

## UN'ESPERIENZA DI INTRODUZIONE AI CONCETTI BASE DELL' INFORMATICA

Giorgio Olimpo, Donatella Persico, Luigi Sarti, Mauro Tavella

► **To cite this version:**

Giorgio Olimpo, Donatella Persico, Luigi Sarti, Mauro Tavella. UN'ESPERIENZA DI INTRODUZIONE AI CONCETTI BASE DELL' INFORMATICA. Atti Convegno Nazionale "Paradigma didattico 85", May 1985, Camerino, Italy. III, pp.131-149, 1985. <hal-01154681>

**HAL Id: hal-01154681**

**<https://telearn.archives-ouvertes.fr/hal-01154681>**

Submitted on 22 May 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



## UN'ESPERIENZA DI INTRODUZIONE AI CONCETTI BASE DELL' INFORMATICA

Olimpo G., Persico D., Sarti L., Tavella M., Istituto per le Tecnologie Didattiche  
C.N.R.

### How to cite:

Olimpo, G., Persico, D., Sarti, L., & Tavella, M. (1985). Un'esperienza di introduzione ai concetti base dell'informatica. In *Atti Convegno Naz. "Paradigma didattico 85", vol.III, (pp.131-149)*. A cura di Università di Camerino e IRRSAE Marche, Camerino, 16-19/5/1985.

Nel corso del lavoro è illustrata una esperienza di introduzione all'Informatica nella formazione di base.

L'intervento, sviluppato nell'ambito del progetto IRIS in coordinazione con il Centro Europeo per l'Educazione, ha comportato lo sviluppo di una unità didattica orientata al primo biennio della scuola media superiore.

L'esperienza viene illustrata sia dal punto di vista dei contenuti che dai metodi adottati, vengono inoltre descritti il contesto in cui è stata condotta, il piano di lavoro seguito, i primi risultati e le prospettive di sviluppo.

In particolare viene presentato il robot Martino, uno strumento software atto ad introdurre i principali concetti alla base dello studio dell'informatica.

### 1 INTRODUZIONE

E' oggi convinzione diffusa che l'informatica non sia soltanto una disciplina tecnico specialistica, ma anche, o forse soprattutto, metodo di pensiero utilizzabile in una molteplicità di discipline e di attività. E' questa l'informatica a cui si riconosce un valore formativo di grande rilievo. Esistono tuttavia punti di vista contrastanti su quali siano effettivamente i contenuti formativi dell'informatica e su come debbano essere inclusi nella formazione di base.

Questa nota intende presentare una specifica esperienza di introduzione ai concetti base dell'informatica in un contesto di scuola media superiore. L'esperienza è descritta per quanto riguarda la scelta dei contenuti e i mezzi con cui tali contenuti possono essere trasmessi, le possibili modalità di condizione e le difficoltà emerse, i primi risultati conseguiti e le prospettive future.

Nell'ambito di tale esperienza è stato utilizzato come strumento software il robot Martino, una versione modificata di "KAREL the robot" di R. Pattis (11).

L'esperienza è stata sviluppata e realizzata dall'Istituto per le Tecnologie Didattiche del C.N.R., nell'ambito del progetto IRIS, coordinato dal Centro Europeo dell'Educazione e condotta in alcune scuole italiane. Il progetto IRIS (Iniziative e Ricerche per l'Informatica nella Scuola) prevede una serie di interventi per l'introduzione dell'informatica a diversi livelli scolari.

### 2 INQUADRAMENTO DELL'ESPERIENZA

#### 2.1 ANALISI DEL.CONTESTO

E' oggi convinzione diffusa che sia sempre più opportuno ed urgente arrivare ad una introduzione organica dell'informatica nella formazione di base.

