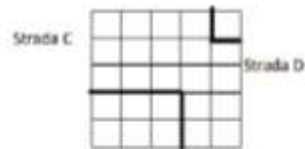
8) Nella città chiamata Quadrata tutte le strade sono disposte su uno schema quadrato. Ci sono solo due strade principali, che sulla mappa sono rappresentate da linee più scure.



Seleziona la risposta che pensi sia corretta:

- La strada A curva di più della strada B
- La strada B curva di più della strada A
- La strada A e la strada B curvano nello stesso modo
- Non puoi dirlo

9) Nella città di Quadrata ci sono due nuove strade C e D



Seleziona la risposta che pensi sia corretta:

- La strada C curva più della strada D
- La strada D curva più della strada C

Le strade C e D curvano nello stesso modo

Non puoi dirlo

10) Osserva gli angoli A e B di questi due triangoli.



Seleziona la risposta che pensi sia corretta:

L'angolo A è maggiore dell'angolo B

L'angolo B è maggiore dell'angolo A

Gli angoli A e B hanno la stessa ampiezza

Non puoi dirlo

11) In ognuno di questi triangoli tutti i lati hanno la stessa lunghezza.



Seleziona la risposta che pensi sia corretta:

L'angolo C è maggiore dell'angolo D

L'angolo D è maggiore dell'angolo C

Gli angoli C e D hanno la stessa ampiezza

Non puoi dirlo

12) Osserva gli angoli A e B



Seleziona la risposta che pensi sia corretta:

L'angolo A è maggiore dell'angolo B

L'angolo B è maggiore dell'angolo A

Gli angoli A e B hanno la stessa ampiezza

Non puoi dirlo

13) Osserva gli angoli



Seleziona la risposta che pensi sia corretta:

L'angolo C è maggiore dell'angolo D

L'angolo D è maggiore dell'angolo C

Gli angoli C e D hanno la stessa ampiezza

Non puoi dirlo

Test 3

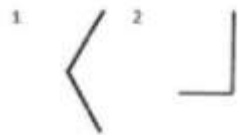
Nome e Cognome.....Data.....

TEST DI GEOMETRIA N°3

1) Cos'è un angolo? Disegnane uno.

.....

2) Quale tra i due angoli è il maggiore? Perché?



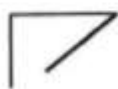
.....

.....

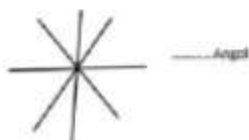
3) In ogni coppia di figura, individua l'angolo e cerchiolo



4) Quanti angoli hanno queste figure? Scrivi il numero di angoli sotto ad ogni figura.



..... Angoli



..... Angoli

5)



Questo è il sentiero che il robot ha seguito per arrivare dal punto A al punto G. Adesso prova tu a disegnare un sentiero che inizi nel punto A e finisca nel punto G. Ad ogni cambio di direzione scrivi una lettera. Indica con una freccia di colore blu il punto in cui devi ruotare di più e con una freccia di colore rosso il punto in cui devi ruotare di meno.

A

G

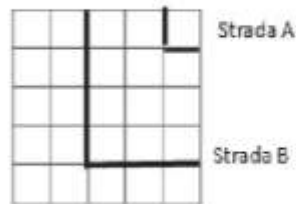
6) Un robot ruota di 120 gradi ogni volta che ruota. Quante rotazioni deve fare il robot per puntare nella stessa direzione iniziale? Perché?

.....

7) Un robot ruota di 45 gradi ogni volta che ruota. Quante rotazioni deve fare il robot per puntare nella direzione opposta? Perché?

.....

8) Nella città chiamata Quadrata tutte le strade sono disposte su uno schema quadrato. Ci sono solo due strade principali, che sulla mappa sono rappresentate da linee più scure.



Seleziona la risposta che pensi sia corretta:

La strada A curva di più della strada B

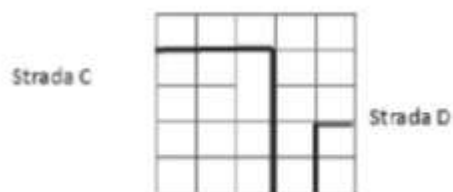
La strada B curva di più della strada A

La strada A e la strada B curvano

nello stesso modo

Non puoi dirlo

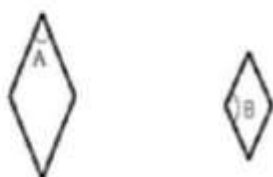
9) Nella città di Quadrata ci sono due nuove strade C e D



Seleziona la risposta che pensi sia corretta:

- La strada C curva più della strada D
- La strada D curva più della strada C
- Le strade C e D curvano nello stesso modo
- Non puoi dirlo

10) Osserva gli angoli A e B di questi rombi



Seleziona la risposta che pensi sia corretta:

- L'angolo A è maggiore dell'angolo B
- L'angolo B è maggiore dell'angolo A
- Gli angoli A e B hanno la stessa ampiezza
- Non puoi dirlo

11) In ognuno di questi triangoli tutti i lati hanno la stessa lunghezza.



Seleziona la risposta che pensi sia corretta:

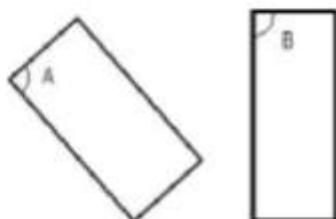
L'angolo C è maggiore dell'angolo D

L'angolo D è maggiore dell'angolo C

Gli angoli C e D hanno la stessa ampiezza

Non puoi dirlo

12) Osserva gli angoli A e B



Seleziona la risposta che pensi sia corretta:

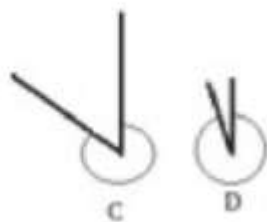
L'angolo A è maggiore dell'angolo B

L'angolo B è maggiore dell'angolo A

Gli angoli A e B hanno la stessa ampiezza

Non puoi dirlo

13) Osserva gli angoli



Seleziona la risposta che pensi sia corretta:

L'angolo C è maggiore dell'angolo D

L'angolo D è maggiore dell'angolo C

Gli angoli C e D hanno la stessa ampiezza

Non puoi dirlo

QUESTIONARIO

Quale programma hai utilizzato per primo in questo laboratorio?

- Logo
- Scratch

1. Quanto ti è piaciuta, in generale, questa esperienza?

- Per nulla
- Poco
- Abbastanza
- Molto
- Moltissimo

2. Come ti sono sembrate le lezioni con Logo?

- Stimolanti
- Noiose
- Divertenti
- Interessanti
- Alternative
-

Altro.....

3. Come ti sono sembrate le lezioni con Scratch?

- Stimolanti
- Noiose
- Divertenti
- Interessanti
- Alternative
-

Altro.....

4. Con quale programma ti è piaciuto di più lavorare?
- Scratch
 - Logo
5. Perché?
-
6. Quali difficoltà hai avuto con Logo?
-
7. Quali difficoltà hai avuto con Scratch?
-
8. Quando hai giocato con Logo, pensi di aver imparato qualcosa?
(Puoi scegliere più di un'opzione)
- No, ho solo giocato
 - No, mi sono annoiato
 - Sì, ma non ho capito molto
 - Sì, ho imparato a comprendere meglio gli angoli
 - Sì, ho imparato che la tartaruga si muove solo con delle indicazioni ben precise
 - Sì, ho imparato a programmare la tartaruga in modo da farla disegnare
 - Sì, ho imparato ad utilizzare meglio il computer
 -
- Altro.....
9. Quando hai giocato con Scratch, pensi di aver imparato qualcosa? (Puoi scegliere più di un'opzione)
- No, ho solo giocato
 - No, mi sono annoiato
 - Sì, ma non ho capito molto
 - Sì, ho imparato a comprendere meglio gli angoli
 - Sì, ho imparato che il gatto si muove solo con delle indicazioni ben precise

- Sì, ho imparato a programmare il gatto in modo da farlo disegnare
- Sì, ho imparato ad utilizzare meglio il computer
-

Altro.....

10. Hai mai utilizzato questi programmi prima di questo laboratorio?

- No
- Sì, ho utilizzato Scratch
- Sì, ho utilizzato Logo
- Sì, ho utilizzato entrambi

11. Ti piacerebbe utilizzare anche a casa questi programmi?

- No
- Sì, mi piacerebbe utilizzare Logo
- Sì, mi piacerebbe utilizzare Scratch
- Sì, mi piacerebbe utilizzare entrambi

12. A casa utilizzi le tecnologie?

- Per nulla
- Poco
- Abbastanza
- Molto
- Moltissimo

13. Quale tecnologia utilizzi di più?

- TV
- Computer
- Videogames
- Cellulare
- Tablet
-

Altro.....