In alcuni paesi esistono già da alcuni anni, specifici progetti nazionali che mirano a stimolare e guidare il graduale processo di introduzione dell'informatica nei curricula. Fra le più rappresentative si possono citare le esperienze del Regno Unito e del progetto nazionale francese che, pur rispecchiando culture differenti e differenti sistemi scolastici sono tuttavia entrambe frutto di precise scelte politiche che hanno a monte la consapevolezza dell'importanza strategica della diffusione della cultura informatica in una società moderna ed industrializzata.

Nel nostro paese, di recente sono state promosse alcune iniziative, ad opera di associazioni di insegnanti, associazioni culturali e ricreative, singole scuole o singoli insegnanti.

Esistono inoltre numerose iniziative di "iniziazione" all'informatica facenti capo al mercato editoriale, alle principali case produttrici di microcomputer e ai rivenditori e concessionari di hardware, che mirano a fornire un quadro prevalentemente tecnico dell'informatica in cui gli ingredienti fondamentali sono la struttura dell'elaboratore e il linguaggio di programmazione BASIC. Esse tendono a soddisfare la presente richiesta del mercato creando una cultura informatica orientata in senso eccessivamente tecnologico in cui sono lasciati fuori proprio quegli elementi di maggior potenziale sia formativo che di collegamento tra le diverse aree disciplinari.

Solo di recente il Centro Europeo dell'Educazione, organo distaccato autonomo del Ministero della Pubblica Istruzione, ha dato vita al progetto IRIS (Iniziativa e Ricerche per l'Informatica nella Scuola). L'esperienza in esame, di respiro più ampio rispetto alle iniziative citate, fa parte di questo progetto, ed è costituita da un corso finalizzato all'introduzione dell'informatica nella formazione di base.

Un altro significativo aspetto della situazione italiana è costituito dalla scarsa diffusione della cultura informatica nell'ambito della classe docente e dalle difficoltà di aggiornamento di un sistema scolastico caratterizzato dall'inerzia sia da una parte degli insegnanti che delle strutture burocratiche. Anche in questo campo ci si è pertanto affidati all'iniziativa di singoli insegnanti motivati dall'interesse personale e dalle istanze degli studenti.

Difficoltà di ordine tecnico si rilevano nello inserimento dell'informatica nei curricula di base e nella non omogenea situazione economica delle scuole.

## 2.2 OBIETTIVI

Esperienza in esame è rivolta non tanto allo addestramento di specialisti quanto all'introduzione dell'informatica nella formazione di base. Gli obiettivi sono pertanto individuabili in quegli aspetti della cultura informatica che hanno particolare valore

formativo e che contribuiscono a facilitare l'inserimento dei giovani in una società informatizzata

- favorire lo sviluppo di capacità logiche e cognitive
- migliorare la qualità dell'apprendimento
- educare a vivere in una società informatizzata.

a) favorire lo sviluppo di capacità logiche e cognitive.

La costante applicazione del metodo informatico favorisce l'acquisizione della capacità di affrontare sistematicamente la realtà in modo organico, sia per quanto riguarda il problem solving che per sviluppare le potenzialità di apprendimento.

In particolare, il metodo di decomposizione gerarchica e l'uso di metodologie formali di descrizione della realtà accrescono, le capacità comunicative mentre la formulazione di modelli e l'astrazione di funzioni e dati costituiscono le basi del pensiero astratto.

b) migliorare la qualità dell'apprendimento.

Numerosissime proposte e sperimentazioni condotte da docenti e da ricercatori hanno contribuito ad evidenziare un'ampia gamma di vantaggi didattici, certamente in parte ancora da scoprire, che si possono ottenere dall'introduzione di metodi e informatiche. Tali vantaggi riguardano la possibilità di ampliare rinnovare i contenuti disciplinari, la possibilità di dare maggiore concretezza ed operatività alle conoscenze ed infine la possibilità di una migliore comunicazione didattica, più agevole e più incisiva, soprattutto in relazione ad argomenti complessi. Questo obiettivo è necessariamente complementare al precedente in quanto le capacità di problem solving e di astrazione, come atteggiamento mentale sistematicamente adottato dallo studente sono naturalmente applicate in qualunque contesto reale.

c) Educare a vivere in una società informatizzata.