14. Per quanto tempo le utilizzi durante la tua giornata?
- Mai
 - Poco (10/20 minuti)
 - Abbastanza (mezz'ora/ un'ora)
 - Molto (un'ora/due ore)
 - Moltissimo (più di due ore)
15. Credi che ti aiutino ad imparare qualcosa?
- Sì
 - No
16. Se sì, che cosa?
-
17. Ti è piaciuto lavorare in coppia con il tuo compagno?
- Per nulla
 - Poco
 - Abbastanza
 - Molto
 - Moltissimo
18. Nel lavoro di coppia:
- Il compagno mi ha aiutato a capire meglio ciò che dovevamo fare
 - Il mio compagno mi ha ostacolato nel lavoro
 - Il mio compagno non mi ha permesso di utilizzare il programma
 - Siamo riusciti a lavorare insieme



UNIVERSITÀ
DEGLI STUDI
FIRENZE
SCIFOPSI
DIPARTIMENTO DI
SCIENZE DELLA FORMAZIONE
E PSICOLOGIA

*Corso di Studi in Scienze
della Formazione Primaria*

Relazione finale di tirocinio

Tirocinante	Claudia Giannelli
Tutor scolastici	Gianna Dell'Aiuto - Annamaria Biondi (scuola dell'infanzia) Luisa Martinucci (scuola primaria)
Tutor universitario	Lucia Donata Nepi

Anno Accademico 2018 - 2019

Indice

Introduzione	3
Capitolo 1 - Le sedi del tirocinio	4
1.1 Il territorio.....	4
1.2 La scuola	4
1.3 Documentazione	5
Capitolo 2 - Il tirocinio nella scuola dell'infanzia	7
2.1 I locali e l'organizzazione della Scuola dell'Infanzia	7
2.2 T1.....	8
2.3 T2.....	10
2.4 T3.....	14
2.5 T4.....	16
Capitolo 3 - Il tirocinio nella scuola primaria	18
3.1 I locali e l'organizzazione della Scuola Primaria	18
3.2 Descrizione delle classi.....	19
3.3 L'esperienza in classe: T1 e T2 nella scuola primaria	20
3.4 La progettazione didattica nella scuola primaria e il modello MARC .	21
Capitolo 4 - La scuola come comunità professionale	28
4.1 L'organizzazione scolastica e le modalità di lavoro collegiale	28
Conclusioni	31
Bibliografia	32
Sitografia.....	33

Introduzione

Giungendo al termine del percorso universitario che ho intrapreso e riflettendo sui 5 anni appena trascorsi, mi rendo conto che scegliere di iscrivermi al Corso di Studi in Scienze della Formazione Primaria non è stato affatto semplice. L'idea di "fare la maestra" per me non è nata sui banchi di scuola, ma è stata la scelta conclusiva di un percorso personale pieno di delusioni, emozioni e vittorie che mi hanno fatto capire che cosa volevo fare nella mia vita.

Le mie esperienze con i bambini sono state molte: all'età di 15 anni ho iniziato come volontaria con i Gruppi Estivi organizzati dalla Parrocchia del mio paese e per alcuni anni ho affiancato i bambini nel percorso catechistico. Dalla maggiore età, ho avuto l'occasione di lavorare come animatrice in modo retribuito, soprattutto nei campi estivi organizzati da cooperative di zona e dalla ASL. Da quel momento sono stata chiamata in molte famiglie come baby-sitter. Dopo il diploma di liceo scientifico ho iniziato il percorso di studi universitari frequentando per un anno la Facoltà di Scienze Biologiche all'Università di Pisa, fino a quando ho scelto di provare a intraprendere la strada dell'insegnamento.

Durante questi cinque anni, ho avuto occasione di approfondire la conoscenza di discipline come la Pedagogia e la Psicologia, indispensabili nel percorso educativo. Il corso di studi inoltre prevede quattro anni di tirocinio diretto nelle scuole, esperienza completa e formativa che permette di svolgere un percorso parallelo a quello teorico per mettere in pratica le conoscenze e sviluppare le abilità necessarie per poter essere un buon insegnante.

Questa relazione ripercorre quattro anni di esperienze e formazione in un'ottica riflessiva, analizzando anno dopo anno gli aspetti positivi e di crescita personale e professionale, ma anche le criticità emerse e i modi in cui sono state affrontate.

Capitolo 1

Le sedi del tirocinio

1.1 Il territorio

L'istituto dove ho svolto il tirocinio è l'Istituto Comprensivo Volterra¹, l'unico presente nella mia città di origine, che mi ha accolto per tutti e quattro gli anni del tirocinio. Esso comprende una larga parte del territorio dei comuni di Volterra e di Montecatini Val di Cecina, infatti i plessi di scuola dell'infanzia in totale sono 6, mentre quelli della scuola primaria sono 5, ubicati nelle diverse frazioni. Le scuole secondarie di 1° grado invece sono due, presenti una in ogni comune. Il territorio, anche se piuttosto ampio, è scarsamente popolato, e con una rete di comunicazioni non sempre efficiente. Dal punto di vista economico è rilevante il settore terziario e molto importante quello primario, anche da un punto di vista culturale. Il fenomeno dell'immigrazione, anche se presente, rimane marginale. Negli ultimi anni si è assistito a una leggera diminuzione della popolazione attiva, che ha influito anche sulle iscrizioni scolastiche, soprattutto nelle frazioni. Il numero degli alunni, pur rimanendo su un totale di 900-950 iscritti all'anno, ha evidenziato un calo soprattutto per quanto riguarda la scuola dell'infanzia, dovuto sicuramente al calo delle nascite. È assente invece il fenomeno dell'abbandono scolastico.

1.2 La scuola

I plessi dove si è svolto il tirocinio sono la Scuola dell'Infanzia e la Scuola Primaria di San Lino, situati nel centro della città di Volterra. Le scuole occupano gli spazi di un vecchio palazzo che una volta era un convento, sono accessibili da due ingressi separati, ma sono comunicanti dall'interno, in quanto condividono il giardino, la mensa e la cucina. La scuola dell'infanzia si trova al piano terra; le tre sezioni, divise in modo omogeneo per età, sono suddivise in tre aule disposte su un lato di un grande corridoio, insieme a una stanza riservata alle insegnanti e

¹ Per consultare la pagina web dell'istituto: <http://www.icsvolterra.it/home.html>

all'aula multimediale, dove è presente un televisore e vari lettori digitali. Dall'altro lato troviamo la cucina, la mensa e un grande salone con giochi. Il corridoio inizia dall'ingresso della scuola, dove è presente anche un atrio e uno spazio per appendiabiti, mentre dall'altro lato ci sono i bagni e l'ingresso al giardino. La scuola primaria occupa tutto il piano superiore, le classi sono divise in due sezioni, A e B, quindi in tutto le aule sono 10. Solamente l'ultimo anno di tirocinio sono state occupate solo 9 aule, in quanto è stata formata una sola sezione Prima. L'aula rimasta vuota è stata usata come classe per l'attività alternativa. La scuola possiede una aula magna, una palestra, un piccolo ambulatorio, la direzione e un laboratorio di informatica; quest'ultimo purtroppo attualmente non è in uso in quanto la strumentazione non è utilizzabile. Comunque ogni classe, compresa l'aula magna, è dotata di una LIM e di almeno un computer portatile per ogni sezione.

1.3 Documentazione

Come documento di presentazione della scuola, il PTOF² ha avuto un ruolo fondamentale durante il tirocinio diretto. Il Piano Triennale dell'Offerta Formativa³, adottato dal Consiglio di Istituto ogni 3 anni, ma modificabile ogni anno in base alle necessità e alle risorse del territorio, contiene lo scopo formativo ed educativo dell'istituto e la strategia utilizzata per perseguire tale scopo, si rivela quindi lo strumento di attuazione dell'autonomia scolastica. È strumento programmatico e informativo, che contiene tutta la programmazione curricolare ed extracurricolare e la dimensione organizzativa dell'istituto. Come il PTOF, anche il RAV, ovvero il Rapporto di Autovalutazione, è stato uno strumento rilevante in quanto al suo interno si possono ritrovare gli obiettivi che si prefissa l'istituto in funzione di una attività di analisi valutativa interna all'istituzione scolastica. Ultimo ma non meno importante, il documento di riferimento di tutta la progettazione curricolare, ovvero le *Indicazioni Nazionali per il curricolo della scuola dell'infanzia e del primo ciclo di istruzione* del 2012. Queste sono servite

² Per consultare il PTOF dell'istituto: <http://www.icsvolterra.it/pof/109-piano-dellofferta-formativa-pof/2137-ptof-del-08-11-2018-allegati.html>

³ Il documento è stato introdotto dalla legge 107/2015, legge della "Buona Scuola".

durante tutto il percorso formativo, in quanto descrivono gli obiettivi di apprendimento e i traguardi per lo sviluppo di competenze che ogni alunno e ogni alunna deve raggiungere, al termine della scuola dell'infanzia, della classe terza e quinta primaria e al termine della scuola secondaria di primo grado. Le Indicazioni Nazionali sono state fondamentali, soprattutto per quanto riguarda la progettazione delle unità di apprendimento.

Capitolo 2

Il tirocinio nella scuola dell'infanzia

2.1 I locali e l'organizzazione della Scuola dell'Infanzia

Il tirocinio diretto nella scuola dell'infanzia si è svolto nel plesso di San Lino, descritto precedentemente⁴. Su consiglio del Dirigente Scolastico, ho svolto i primi tre anni (T1, T2 e T3) nella stessa classe, seguendo i bambini dai 3 ai 6 anni. Questo percorso di continuità è stato molto interessante in quanto mi ha dato la possibilità di affrontare un vero percorso di crescita personale insieme a quello dei bambini che ho avuto la fortuna di conoscere. Le aule della scuola sono molto spaziose, organizzate più o meno tutte allo stesso modo: sono presenti dei grandi tavoli che accolgono fino a 6 bambini ciascuno, utilizzati per attività laboratoriali, manuali o di pittura; in un angolo sono sistemate le panchine, che servono per attività di ascolto e di conversazione, vicino alle quali è organizzata una piccola biblioteca; in un lato della stanza sono riposti alcuni giochi, mentre alle pareti sono appesi cartelloni, lavori fatti dai bambini, vari materiali per le insegnanti. In ogni aula è presente inoltre uno scaffale, suddiviso in spazi assegnati a ciascun bambino, per riporre oggetti personali, e una scrivania per le insegnanti. La scuola è aperta dalle ore 8 alle ore 16, le insegnanti, due per sezione, hanno un orario di 5 ore giornaliere suddiviso in turni, uno dalle 8 alle 13 e l'altro dalle 11 alle 16. In ogni sezione ci sono quindi due ore di compresenza. La giornata si suddivide in diversi momenti: l'accoglienza, momento di gioco libero, la colazione in contemporanea per le tre sezioni, le attività curricolari fino alla preparazione per il pranzo. Nel pomeriggio si alternano momenti di gioco libero, di attività organizzate e attività didattiche di rilassamento.

⁴ Per la descrizione della scuola vedere paragrafo 1.2 a pag. 4.



Figura 1 - Aula della sezione dei 4 anni della Scuola dell'Infanzia

2.2 T1

Sotto consiglio del Dirigente Scolastico, ho iniziato il mio T1, il primo tirocinio presente al secondo anno del corso di studi, nella sezione dei tre anni, in previsione di un possibile percorso da svolgere nella stessa classe negli anni successivi. Le ore totali da svolgere erano 65, divise tra i due ordini di scuola in modo diverso: 30 ore nella scuola dell'infanzia e 35 nella scuola primaria. Gli obiettivi generali del T1 prevedono la conoscenza dell'organizzazione della sezione nella scuola dell'infanzia, tramite attività di osservazione, durante le ore scolastiche e una riflessione sull'esperienza. Oltre alla conoscenza delle docenti e degli alunni e alla osservazione dello svolgimento delle attività didattiche condotte nella sezione, la mia Tutor Scolastica mi ha guidato nella scoperta dell'ambiente scolastico, del PTOF, mi ha illustrato documenti quali il registro di classe, della mensa e la progettazione curricolare della scuola, basata sui campi di esperienza contenuti nelle *Indicazioni Nazionali*, compresi i progetti in corso. L'osservazione durante l'annualità di tirocinio è stata documentata con protocolli osservativi e griglie consegnate dal tutor universitario, in base ai quali è stato

possibile seguire un percorso di conoscenza il più completo possibile. Di seguito un protocollo osservativo utilizzato durante una lezione di gioco/attività motoria.

PROTOCOLLO OSSERVATIVO

Osservatore: **CLAUDIA GIANNELLI**

Data Osservazione: **14/04/2016**

Ora Inizio: **10.00** Ora Fine: **10.30** (15-20 minuti circa)

1 - Luogo/Contesto: **Salone/palestra della scuola dell'infanzia**

2 - Oggetto dell'osservazione: **Attività motoria, il percorso a ostacoli per sviluppare e allenare coordinazione e equilibrio.**

3 - Finalità: **Conoscere le modalità con cui l'insegnante prepara, spiega e fa eseguire un percorso ai bambini di 3 anni.**

4 - Modalità: **Osservazione passiva**

5 - STESURA (INDIVIDUALE)

DESCRIZIONE ELEMENTI	DEGLI	IMPRESSIONI/VALUTAZIONI/ INTERPRETAZIONI
L'insegnante ha disposto i materiali al centro del salone mentre i bambini stavano seduti sulle panchine. L'insegnante ha mostrato come eseguire il percorso; gli alunni uno alla volta lo hanno ripetuto. Alcuni bambini non sono riusciti a superare		I bambini hanno osservato il percorso mentre lo eseguiva l'insegnante, ma alcuni stavano seduti per terra e parlavano tra loro. Un bambino si è alzato ed ha iniziato a correre per il salone; alcuni bambini lo hanno seguito. L'insegnante lo ha richiamato ma lui non

tutti gli ostacoli senza il sostegno della maestra. Hanno trovato difficoltà soprattutto nei salti a piedi uniti.	ascoltava. Ha eseguito il percorso per ultimo. Alcuni bambini cercavano come sostegno la mano dell'insegnante.
<p>6 - DISCUSSIONE DELL'OSSERVAZIONE (INDIVIDUALE/COLLETTIVA)</p> <p>L'osservazione individuale è avvenuta in compresenza insieme all'insegnante della classe, ma senza alcun intervento diretto. La presenza di una persona in più nel gruppo non ha rotto l'equilibrio della classe, che ha svolto la lezione come qualsiasi altro giorno. Non ci sono state particolari criticità se non qualche bambino che ha disturbato gli altri mentre cercavano di eseguire il comando dato, attirando l'attenzione. L'insegnante è riuscita comunque a mantenere l'ordine e a fare in modo che tutti eseguissero il percorso nel migliore dei modi.</p>	

2.3 T2

Durante il terzo anno del corso di studi, ho svolto il tirocinio 2, come già spiegato, nello stesso plesso e nella stessa sezione dell'anno precedente e sono stata seguita dalla stessa Tutor Scolastica. Al contrario, nell'anno accademico sono avvenuti dei cambiamenti per quanto riguarda l'organizzazione del progetto di tirocinio ed è cambiato anche il Tutor Universitario assegnato al mio gruppo; questo ha implicato dei cambiamenti gestionali. Sono stati introdotti due nuovi strumenti: il Registro di Tirocinio⁵, contenente le attività necessarie per il corretto svolgimento del tirocinio, compilabile dallo studente, dal Tutor Universitario, dai Tutor Scolastici e dal D.S. dell'istituto ospitante; il Quaderno di Lavoro⁶, strumento operativo utilizzato dallo studente per documentare, in modo sia descrittivo che critico, l'esperienza di tirocinio diretto nelle scuole. Nel T2 le ore complessive aumentano a 105, di cui 40 da svolgere nella scuola dell'infanzia. Gli obiettivi

⁵ Il Registro di Tirocinio è stato attualmente rimosso in quanto i suoi contenuti sono stati integrati nel nuovo Quaderno di Lavoro.

⁶ Per avere più informazioni e visionare il Quaderno di Lavoro consultare la pagina raggiungibile al seguente link: <http://www.qualitaformazionemaestri.it/index.php/tirocinio/documenti-e-modelli>.

invece sono: osservare e analizzare le diverse attività didattiche nella scuola; conoscere le modalità di progettazione e valutazione dei risultati; coadiuvare attivamente alcuni momenti dell'attività didattica; riflettere sull'esperienza. Svolgendo il tirocinio in una classe che già conoscevo, ho potuto concentrarmi di più sugli aspetti didattici e curricolari, avendo già instaurato un rapporto sia con il team docenti che con i bambini. Essendo stati introdotti dei task specifici per ogni annualità di tirocinio, le attività da svolgere sono state definite nel Registro e documentate nel Quaderno di Lavoro o tramite specifiche schede di osservazione predisposte dal Tutor Universitario. È stato educativo e interessante poter partecipare per la prima volta a incontri collegiali, ma soprattutto avere un ruolo attivo nella didattica. Per quanto riguarda gli incontri collegiali, è stato opportuno documentare ogni riunione redigendone il verbale, mentre per le altre attività da svolgere mi sono servita delle schede descrittive consegnate dal Tutor Universitario. I verbali che ho potuto scrivere sono stati di intersezione tecnica per l'infanzia e di interclasse per la scuola primaria. Nel Quaderno di Lavoro inoltre è stata introdotta la Scheda di Progettazione dell'Attività/Lezione, da completare in funzione dell'intervento didattico da svolgere nella classe. In questo anno ho svolto l'attività nella scuola dell'infanzia, preparando un cartellone interattivo ed una storia sul letargo. Nonostante non fosse la prima esperienza con i bambini, è stato emozionante ed anche un po' difficile organizzare e gestire, pur con il sostegno dell'insegnante della sezione, un intervento didattico in autonomia. I dubbi all'inizio sono moltissimi e la paura di sbagliare influisce molto sulla capacità di autocontrollo e sulla sicurezza. Grazie al supporto della Tutor comunque sono riuscita a svolgere al meglio questa prima prova generale.