Questo obiettivo implica l'acquisizione di una certa familiarità con l'utilizzazione dei mezzi informatici e le loro possibili applicazioni, diffondendo il concetto di computer come strumento da "utilizzare" piuttosto che "subire" e smitizzando l'informatica come disciplina inaccessibile.

### 2.3 CONTENUTII

In accordo con gli obiettivi detti è possibile individuare (9) un insieme di contenuti basati sul criterio, già accennato, di privilegiare metodi di pensiero di portata generale piuttosto che aspetti tecnologici di valore contingente.

Tali contenuti possono essere suddivisi in due categorie fondamentali:

- costruzione strutturata di algoritmi e programmi
- educazione all'accesso all'informazione e alla sua manipolazione mediante l'uso didattico dei

programmi precostituiti.

Naturalmente, ad ogni livello scolare corrisponderanno diversi gradi di completezza, di approfondimento, di complessità e una serie di considerazioni che possono condizionare una ulteriore selezione nell'ambito di tali contenuti.

Nel seguito sono illustrate in maggior dettaglio le due classi di contenuti citate e vengono motivate le scelte operate.

a) costruzione strutturata di algoritmi e programmi

Possedere il concetto di algoritmo è condizione necessaria per conoscere le potenzialità dell'elaboratore; acquisire le capacità di sintetizzare algoritmi significa aumentare le proprie capacità di inventare soluzioni di generici problemi ed ha di conseguenza particolare lavoro formativo.

La cosiddetta programmazione strutturata è qualcosa di più della semplice costruzione di un algoritmo e della sua codifica in un linguaggio di programmazione: è un metodo di pensiero che si basa sulla capacità di scoprire e costruire la struttura del reale ed adotta tecniche definite per dominarne la complessità e guidare in modo organico il processo di invenzione. I cardini di questo metodo di pensiero fra loro intimamente collegati, sono il concetto di algoritmo, il metodo di decomposizione gerarchica, chiamato in gergo "top down" e l'astrazione funzionale e dei dati.

b) Educazione all'accesso all'informazione ed alla sua manipolazione mediante l'uso didattico di programmi precostituiti.

Oltre alla costruzione di algoritmi e programmi, anche l'uso di programmi precostituiti, come data base, programmi di simulazione, editor grafici, giochi ed in generale qualunque "ambiente" didattico può costituire un fattore educativo di grande rilievo.

Esso infatti comporta:

- l'abitudine a vedere l'elaboratore come uno strumento di lavoro in grado di facilitare, migliorare o accelerare attività produttive e cognitive
- l'abitudine a modellare e formalizzare situazioni ed idee che permette di comunicare con i mezzi informatici e di integrarli funzionalmente con l'attività che si sta svolgendo
- l'arricchimento della didattica con capacità computazionali e di manipolazione della informazione altrimenti inaccessibili
- uno stimolo di creatività grazie al fatto che lo studente ha a disposizione mezzi che ne potenziano le capacità espressive.

Come è già stato detto all'inizio di questo paragrafo, la sperimentazione in esame non ha coperto l'intera area di contenuti descritta. Nell'ambito delle due classi citate, tenendo conto del livello scolastico

dell'intervento e dei requisiti di compattezza che il corso stesso doveva soddisfare, è stata operata una selezione che ha portato ad un insieme più dettagliato e ristretto di punti, calibrato per quanto riguarda il grado di completezza, di approfondimento e di complessità', i fattori motivanti, i collegamenti culturali e quelli con il contesto reale.

La prima classe di contenuti è integralmente presente nel corso in quanto sia l'approccio algoritmico che il processo di decomposizione gerarchica e l'astrazione funzionale sono stati ritenuti indispensabili e basilari, concordemente con quanto detto prima.

La seconda classe è rappresentata dall'uso di un ambiente operativo e di un ambiente didattico facilitato.