IL LETARGO

FINALITÀ: Il bambino ascolta e comprende narrazioni, racconta e inventa storie, chiede e offre spiegazioni, usa il linguaggio per progettare attività e per definirne regole. Osserva con attenzione il suo corpo, gli organismi viventi e i loro ambienti, i fenomeni naturali, accorgendosi dei loro cambiamenti. Individua le posizioni di oggetti e persone nello spazio, usando termini come avanti/dietro, sopra/sotto, destra/sinistra, ecc; segue correttamente un percorso sulla base di indicazioni verbali.

<p>OBIETTIVO DIDATTICO: Il bambino arricchisce il patrimonio linguistico, apprende il significato della parola "letargo"; comprende un testo narrativo, riconosce e comprende i mutamenti e i fenomeni legati alle stagioni; conosce i rapporti topologici sopra/sotto, in alto/in basso, aperto/chiuso, dentro/fuori, vicino/lontano, riconosce la possibilità di interazione col cartellone e colloca gli elementi grafici al loro posto.</p>
<p>STRUMENTI E MODALITÀ DI VERIFICA: Elaborazione individuale autonoma o rispondendo domande stimolo; rappresentazione grafica della storia ascoltata, di una parte o di un elemento di essa tramite disegno individuale.</p>
<p>STRATEGIE: L'attività prevede il lavoro di gruppo e il lavoro individuale. L'insegnante introduce l'argomento chiedendo ai bambini se conoscono il significato della parola letargo, quando avviene e che cosa succede in questo periodo dell'anno (l'autunno). Partendo da un cartellone interattivo e da alcuni personaggi, creati appositamente, l'insegnante legge una storia sul letargo, coinvolgendo i bambini per esempio chiedendo il significato di una parola, descrivere un animale, o mimare una azione. Durante la lettura della storia a turno i bambini vengono chiamati per inserire i personaggi al loro posto nel cartellone. Poiché i personaggi sono in numero inferiore ai bambini, dopo la lettura e rielaborazione della storia viene consentito anche a chi non lo aveva fatto di sistemare i personaggi nel cartellone. Per l'esecuzione del disegno individuale ai bambini viene chiesto di disegnare a piacere un elemento della narrazione ascoltata. I bambini vengono aiutati con dei disegni come esempio per rappresentare gli animali più difficili. I bambini che si trovano in difficoltà vengono seguiti dalle insegnanti.</p>
<p>TEMPI: L'attività si svolge durante l'ora di attività didattica, dalle 10.30 alle 11.45 circa.</p>
<p>PREDISPOSIZIONE DELL'AMBIENTE FISICO (luce, organizzazione spazi, arredi ...): La prima parte dell'attività si svolge nell'angolo della conversazione, in modalità Circle Time; davanti ai bambini è posto il cartellone sul quale si sviluppa la storia e con il quale i bambini interagiscono. La seconda parte, che comprende il lavoro individuale, si svolge ai tavoli, sui quali sono sistemati fogli bianchi e pennarelli colorati.</p>
<p>RIDUZIONE DI EVENTUALI FATTORI DI DISTURBO: Per evitare disturbi dovuti al comportamento dei bambini, l'insegnante chiede loro di sedersi alternando un maschio e una femmina.</p>
<p>DESCRIZIONE DELL'ATTIVITÀ: L'insegnante, dopo aver sistemato i bambini in semicerchio nell'angolo della conversazione, richiama la loro attenzione facendo domande riguardanti il periodo dell'anno in cui si trovano in quel momento (il mese, il clima...). Dopo sistema un cartellone con finestrelle</p>

davanti ai bambini in modo che lo possano vedere tutti: è rappresentato l'ambiente del bosco dove sono presenti delle finestrelle che fungono da tane; in ogni finestrella è collocato un pezzo di feltro per attaccarci i rispettivi animali abitanti. Gli altri personaggi sono ritagliati singolarmente e durante il racconto della storia l'insegnante li mostra ai bambini in modo che li riconoscano e che possano collocarli nella giusta tana. I bambini vengono chiamati a turno per svolgere questo compito.

L'insegnante racconta quindi, aiutandosi con questi materiali, la storia di un piccolo riccio che non vuole andare in letargo e che vuole restare a giocare, ma non trova nessun amico poiché tutti gli animali del bosco stanno preparandosi per rifugiarsi nelle loro tane.

Nel corso del racconto non tutti i bambini riescono a collocare un personaggio nel cartellone, così l'insegnante, quando ha finito la storia, stacca gli animali e dà la possibilità anche a chi non lo aveva fatto di reinserirli.

Dopo l'insegnante invita i bambini a rielaborare il racconto e durante la conversazione formula alcune domande, alle quali devono rispondere individualmente e autonomamente, per sollecitare e stimolare gli alunni come: Chi sono i personaggi del racconto? Perché vanno in letargo? Che cosa fanno gli animali? Cosa rispondono al piccolo riccio? Dove va a dormire l'orso? E lo scoiattolo?

Dopo l'insegnante invita i bambini a sedersi ai tavoli, consegna loro un foglio a testa e chiede disegnare una parte della storia o un elemento che si ricordano.



2.4 T3

Il terzo anno di tirocinio, per mantenere una continuità, sono stata assegnata dal Dirigente Scolastico alla stessa sezione della Scuola dell'Infanzia di San Lino; questo mi ha permesso di compiere un'osservazione diretta sui progressi e la crescita dei bambini. Poiché l'insegnante che negli anni passati mi ha seguito questo anno ha cambiato classe, sono stata assegnata ad una nuova Tutor Scolastica. Le ore complessive del terzo anno di tirocinio ammontano a 120 ore, di cui 50 da svolgere nella scuola dell'infanzia. Gli obiettivi invece sono: progettare e condurre azioni didattiche (modello MARC⁷), comparare e differenziare esperienze didattiche, riflettere sul proprio discorso formativo. Nella terza sezione, i bambini avevano già 5-6 anni e si stavano preparando all'ingresso nella scuola primaria. In questo anno ho potuto osservare tutte le attività di pregrafismo che sono state affrontate in previsione dell'apprendimento della lettura e della scrittura, come i bambini imparano a scrivere le prime parole e come giocano con esse; inoltre ho potuto assistere ad alcune lezioni di Lingua Inglese che venivano svolte nelle sezioni da una insegnante del plesso. L'attività in cui ho avuto modo di esercitarmi e migliorare è stata quella della lettura espressiva, nella quale ho sempre avuto più difficoltà. Durante questi momenti, nonostante la Tutor Scolastica mi abbia confermato la buona riuscita dell'intervento didattico, ho osservato alcune criticità, che mi hanno fatto porre l'attenzione sulla stretta relazione tra la lunghezza dell'attività di ascolto e la capacità di attenzione dei bambini. La scelta del brano da leggere si è rivelata inadeguata, essendo molto lungo, quindi i bambini hanno mantenuto con difficoltà l'attenzione. Ho deciso sul momento, osservando la risposta della classe, di interrompere la lettura e iniziare a lavorare solo sulla prima parte, per poi concluderla il giorno seguente. Un'altro aspetto importante è stato il poter osservare alcune modalità di screening che sono state eseguite nella sezione, interessante per capire come operare nel contesto classe per una prevenzione dei principali disturbi linguistici e non solo, che potrebbero presentarsi nel futuro. Come richiesto dai task specifici del nuovo Quaderno di Lavoro, ho progettato e condotto insieme all'insegnante curricolare

⁷ Per maggiori dettagli vedere paragrafo 3.4, pag. 21.

un'attività di gioco, con l'obiettivo di consolidare e il riconoscimento dei concetti di destra e sinistra.