#### 2.4 ASPETTI METODOLOGICI

Coerentemente con gli obiettivi ed i contenuti esposti si è deciso di impostare un'unità didattica in cui i protagonisti non fossero, come oggi quasi d'obbligo, la struttura dell'elaboratore ed il linguaggio BASIC. Ciò non significa che s'intenda fare dell'informatica "povera", solo con l'uso di carta e matita: tutti gli aspetti concettuali del metodo informatico possono prendere corpo soltanto attraverso la loro applicazione alla soluzione di specifici problemi e, in ultima analisi, attraverso la costruzione di programmi scritti in uno specifico linguaggio e la loro esecuzione sull'elaboratore.

Le scelte operate per quanto riguarda l'ambiente in cui lo studente opera sono: il linguaggio Pascal, il sistema operativo UCSD e il robot Martino come ambiente didattico facilitato nonché' la adozione di personal computer come supporto hardware.

##### a) Il linguaggio.

Molti fattori hanno considerato la scelta: la disponibilità del linguaggio, l'età ed il livello scolare degli allievi, le conoscenze dei docenti, etc.; ma a queste considerazioni di carattere prevalentemente tecnico è stato anteposto il fatto che ogni linguaggio ha una sua filosofia e sottende un differente metodo di pensiero: il linguaggio Pascal, in particolare, si costituisce non solo come strumento di codifica, ma soprattutto come strumento di pensiero e di ideazione; non lascia trasparire elementi che dipendono dalla specifica architettura dell'elaboratore e non costringe ad implementare strutture logiche intrinsecamente semplici in modo complesso da realizzare e da capire, prestandosi naturalmente all'applicazione degli strumenti concettuali discussi.

b) L'hardware e il sistema operativo. La non uniforme situazione economica delle scuole coinvolte suggerisce l'adozione di personal computer come supporto hardware per permettere anche a scuole con limitata capacità di investimento l'acquisizione di posti di lavoro inferiori in numero, ma non in qualità, rispetto a scuole con budget superiore. La scelta del sistema operativo UCSD deriva naturalmente

dalla scelta del linguaggio e del supporto hardware per la sua diffusione, che lo rende uno standard de facto; esso è inoltre facile da imparare e da usare anche per utenti inesperti.

c) Il robot Martino. Per temperare le difficoltà dei contenuti è stato deciso di ricorrere ad uno strumento software, il robot Martino, basato su di una variante del robot Karel di R. Pattis (11), capace di facilitare l'acquisizione dei concetti informatici e di accrescere la motivazione degli studenti grazie, alla presenza di elementi di gioco.

Il ruolo di Martino è quello di fornire un riferimento concreto nel momento dell'acquisizione di nuovi concetti e della invenzione delle astrazioni per la verifica della correttezza delle soluzioni adottate; ed infine un contesto motivante di gioco. Il mondo in cui il robot Martino si muove è costituito da un reticolo di "strade" fra loro ortogonali (fig. 1).

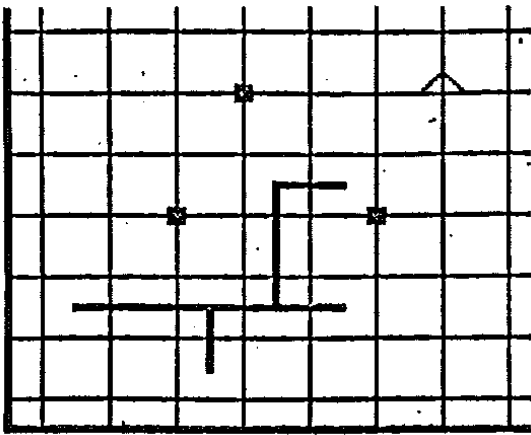


Fig.1

Il robot è rappresentato dalla freccia, che indica la direzione in cui è orientato. Negli incroci si possono avere dei generici oggetti (rappresentati da un piccolo quadrato pieno) che Martino può raccogliere e posare. Si possono infine avere ostacoli di varia forma e lunghezza che rappresentano un vincolo agli spostamenti di Martino.

L'unico modo che lo studente ha per comunicare con Martino è scrivere un programma che gli faccia eseguire un dato compito; in particolare, si è scelto di immergere nel linguaggio Pascal alcune primitive di controllo del robot: si hanno istruzioni che fanno eseguire a Martino particolari operazioni (avanza, destra, sinistra, prendi, posa) e predicati che restituiscono informazioni sulla situazione in cui si trova Martino (oggettopresente, liberodavanti, liberoadestra, anord, asud, etc.).

Si noti che, a differenza del robot Karel, per programmare Martino si utilizzano un linguaggio ed un ambiente di programmazione reali: l'UCDS Pascal.

In questo modo sarà possibile far compiere a Martino specifiche operazioni in un dato "mondo" opportunamente configurato mediante un vero proprio

programma Pascal. Per esempio il compito "attraversare un corridoio raccogliendo tutti gli oggetti eventualmente presenti" potrà essere risolto da un semplice programma come illustrato in figura 2.

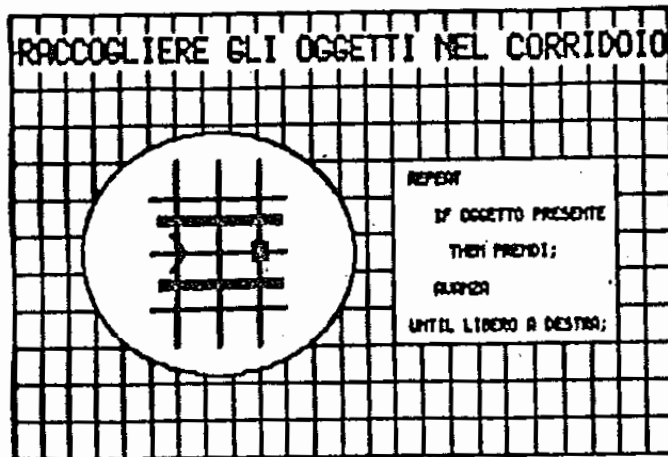


Fig.2

In questo modo il docente guiderà gli studenti a programmare Martino per 'risolvere compiti sempre più complessi e fornirà ad essi i "mondi" in cui tali compiti devono essere svolti. Lo studente costruisce i programmi di Martino come normali programmi Pascal ed in fase di esecuzione gli vengono visualizzate le operazioni compiute da Martino per effetto del programma da lui costruito.

Il mondo di Martino costituisce un riferimento estremamente concreto per l'introduzione graduale dei vari concetti informatici: algoritmi, astrazione funzionale, procedure, strutture di controllo, sviluppo top down. E' importante osservare che tutti questi concetti possono essere introdotti e resi operativi in assenza di dati. Questi ultimi vengono introdotti in un secondo momento, inizialmente con riferimento a Martino e, successivamente, su problemi del mondo reale. Ciò rappresenta un'ulteriore semplificazione dell'apprendimento perché consente di apprendere concetti distinti in tempi successivi.

La struttura dei contenuti ne riflette i criteri di priorità concettuale introducendo per primi, in modo graduale, gli elementi ritenuti più formativi. Il mondo di Martino permette di isolare tali concetti e consente allo studente di concentrarsi su di essi senza essere distratto da elementi di natura diversa e meno importanti. Ad esempio, l'astrazione funzionale e l'uso di procedure sono introdotte quasi subito, prima ancora delle strutture di controllo, mentre i dati non sono introdotti affatto, grazie all'autoconsistenza del linguaggio di Martino, che consente un approccio "pulito" al concetto di algoritmo come sequenza di passi per l'esecuzione di un compito prefissato. Questo tipo di struttura dei contenuti è decisamente differente da quello che è la tradizionale sequenza di presentazione dei contenuti:



variabili, struttura di controllo ed infine procedure.