Attività di progettazione	
IL GIOCO	
Studentessa: Claudia Giannelli	
Matricola: 5775311	
Scuola: Infanzia	
Classe: 5 anni	
Alunni: 25 bambini	
Obiettivi: Consolidare il riconoscimento dei concetti di destra e sinistra	
Materiali: 15 cerchi sistemati per terra che formano un reticolato; tre cartellini a forma di cerchio di tre colori: verde (passo avanti), rosso (girare a destra), giallo (girare a sinistra); filo di lana colorato rosso e giallo.	
Spazi: Il salone della scuola	
Tempi: 40 minuti circa	
Attività	
Ai bambini vengono legati ai polsi dei nastri colorati: un nastro rosso al polso destro e uno giallo al polso sinistro. Un bambino alla volta viene chiamato davanti al reticolato. L'insegnante mostra uno alla volta un cartellino colorato: quando mostra il verde il bambino deve fare un passo avanti nel cerchio successivo; quando mostra il giallo deve voltarsi a sinistra, invece quando mostra il rosso deve voltarsi verso destra. Questo avviene fino a quando il bambino ha eseguito un percorso seguendo i comandi, per uscire dal reticolato. L'esercizio viene svolto da tutti i bambini della classe, uno alla volta.	



2.5 T4

L'ultimo anno di tirocinio sono stata inserita nella sezione dei 4 anni, di nuovo a fianco della stessa insegnante che mi ha seguita durante il T1 e il T2 all'infanzia. Questa è stata per me l'esperienza forse più completa, faticosa e gratificante di tutta la mia carriera universitaria, in cui mi sono arricchita sia dal punto di vista professionale che umano. Le ore svolte nella scuola dell'infanzia sono state 80; gli obiettivi del T4 sono gli stessi del T3, è stato importante quindi in queste due ultime annualità il lavoro di progettazione e di conduzione di interventi didattici per poi effettuare una attenta analisi di riflessione sul percorso svolto. Ho avuto occasione di partecipare ad una grande parte delle attività didattiche e dei progetti che sono stati svolti durante l'anno. Nonostante fossi assegnata alla seconda sezione, delle circostanze interne alla scuola hanno fatto sì che prendessi parte ad attività anche con altre sezioni, soprattutto quella dei 3 anni. Le insegnanti mi hanno considerata e trattata come una di loro, così i bambini, anche se non mi conoscevano, mi hanno fin da subito rispettata oltre ad avermi accolta con molto affetto. Ho partecipato a diversi progetti tra i quali "Andiamo alla Biblioteca Comunale", un progetto interessante al quale ho preso parte attiva. È iniziato con la lettura di un libro scritto da una autrice di zona, che racconta la storia di un porcellino, statua situata su di una delle case-torri della città di Volterra, che

aiutato dal vento vola sulla città visitandone i luoghi più caratteristici. Al lavoro in classe si sono aggiunte le visite in biblioteca, dove è avvenuta la lettura del libro da parte dell'autrice stessa e le visite nei luoghi citati, per ripercorrere le tappe del breve viaggio del porcellino e sperimentare le stesse emozioni. Un'altra esperienza che ho fatto è stata quella del progetto di "Musicologia", tenuto da un esperto esterno, maestro di musica. Purtroppo per questo motivo ho potuto solo osservare le attività, ma sono state interessanti e sicuramente ne farò tesoro per il futuro. Durante il tirocinio, insieme all'insegnante assegnata al potenziamento, ho ipotizzato delle attività da effettuare nella sezione riguardo alla sillabazione. In questa attività i bambini hanno giocato con il proprio nome, imparando a dividerlo saltando dentro a dei cerchi colorati sistemati in fila sul pavimento e pronunciando una sillaba ad ogni salto; inoltre hanno imparato ad usare il cosiddetto "Sillabometro", strumento costruito appositamente per "vedere" in quante parti si suddivide una parola e calcolarne la lunghezza.

Capitolo 3

Il tirocinio nella scuola primaria

3.1 I locali e l'organizzazione della Scuola Primaria

Il tirocinio nella scuola primaria si è svolto per i quattro anni consecutivi con la stessa Tutor Scolastica, con la quale ho accompagnato due sezioni dalla classe seconda fino alla quinta. Il percorso, come per la scuola dell'infanzia, mi è stato suggerito dal Dirigente Scolastico, vista anche la disponibilità dell'insegnante nell'accogliermi e nel guidarmi nel mio percorso di tirocinio. La scuola Primaria di San Lino, come descritto, si sviluppa al primo piano di un vecchio convento, per questo le aule sono molto grandi e molto luminose. Le classi in cui sono stata ospitata sono rimaste per tutto il percorso scolastico nelle stesse aule, che sono arredate in modo molto simile: i banchi sono disposti a file o a coppie rivolti verso la lavagna di ardesia, le cattedre sono posizionate in cima alla classe e alle pareti sono appesi cartelloni e cartine geografiche. Nella scuola, il primo anno era presente una sola LIM, sistemata nell'Aula Magna, in modo che tutte le classi potessero accedervi al bisogno. Dall'anno successivo, grazie ai finanziamenti Europei ricevuti tramite i progetti PON⁸, la scuola è stata fornita di 6 LIM (divisi nelle varie classi) e di un computer portatile per ogni sezione. Questo è stato un grande passo avanti per la didattica, in quanto questi ausili multimediali sono un valido aiuto nell'apprendimento, soprattutto degli alunni con bisogni educativi speciali. Inoltre è stata finanziata l'installazione della rete WIFI in tutto il plesso. La scuola offre un orario di 28 ore e trenta minuti alla settimana, con un rientro pomeridiano per ogni classe. Le lezioni iniziano alle ore 8 e terminano alle ore 13 oppure alle 16.30 nei giorni del tempo prolungato. La pausa mensa dura un'ora e 30.

⁸ Il Programma Operativo Nazionale (PON) del Ministero dell'Istruzione, dell'Università e della Ricerca, intitolato "Per la Scuola - competenze e ambienti per l'apprendimento", finanziato dai Fondi Strutturali Europei, contiene le priorità strategiche del settore istruzione e ha una durata settennale, dal 2014 al 2020. Per maggiori dettagli consultare la pagina web al seguente link: <http://www.istruzione.it/pon/ilpon.html>



Figura 2 - Aula della sezione 4B della Scuola Primaria

3.2 Descrizione delle classi

Nelle due sezioni in cui ho svolto il tirocinio, nel corso degli anni sono avvenuti molti cambiamenti, sia per quanto riguarda alcuni trasferimenti degli alunni, sia per le certificazioni dei bambini con BES che, non potendo essere dichiarati tali in classe seconda, sono giunte successivamente. Le insegnanti curricolari sono in tutto 4, suddivise per materia: una insegnante di Italiano, Arte e Immagine e Musica, una di Storia, Geografia e Attività Motoria, una di Matematica, Scienze e Inglese e una di Religione Cattolica. In ogni classe è presente inoltre una insegnante di sostegno in quanto nella sezione A è presente una bambina con Sindrome di Asperger ed un bambino con disturbi del linguaggio, mentre nella B è presente un bambino con un ritardo cognitivo. Dalla classe terza, sono stati certificati altri tre alunni con disturbi specifici dell'apprendimento e nella sezione A è stata diagnosticata ad un bambino la sindrome di Tourette. Gli alunni nella sezione A sono rimasti per tutta la durata della scuola primaria 20, mentre nel corso degli anni il numero di alunni nella sezione B è cambiato, passando da 22 nella classe seconda a 21 in terza, a causa del trasferimento di un bambino e a 23 in quinta, grazie all'arrivo di due bambini stranieri.

3.3 L'esperienza in classe: T1 e T2 nella scuola primaria

In questi quattro anni ho potuto svolgere un percorso scolastico insieme alle classi in cui sono stata inserita, allo stesso modo, o quasi, delle insegnanti. La relazione con il gruppo classe e con i docenti si è instaurata fin dal primo anno, nonostante il mio ruolo a scuola fosse molto marginale. Gli obiettivi del T1, come già spiegato infatti prevedono per il tirocinante solamente attività di osservazione. Le ore svolte in questa annualità sono state 35. Ho passato la maggior parte del tempo con la mia tutor, insegnante di italiano, esaminando attentamente le sue azioni, il suo modo di lavorare e di gestire la classe. L'ho affiancata durante le attività più significative, oltre alla lezione frontale, come il dettato, la correzione dei compiti, la preparazione per le prove invalsi. Durante questo anno, essendo i bambini in seconda, ho potuto osservare anche l'esecuzione di una prova di screening. Uno dei progetti a cui ho assistito e che per me è stato significativo è quello in collaborazione con la Società della Salute e la ASL del territorio. Questa collaborazione a livello di Istituto viene riproposta ogni anno e vuole promuovere un atteggiamento positivo e di accettazione verso la diversità. Durante l'orario scolastico infatti vengono accolti nelle classi alcuni ragazzi e adulti con disabilità, con i quali i bambini condividono alcune attività laboratoriali, di solito lavoretti manuali, in base al tema del progetto, scelto durante l'anno. In questo caso oggetto di studio era la fiaba *"Storia di una gabbianella e del gatto che le insegnò a volare"* di Luis Sepúlveda.

L'esperienza del T2, essendo regolata da task specifici da svolgere indicati nel Registro di Tirocinio⁹, è stata densa di attività: lettura di un testo in classe, organizzazione di uno spazio in funzione di un gioco/attività, partecipazione e gestione di una discussione, partecipazione a incontri collegiali, redazione di un verbale; le ore svolte in classe sono state 65. Per documentare le esperienze svolte è stato utile utilizzare alcune schede osservative date dal Tutor Universitario e il Quaderno di Lavoro. La partecipazione all'incontro di Interclasse Tecnica Completa è stato molto interessante. Osservare per la prima volta una riunione degli insegnanti insieme ai genitori è stato infatti un'esperienza significativa per

⁹ Il documento è descritto nel paragrafo 2.3 a pag. 10.

capire le modalità di interazione delle due parti, il tipo di relazione instaurata e il modo in cui i docenti si avvicinano al gruppo genitori per un dialogo costruttivo.

3.4 La progettazione didattica nella scuola primaria e il modello M.A.R.C.

Gli ultimi due anni di tirocinio diretto sono stati i più significativi per quanto riguarda l'esperienza pratica. Gli obiettivi del T3 e T4 infatti prevedono la progettazione e la conduzione di azioni didattiche, che, come nella scuola dell'infanzia, ho eseguito anche alla scuola primaria, ma in questo caso ho scelto di eseguire la videoregistrazione prevista dal modello M.A.R.C.¹⁰. La sigla M.A.R.C. (Modellamento, Azione, Riflessione, Condivisione) indica la metodologia che si avvale della videoripresa di un intervento didattico, con lo scopo di poter essere riesaminata dal tirocinante stesso, dai tutor e da altri tirocinanti, per favorire comportamenti didattici di migliore qualità e di maggiore efficacia. L'esecuzione di questa metodologia in classe rende necessario l'utilizzo di liberatorie rivolte ai genitori degli alunni, che possono autorizzare o non autorizzare la pubblicazione del filmato da parte dell'Università. Nei casi più difficili, noi tirocinanti dobbiamo stare attenti e organizzare il lavoro facendo in modo che nel filmato non appaia nessun volto all'infuori dello studente stesso. Nel mio caso tutti i genitori, informati con un avviso dal D.S. e consapevoli che il lavoro sarebbe stato importante, hanno firmato la delibera autorizzando le registrazioni video. Un altro aspetto importante delle ultime due annualità di tirocinio è quello che pone l'attenzione sui BES (Bisogni Educativi Speciali). L'orario di tirocinio infatti prevede che 35 ore nel T3 (ore complessive 70) e 60 ore nel T4 (ore complessive 130) vengano dedicate all'inclusione, quindi svolte in classi con presenza di bambini con bisogni educativi speciali. Come possiamo leggere nei task specifici del quaderno di lavoro, è richiesta la rilevazione di abilità e difficoltà di alunni con BES, da effettuare svolgendo colloqui con i

¹⁰ L'acronimo MARC significa Modellamento, Azione, Riflessione, Condivisione e indica la metodologia circolare teorico-pratica, focalizzata sull'interazione maestro-bambino, basata sulla videoregistrazione di una lezione. Per maggiori informazioni consultare la pagina al seguente link: <http://www.qualitaformazionemaestri.it/index.php/tirocinio/marc>.

docenti e visionando PEI¹¹ e PDP¹² e differenziare almeno un'attività didattica in funzione della presenza di questi alunni, quindi progettare una attività inclusiva. Come ho descritto nella presentazione delle classi, nelle due sezioni erano presenti ben 4 alunni certificati dalla legge 104 e tre bambini con disturbi specifici dell'apprendimento. Data la molta varietà di disturbi e la mia poca esperienza, nonostante i molti studi effettuati all'università, non è stato facile riuscire a trovare un metodo che potesse andare bene per tutti, quindi ho deciso di prendere da esempio le insegnanti che conoscevano meglio ogni situazione e seguire le strategie già utilizzate nelle classi.

Per quanto riguarda il T3, l'attività progettata e svolta è stata inclusiva in quanto sono stati utilizzati metodi e strumenti inclusivi, scelti per facilitare l'apprendimento degli alunni con BES, ma anche per il resto della classe. Sono stati privilegiati metodi come il *Cooperative learning*¹³, il *Peer tutoring*¹⁴ e l'uso della LIM. L'attività didattica, inserita in un progetto più ampio sulla non violenza e il rispetto per le differenze, ha avuto lo scopo di far riflettere i bambini sugli stereotipi di genere; le attività sono state basate su alcuni testi tratti dal libro "*Storie della buonanotte per bambine ribelli*" di Francesca Cavallo e Elena Favilli. Il risultato è stato positivo in quanto nei gruppi, scelti dalle insegnanti, c'è stata collaborazione e intesa e le metodologie scelte hanno permesso il confronto tra pari e l'aiuto reciproco. I bambini con BES, infatti, che potevano avere difficoltà in alcune attività come l'esposizione orale, hanno ricevuto il supporto dai compagni, che li hanno sostenuti e non sono stati costretti a sostenere situazioni che potevano creare loro disagio in quanto hanno potuto scegliere il ruolo da svolgere nel gruppo. Di seguito la scheda di progettazione dell'attività.

¹¹ Piano Educativo Individualizzato: documento che si redige in presenza di alunni con disabilità certificata dalla legge 104/92. Definisce un percorso individualizzato che può svolgere l'alunno per potenziare determinate abilità o per acquisire specifiche competenze.

¹² Piano Didattico Personalizzato: documento che si redige in presenza di alunni che presentano disturbi non certificati con legge 104, da, come disturbi specifici dell'apprendimento, ADHD o svantaggio sociale ed economico. Definisce un percorso personalizzato per quanto riguarda l'offerta didattica, le strategie da utilizzare e le modalità relazionali, sulla specificità dei bisogni educativi che caratterizzano l'alunno.

¹³ Metodologia di insegnamento attraverso la quale gli studenti lavorano e apprendono in piccoli gruppi, aiutandosi reciprocamente (dal sito: http://www.edscuola.it/archivio/comprendivi/cooperative_learning.htm).

¹⁴ Strategia educativa che favorisce un passaggio spontaneo di conoscenze da pari a pari (dal sito <https://www.tecnicadellascuola.it/che-cos-e-il-peer-tutoring>).

COMBATTIAMO GLI STEREOTIPI DI GENERE
ATTIVITÀ SVOLTA NELLA SCUOLA PRIMARIA
FINALITÀ: L'attività didattica si colloca nel progetto "Noi scegliamo la non violenza", che la classe sviluppa in collaborazione con l'università di Bari ed ha la finalità di orientare gli interventi educativi e didattici in un orizzonte di senso fondato su nuclei basilari quali la dignità della persona, il rispetto della libertà, il rispetto delle differenze, la prevenzione della violenza contro il diverso da sé e in particolare verso le donne. Inoltre ha lo scopo di far riflettere i bambini sugli stereotipi di genere, per iniziare a mettere in discussione alcuni modelli maschili e femminili che la società impone ed educarli verso principi di uguaglianza e pari opportunità.
OBIETTIVO DIDATTICO: Produrre testi con idee e riflessioni personali, espressi logicamente e ordinatamente, con lessico appropriato e ortograficamente corretti. Rispettare i turni di parola negli scambi comunicativi. Intervenire con pertinenza ed esprimere il proprio punto di vista in conversazioni e discussioni. Organizzare un semplice discorso orale su un tema affrontato in classe con un breve intervento preparato in precedenza o un'esposizione su un argomento di studio utilizzando una scaletta. Usare, nella lettura di vari tipi di testo, opportune strategie per analizzare il contenuto; porsi domande all'inizio e durante la lettura del testo; cogliere indizi utili a risolvere i nodi della comprensione.
STRUMENTI E MODALITÀ DI VERIFICA: Gli strumenti utilizzati sono la LIM, materiale scolastico e un libro contenente molte storie su donne famose. La verifica avviene in itinere e al termine dell'attività sui lavori finali dei bambini che andranno ad allestire una "Galleria delle donne famose". Inoltre la valutazione avviene anche sull'esposizione a gruppi del lavoro svolto al resto della classe.
STRATEGIE: Lezione di didattica laboratoriale e lavoro a gruppi (Cooperative Learning).
TEMPI: La lezione colloca nella parte finale del progetto (gennaio-marzo), per la durata di 4 ore divise in due giornate.
PREDISPOSIZIONE DELL'AMBIENTE FISICO (luce, organizzazione spazi, arredi ...): La lezione si svolge in classe con l'uso della LIM. I tavoli sono disposti a isole e formano cinque gruppi di 4 bambini ciascuno.