### 3 CONDUZIONE DELL'ESPERIENZA.

#### 3.1 IL MATERIALE

La sperimentazione in esame costituisce, come già detto, una unità didattica del progetto IRIS e in quanto tale è stata sviluppata in conformità con alcuni standard prefissati dal Centro Europeo Dell'Educazione: tra questi si può citare la struttura modulare del materiale cartaceo, la presenza di una sequenza di schede destinate allo studente e di una guida per il docente. Il materiale del corso si compone di un manuale studente, un manuale docente, di alcuni dischetti contenenti il software di Martino, più alcune schede di test.

Martino accompagna l'introduzione di concetti di base dell'informatica in cinque fasi, ognuna identificata da un insieme logicamente correlato di contenuti:

fase 1: Prime nozioni sugli elaboratori

fase 2: I primi passi

fase 3: Come definire nuove istruzioni

fase 4: Se ... allora ...

fase 5: Ripeti ... finche'...

Ogni fase si articola in attività inquadrata in tre contesti distinti: in aula, a casa e in laboratorio; mentre in classe il docente introduce i nuovi concetti utilizzando eventualmente la macchina per brevi dimostrazioni, l'attività a casa consiste nella preparazione individuale di esercitazioni che si svolgeranno a gruppi, sul computer, in laboratorio. La distribuzione temporale dei tre tipi di attività non è costante, ma varia da fase a fase. La suddivisione in fasi è presente sia all'interno delle schede dello studente che della guida per il docente. Le prime, coerentemente con la convinzione degli autori della necessità di una partecipazione attiva dello studente, costituiscono una vera e propria dispensa interattiva, contenendo frequenti stimoli ad eseguire esercizi e mantenendo una struttura interlocutoria anche durante l'introduzione di nuovi concetti.

La guida per il docente, indispensabile se si considerano le già citate difficoltà di aggiornamento degli insegnanti, fornisce indicazioni di carattere generale circa gli scopi del corso, i contenuti, i tempi ed i mezzi richiesti, un piano d'uso del materiale, le modalità per lo svolgimento di ogni fase.

Queste cinque fasi costituiscono la prima parte di un corso di introduzione all'informatica la cui seconda parte, a cura dell'Istituto per la Matematica Applicata del Consiglio Nazionale delle Ricerche, abbandonato l'ambiente di Martino, introduce le strutture dati mediante un esempio reale.

#### 3.2 CARATTERISTICHE

La sperimentazione si è svolta in un numero limitato di classi di scuole distribuite sul territorio italiano (Torino, Genova, Napoli, Pavia). Gli studenti appartenevano alle prime classi di scuola media superiore e pertanto la loro età media era circa 14 anni. E' da notare che nel curriculum ufficiale di tali classi non era prevista alcuna materia dell'area informatica.

I docenti coinvolti nella sperimentazione hanno svolto questa attività a titolo volontario e con molto entusiasmo. Essi conoscevano già i ragazzi essendo i loro insegnanti di qualche materia scolastica e avevano già raggiunto un buon livello di affiatamento. In generale, nessuno dei docenti possedeva una precedente esperienza nell'insegnamento dell'informatica, anzi per alcuni di essi si è trattato di un'esperienza del tutto nuova ed estranea alla propria esperienza professionale. Tra gli sperimentatori erano compresi docenti di estrazioni differenti come matematica, fisica, tecnica aziendale, ma anche lettere e storia.

Il problema della preparazione dei docenti si è dunque rivelata particolarmente delicata a causa del fatto che essi non disponevano della conoscenza e dell'impostazione culturale necessarie. La tematica è stata affrontata su due linee distinte. Da una parte sono stati organizzati dei corsi di aggiornamento (della durata di una settimana) utilizzando come materiale didattico la stessa unità che essi avrebbero successivamente usato con gli studenti. Dall'altra si è curata in modo particolare la "guida del docente", che costituisce una vera e propria "regia" delle singole lezioni ed esercitazioni, completa di suggerimenti per affrontare eventuali situazioni critiche e di spunti per possibili sviluppi applicativi. L'ipotesi, ancora in parte da verificare, è che una tale guida consenta l'uso del corso anche a docenti privi di una preparazione specifica e lo renda un vero e proprio strumento di autoaggiornamento.