RIDUZIONE DI EVENTUALI FATTORI DI DISTURBO: I gruppi sono stati scelti dalle insegnanti tenendo presente le varie tipologie di alunni che compongono la classe, quindi anche per favorire l'inclusività.

SVOLGIMENTO DELL'ATTIVITÀ (AVVIO, SVOLGIMENTO, CONCLUSIONE):

Prima parte

L'insegnante scrive alla lavagna la domanda "Cosa vorrei fare da grande" e invita i bambini a rispondere a turno. Vengono scelti insieme due colori, uno per i lavori dei maschi e uno per quelli delle femmine. I bambini vengono invitati a pensare alle reali differenze che ci sono tra i mestieri scelti e a ragionare sul perché ci siano "lavori da maschi" e "lavori da femmine". Le riflessioni vengono scritte sul quaderno in un discorso costruito dagli stessi alunni.

Seconda parte

I bambini lavorano a gruppi. L'insegnante consegna ad ogni gruppo una storia di una donna famosa ed il disegno del ritratto del personaggio. Inoltre dà il comando del lavoro che ogni gruppo deve svolgere: devono leggere la storia del personaggio e individuarne le principali caratteristiche per scrivere la "carta d'identità". Poi devono scrivere una riflessione collettiva su quello che li ha colpiti di più di quel personaggio e il perché.

Terza parte

Ogni gruppo presenta alla classe il proprio personaggio, esponendo il lavoro svolto. Durante l'attività i bambini sono invitati a disegnare il ritratto della donna famosa che avevano assegnata, che servirà, insieme alla carta d'identità e alle riflessioni, ad allestire una "Galleria delle donne famose" nei corridoi della scuola.

Un altro task previsto nel T3 è stato guidare un'attività di comprensione del testo. Nell'ambito di un progetto sull'alabastro, è stato proposto il primo capitolo del libro "Paura e tristezza" di Carlo Cassola. Il testo in questione contiene descrizioni dettagliate di personaggi, per la precisione alabastrai, del loro lavoro e del paesaggio, espresse con un gergo popolare.

Per quanto riguarda gli incontri collegiali, nel corso del tirocinio ho partecipato al Collegio Docenti unitario.

L'ultimo anno di tirocinio diretto è stato impegnativo ed intenso. Le ore complessive da svolgere erano 210, di cui 130 alla scuola primaria. I mesi trascorsi nelle classi quinte della scuola di San Lino, conosciute ormai da 4 anni, sono stati molto emozionanti, visto che sono stati i mesi conclusivi del lungo e bellissimo percorso che è stato quello nella scuola primaria: per me, perché questo

anno è stato l'anno conclusivo dei miei studi, per i bambini, che intraprenderanno un nuovo viaggio nella Scuola Secondaria di Primo Grado. Durante il tirocinio ho affiancato la mia Tutor Scolastica e le altre insegnanti in ogni aspetto dell'insegnamento, sia per quanto riguarda la didattica, che nella parte burocratica e organizzativa. Sono state molte le attività a cui ho avuto la possibilità di partecipare attivamente, dalle interrogazioni, alla correzione dei compiti, alle letture collettive in classe e molte attività laboratoriali. Per quanto riguarda le ore da dedicare all'inclusione, ho svolto diversi ruoli nelle classi affiancando gli alunni che ne avevano più bisogno. Oltre all'attività di sostegno, ho potuto svolgere, sotto il controllo delle insegnanti curricolari, alcuni interventi didattici, differenziandoli in funzione dei bisogni specifici dei bambini. L'attività che ha avuto più successo ed approvazione tra gli alunni si è svolta in occasione del ripasso/interrogazione di Storia: la proposta è stata quella di dividere la classe a squadre per un torneo in cui ogni gruppo avrebbe dovuto sfidare quello avversario in una gara di domande e risposte, formulate dagli stessi alunni. Il gioco è stato svolto nella sezione 5B, per consolidare gli apprendimenti, ma anche e soprattutto per coinvolgere i bambini stranieri, i quali hanno dimostrato di avere difficoltà ad esprimersi non per la scarsa conoscenza della lingua, ma per la timidezza. Infatti, giocare insieme stimola e rafforza le relazioni del gruppo classe, in un'ottica di condivisione a aiuto reciproco. Sempre per quanto riguarda i bambini stranieri, ho potuto prendere parte nella scelta e somministrazione delle prove per la valutazione degli apprendimenti. Essendo il bambino in questione di madrelingua tedesca, è stato necessario preparare delle prove semi strutturate, con il giusto bilanciamento tra esercizi di grammatica e di lessico e fornite di esempi per facilitare la comprensione. Nel mese di dicembre, ho potuto assistere ai colloqui con i genitori. Gli incontri si sono svolti in classe, con le insegnanti riunite che convocavano una famiglia alla volta. Partecipando da spettatrice ho osservato attentamente i modi e gli aspetti principali su cui porre l'attenzione quando si parla dell'andamento scolastico di un alunno, sia dal punto di vista del rendimento che comportamentale.

In questo ultimo anno ho deciso di fare la videoregistrazione della lezione nella classe 5A, la classe provvista della LIM. Dopo aver deciso insieme

l'argomento, la mia tutor mi ha dato completa libertà nella progettazione dell'attività; la lezione sui verbi transitivi e intransitivi si è sviluppata a partire dall'attivazione delle preconoscenze (in questo caso la forma attiva e passiva), con la spiegazione avvenuta in forma di dialogo, con il supporto della LIM, fino a concludersi con degli esercizi, per verificare la comprensione dell'argomento. Di seguito la scheda di progettazione dell'attività.

I VERBI TRANSITIVI E INTRANSITIVI
LEZIONE SVOLTA NELLA SCUOLA PRIMARIA
OBIETTIVO DIDATTICO: Conoscere la differenza tra i verbi transitivi e intransitivi e imparare a riconoscerli nelle frasi.
STRUMENTI E MODALITÀ DI VERIFICA: Verifica in itinere. Utilizzo di esercizi per la verifica delle conoscenze acquisite.
STRATEGIE: Lezione frontale dialogata.
TEMPI: Lezione della durata di due ore circa.
PREDISPOSIZIONE DELL'AMBIENTE FISICO (luce, organizzazione spazi, arredi ...): La lezione si svolge in classe, con l'uso della LIM I bambini sono disposti a ferro di cavallo.
RIDUZIONE DI EVENTUALI FATTORI DI DISTURBO: In supporto alla classe, poiché sono presenti alcuni alunni con BES, è presente l'insegnante di sostegno.
FASI DELL'ATTIVITÀ/LEZIONE (AVVIO, SVOLGIMENTO, CONCLUSIONE): La lezione si avvia con una lettura in classe da parte dell'insegnante. La storia è tratta da un libro che utilizza delle storie per l'insegnamento di regole grammaticali e il capitolo letto riguarda i verbi

transitivi e intransitivi. In questo modo viene catturata l'attenzione degli alunni e vengono attivate anche le preconoscenze (forma attiva e passiva).

Successivamente viene svolto un piccolo esercizio di comprensione in cui i bambini devono scegliere quali sono le frasi giuste relative a ciò che hanno ascoltato.

L'insegnante scrive due frasi differenti alla lavagna facendosi consigliare dai bambini, a cui viene poi chiesto, a voce, di farne l'analisi logica, per fare in modo che si focalizzino sulla presenza o meno del complemento oggetto. A questo punto la regola grammaticale viene formulata collettivamente e scritta sul quaderno. L'insegnante consegna degli esercizi da svolgere singolarmente per verificare se l'argomento è stato compreso.

In supporto ai bambini con BES l'insegnante ha preparato delle schede semplificative in cui le regole importanti sono evidenziate e per ogni regola viene riportato un semplice esempio.

Capitolo 4

La scuola come comunità professionale

4.1 L'organizzazione scolastica e le modalità di lavoro collegiale

Durante i 4 anni di tirocinio ho avuto la possibilità di partecipare alla vita della scuola con un coinvolgimento totale, osservando e vivendo i vari aspetti non solo della didattica, ma anche dell'organizzazione a livello di classe e di istituto. Fin dal primo anno ho capito che la scuola è come un organismo, che funziona quando tutte le sue singole parti lavorano interagendo nel modo giusto. Purtroppo non sempre questo succede. Infatti le relazioni che si creano nell'ambiente scolastico sono molte: la più chiara e esplicita è quella tra docenti e alunni, subito dopo abbiamo quella tra docenti e custodi e tra bambini e custodi; è evidente anche la relazione tra insegnanti e genitori. Molto meno visibile ma importante è poi il personale scolastico con ruolo amministrativo, ovvero della Segreteria. Avendo il compito di gestione e organizzazione di tutto il funzionamento scolastico, il rapporto tra il corpo docente e gli uffici è indispensabile e fondamentale. In questi anni tutti i miei progetti di tirocinio sono stati accettati e gestiti dalla segreteria, che ha regolato i rapporti tra istituto e Università, quindi anche io ho potuto conoscere il personale amministrativo e capire l'importanza del lavoro che svolge. Questi uffici non gestiscono solamente il lavoro dei docenti, ma anche i rapporti tra i genitori e la scuola. Per ultimo ma non meno importante il ruolo del Dirigente Scolastico. Durante tutto il tirocinio ho potuto instaurare un rapporto professionale con il D.S. dell'istituto, il suo supporto e approvazione sono stati fondamentali per poter andare avanti nel mio percorso. Una relazione serena e sincera apre al dialogo e questa è stata l'impressione che ho avuto nella mia esperienza, l'intenzione di creare legami solidi su cui contare sempre, per garantire professionalità ed efficacia. Purtroppo è molto facile che la situazione sfugga di mano quando ci sono molte persone che la pensano in modo diverso, quindi è stato frequente dover assistere a discussioni nate da malintesi, soprattutto per quanto riguarda l'idea di gestione della classe tra i docenti. Durante le riunioni a cui ho partecipato, gli incontri di interclasse e intersezione, ma anche durante le

programmazioni, ho potuto notare a volte un andamento negativo, dato dalla scarsa capacità di comprensione che alcuni docenti hanno dimostrato di fronte a proposte e scelte di altri. In generale comunque queste situazioni vengono gestite e superate cercando di rispettare le opinioni di tutti e trovando soluzioni in comune accordo.

Nel corso del tirocinio ho potuto partecipare al Collegio Docenti, ovvero la riunione plenaria di tutti i docenti dell'istituto che si tiene periodicamente nell'Aula Magna del plesso di San Lino. L'incontro è tenuto dal Dirigente Scolastico, affiancato dal Vicario, che illustrano l'ordine del giorno per discuterne e ascoltare eventuali opinioni degli insegnanti. Durante il Collegio vengono prese delle decisioni a livello di Istituto e ognuno può esprimere accordo o disaccordo rispetto agli argomenti trattati. Nella riunione a cui ho presenziato, non ho notato una completa partecipazione dei docenti, che per la maggior parte hanno ascoltato passivamente ciò che veniva discusso. Solamente alcuni hanno espresso il proprio parere su argomenti che li riguardavano da vicino.

Le due esperienze nella scuola dell'infanzia e nella scuola primaria per me sono risultate abbastanza diverse, soprattutto per quanto riguarda il modo di lavorare. Nella scuola dell'infanzia ci sono solo due insegnanti per ogni sezione, quindi, anche in base all'organizzazione oraria delle docenti, ci sono solamente due ore al giorno di compresenza. L'insegnante si ritrova per la maggior parte del tempo da sola nel gruppo classe, affrontando anche momenti difficili. Questo mi ha spinto ad imparare a gestire da sola il mio lavoro, a prendere decisioni in modo autonomo (sempre nel rispetto delle regole scolastiche e della programmazione didattica) e ad acquisire sicurezza di fronte ai bambini. È importante saper prendere in mano la situazione, senza lasciare niente al caso e senza lasciarsi prendere dal panico, anche quando siamo costrette a "improvvisare". Di grande esempio è stata la mia Tutor Scolastica, che mi ha trasmesso la forza di volontà e la passione necessarie per poter svolgere a pieno la professione docente. Nella scuola primaria invece l'insegnamento è organizzato a moduli con insegnante prevalente su Italiano o Matematica. I moduli furono introdotti con la Legge

148/90¹⁵ e prevedono tre insegnanti ogni due classi, che si suddividono equamente gli ambiti disciplinari. Con la Riforma Moratti¹⁶ è stata poi introdotta la figura dell'insegnante prevalente nella classe e grazie all'autonomia scolastica e alla flessibilità organizzativa questa modalità permane ancora. In questo modo abbiamo insegnanti che ruotano su due o tre classi, con prevalenza su discipline come Italiano e Matematica e che completano il proprio orario equamente in altre sezioni. Nelle due classi ci sono quindi tre insegnanti curricolari, più due insegnanti di sostegno. Il gruppo delle docenti è più numeroso rispetto alla scuola dell'infanzia e ci sono molte ore di contemporaneità. Per questo il lavoro di team è indispensabile e fondamentale per una didattica sempre più efficace e inclusiva. Posso affermare che questa modalità di lavorare non è risultata per me nuova, né mi ha messa in difficoltà; all'Università sia Tutor che Professori hanno favorito questa pratica, invitandoci a lavorare in gruppi più o meno numerosi, proprio per abituarci a condividere, ad ascoltare, ad accogliere le idee altrui per arricchirci ed arricchire il lavoro da svolgere. Fortunatamente sono stata subito accolta dal gruppo dei docenti, in modo che anche io potessi partecipare alla vita scolastica e della classe a pieno.

¹⁵ Legge n. 148 del 5 Giugno, 1990, articolo 4, comma 3.

¹⁶ Legge n. 53 del 28 marzo 2003.

Conclusioni

Al termine del mio lungo percorso di studi posso affermare che l'esperienza di tirocinio è l'aspetto fondante della professione docente, sicuramente la più formativa dal punto di vista pratico. Infatti è il "fare" e non semplicemente "sapere" o "osservare" che permette la crescita personale. Come afferma Dewey, *«c'è un'intima e necessaria relazione fra il processo dell'esperienza effettiva e l'educazione»* e sarà questa idea che cercherò di portare avanti come base nel mio futuro di insegnante. Parallelamente a questo difficile percorso, è stato indispensabile il ruolo del tirocinio indiretto svolto all'Università. Grazie alla competenza e formazione della Tutor Scolastica che ha affiancato il mio gruppo, è stato meno difficile affrontare le problematiche incontrate nel corso degli anni, sia da me che dai miei colleghi. Infatti ogni incertezza, dubbio o preoccupazione venivano condivise, così da poterne discutere insieme e fare in modo che la nostra Tutor potesse consigliarci il modo migliore di come comportarsi all'interno della scuola, sia per quanto riguarda la didattica che le relazioni interpersonali con alunni e docenti. Il tirocinio indiretto è stato quindi un supporto, oltre che una "scuola per affrontare la scuola". Se dovessi valutare in modo critico la mia esperienza, potrei affermare che il *gap* tra i punti di forza e di debolezza si è ridotto notevolmente, in quanto, grazie all'esperienza diretta, sono riuscita ad acquistare la sicurezza necessaria e mettere da parte quella timidezza che mi ha sempre impedito di esprimermi al meglio e dimostrare le mie capacità.

Durante questo percorso ho avuto l'occasione di conoscere persone che mi hanno insegnato che non esiste "Il modo giusto" per avere successo, ma che sta a me capire quale sia, l'importante è metterci sempre volontà, passione e impegno. Con questo spirito spero di riuscire al meglio nel futuro nel mio lavoro e anche nella vita, cercando di affrontare sempre tutto con positività e guardando il mondo un po' con gli occhi da "bambina", in quanto non si finisce mai di apprendere ed imparare.

Bibliografia

Ministero dell'Istruzione, dell'Università e della Ricerca, *Indicazioni Nazionali per il curriculum della scuola dell'infanzia e del primo ciclo d'istruzione*, Roma, 2012.

Capperucci D., Piccioli M., *L'insegnante di scuola primaria*, Franco Angeli, Milano, 2012.

Calvani A., *Come fare una lezione efficace*, Carocci Editore, Roma, 2014

Dewey J., *Esperienza e educazione*, Cortina Raffaello, 2014

Trisciuzzi L., *Manuale di didattica per l'handicap*, Editori Laterza, Bari, 1993

Zappaterra T., *La lettura non è un ostacolo. Scuola e DSA*, Edizioni ETS, Pisa, 2012

Ministero dell'Istruzione, dell'Università e della Ricerca, *Linee guida per il diritto allo studio degli alunni e degli studenti con disturbi specifici di apprendimento*, 2011

Sitografia

http://www.edscuola.it/archivio/comprendivi/cooperative_learning.htm
(consultato in data 3/7/2019)

<https://www.tecnicaldellascuola.it/che-cos-e-il-peer-tutoring> (consultato in data 3/7/2019)

<http://www.istruzione.it/pon/ilpon.html> (consultato in data 2/7/2019)

www.qualitaformazionemaestri.it (consultato in data 1/7/2019)

www.gazzettaufficiale.it (consultato in data 5/7/2019)

<https://www.miur.gov.it/web/guest/> (consultato in data 7/7/2019)