### 3.3 CONTROLLO

Nell'ambito della sperimentazione sono stati previsti ed utilizzati strumenti atti a raccogliere dati il più passibile oggettivi circa l'andamento dell'apprendimento degli studenti:

- un questionario di inizio corso ha fornito informazioni utili a caratterizzare gli studenti quanto a precedenti conoscenze sugli elaboratori
- un test di gradimento, compilato dagli studenti al termine di ogni fase, ha evidenziato elementi di incompletezza, ambiguità o ridondanza nella struttura di ogni lezione
- un test di valutazione proposto al termine delle fasi 3 e 5 ha permesso di verificare che i contenuti base del COI:SO venissero correttamente acquisiti e consolidati dagli studenti
- una prova finale, consistente nella produzione di un programma di una certa complessità e generalità, e proposta agli studenti come gara individuale in cui il vincitore sarebbe stato

l'autore del programma più corretto ed elegante, ha permesso di misurare globalmente l'andamento dell'intero corso per ogni singolo individuo.

A questi controlli di carattere sistematico si è affiancata una attività di controllo meno formale ma più diretta:

- prima, durante ed al termine della sperimentazione, vi sono stati incontri tra gli insegnanti coinvolti e gli autori dell'unità didattica, con proficui scambi di opinioni, consigli, critiche.
- il corso è stato gestito in alcuni casi dagli insegnanti, in altri casi ad essi si sono affiancati gli autori del corso, in altri casi ancora gli autori stessi hanno completamente sostituito gli insegnanti. Ciò per valutare l'incidenza del fattore "docente" e per ottenere un feedback immediato dell'impatto sugli studenti.

### 3.4 PRIMI RISULTATI

I meccanismi di controllo predisposti, unitamente alle sensazioni degli autori e degli insegnanti coinvolti, hanno consentita una misura sufficientemente accurata dei risultati di questa prima sperimentazione.

Dai questionari di inizio corso è emerso che nessuno degli studenti possedeva precedenti conoscenze di informatica né di programmazione.

Il livello di gradimento delle varie fasi è stato ingenerale molto elevato ed è aumentato fortemente dal momento dell' introduzione del robot Martino, evidentemente grazie alle motivazioni di carattere ludico. Molta gradite sono state anche le attività in laboratorio, che si sono svolte a gruppi in generale molto affiatati. Sono emerse difficoltà sola nel caso di gruppi troppo numerosi (dovuti alla scarsità di macchine disponibili in alcune scuole) o in situazioni in cui uno degli elementi del gruppo, tendendo ad emergere ed a costituirsi leader, monopolizzava l'uso della macchina. In generale si è potuto notare come la dimensione ottimale del gruppo sia di circa tre persone; si deve tuttavia tener conto del fatto che un solo insegnante non può assistere troppi gruppi contemporaneamente.

I test di valutazione, unitamente alla prova finale, hanno evidenziato un ottimo livello di apprendimento del linguaggio, dell'uso del sistema operativo e del computer in generale; sorprendenti sono la rapidità e l'entusiasmo con cui i ragazzi di questa età acquisiscono capacità operative sui microcomputer. La capacità di problem solving è risultata senza dubbio potenziata anche se la brevità del corso non ne ha consentito, in alcuni casi, un consolidamento a livello di metodo sistematico.

Una considerazione a parte va riservata alle stime che, nella guida per il docente, fornivano un'indicazione dei tempi necessari alle varie attività. Mentre tali valutazioni sono risultate per lo più corrette per le attività in classe e a casa, non altrettanto si può dire dei tempi per le esercitazioni in laboratorio, spesso

sottostimati soprattutto nel caso di gruppi numerosi o di hardware soggetto a frequenti guasti.

#### 4 CONCLUSIONI E PROSPETTIVE

Per concludere, si può notare che l'unità didattica in esame ha avuto un notevole successo, sia per quanto riguarda l'impatto sugli studenti, che hanno rapidamente acquisito le capacità di problem solving di cui inizialmente non disponevano, sia per quanto riguarda i docenti, che indipendentemente dalla loro formazione hanno lavorato duramente, nonostante l'assenza di riconoscimenti ed inquadramenti ufficiali della loro attività, motivati oltre che dall'interesse personale anche dalle sollecitazioni e dalle entusiastiche reazioni dei ragazzi.

Dalla sperimentazione sono inoltre emerse alcune indicazioni di revisione del materiale e dei contenuti stessi del corso, che si articolano su linee distinte a seconda dei tempi di incidenza.

Una serie di suggerimenti molto puntuali proposti dagli insegnanti, emersi durante gli interventi diretti degli autori nelle classi o dai questionari e dai test, hanno costituito l'input di una fase di revisione che ha prodotto un'edizione più completa ed efficace del materiale. Esempi di tali modifiche sono un ampliamento delle funzionalità del sistema operativo, l'inserimento di "pagine" intercambiabili a seconda del modello di elaboratore utilizzato, etc.

A livello di contenuti si intende prolungare l'unità didattica inserendo, dopo le strutture di controllo fondamentali, il concetto di dato. Questo dovrebbe consentire, da un lato, di completare in modo naturale il concetto di astrazione aggiungendo all'astrazione funzionale quella dei dati, dall'altro di prolungare nel tempo l'uso di metodologie top down, applicandole nel processo di strutturazione dei dati; ovviando in questo modo all'inconveniente, già accennato, costituito dal fatto che non tutti gli studenti si sono abituati in modo ben radicato a questo metodo di pensiero.

Una prospettiva a più lunga scadenza è la realizzazione di un ambiente di sviluppo e testing per la programmazione concorrente, molto simile all'attuale mondo di Martino, in cui si muovano più robot: "Martino e i suoi fratelli".

#### 5 BIBLIOGRAFIA

- (1) Andrews, G.R., Schneider, F.B. (1983). Concept and notation for concurrent programming. *ACM Computing Surveys*, vol.15, n.1, pp. 3-43.
- (2) Ferraris, M., Midoro, V., Olimpo, G., Persico, D., Sarti, L., Tavella, M. Bottino, R.M., Forcheri, P., & Molfino, M.T. (1985). *Corso di Introduzione all'informatica*. Torino (Italy): SEI.
- (3) Bayman, P., Mayer, R.E.(1983). A Diagnostic of Beginning Programmers Misconceptions of BASIC Programming Statements. *Communications of ACM*, vol.6, n.9.
- (4) Chen, P.P. (1976). The Entity - Relationship Model:

Toward a Unified View of Data. *ACM Transaction on data-base systems*, vol.1, n.1.

- (5) Dijkstra, E.W. (1982). How do we tell truths that might hurt?. *ACM SIGPLAN*, vol. 17, n.5.
- (6) Ferraris, M., Midoro, V., Olimpo, G.(1984). Petri net as a modelling tool in the development of CAL courseware. *Computers and Education*, vol.8, n.1.
- (7) Fierli, M. (1982). Aspetti formativi dell'informatica di base. In *Atti della giornata AICA su Contenuti formativi dell'informatica di base*.
- (8) Olimpo, G. (1982). Aspetti formativi del metodo informatico, In *Atti della giornata AICA su Contenuti formativi dell'informatica di base*.
- (9) Olimpo, G.(1984). Informatica nella formazione di base: obiettivi, contenuti e metodi, In *Proceedings of Simposio sobre Informatica y Educacion*, Tucuman.
- (10) Olimpo, G., Persico, D., Sarti, L., & Tavella, M. (1985). An experiment in introducing the basic concepts of informatics. In *Proceedings WCCE'85*. North-Holland Amsterdam.
- (11) Pattis, R.E.(1981). *Karel the robot*. New York:John Wiley.