



Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo

Michele Cerulli

► To cite this version:

Michele Cerulli. Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo. 1999. hal-00190499

HAL Id: hal-00190499

<https://telearn.hal.science/hal-00190499>

Submitted on 23 Nov 2007

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.

UNIVERSITÀ DEGLI STUDI DI PISA
FACOLTÀ DI SCIENZE MATEMATICHE FISICHE E NATURALI

Corso di Laurea in Matematica

TESI DI LAUREA

Uso di manipolatori simbolici
nell'introduzione all'algebra: realizzazione
e sperimentazione di un prototipo

CANDIDATO
Michele Cerulli

RELATORE
PROF.ssa
M.Alessandra Mariotti

CONTRORELATORE
PROF.
Roberto Dvornicich

Anno Accademico 1997/1998

1 Introduzione

1.1 Le nostre motivazioni

Il rapporto uomo-matematica-tecnologia

Uno degli argomenti di dibattito che lo sviluppo delle tecnologie informatiche porta con sé è certamente la rivisitazione e riformulazione dei programmi scolastici in funzione dei nuovi mezzi a disposizione. A questo scopo ci sono alcune domande che ci possiamo porre: quali sono le capacità e le conoscenze che uno studente dovrebbe avere alla fine dei suoi studi? È ancora necessario che l'alunno sviluppi notevolmente la capacità di svolgere compiti che vengono usualmente svolti da un computer?

Per rispondere a queste domande a mio avviso dobbiamo indagare qual è il ruolo giocato dal computer (dalle tecnologie in generale) nello sviluppo della matematica e quale invece è il ruolo giocato dall'uomo nello stesso ambito. In particolare non solo dobbiamo studiare il rapporto triangolare *uomo-tecnologia-matematica*¹, così come esso si configura al presente, ma dobbiamo anche cercare di prevedere come (per lo meno in un futuro prossimo) potrebbe mutare con il passare del tempo. In questo modo potremmo individuare degli elementi caratterizzanti, delle *invariabili*², su cui basare gli interventi didattici.

Nell'ambito della Matematica abbiamo la Ricerca, la Tecnologia e l'Uomo. La Matematica si evolve grazie alla ricerca fatta dall'uomo. Questa ricerca avviene anche con l'aiuto di strumenti tecnologici³. I progressi matematici contribuiscono

¹ Intendo sviluppo della matematica, ricerca.

² Prendiamo questo termine in prestito dall'urbanistica, in tale ambito esso sta ad indicare gli elementi di un luogo che sono rimasti invariati durante tutta la sua e che quindi lo caratterizzano, essi sono elementi "da non mutare, da tutelare/conservare per il loro duplice e correlato senso di luoghi storici che esprimono il carattere dell'area presa in esame" (Cervellati pg. 83). In particolare, "le invariabili, nei fatti, rappresentano l'identità del luogo da pianificare".

Nel nostro caso quindi, con la parola *invariabili*, intendiamo gli elementi che hanno caratterizzato la matematica durante tutta la sua storia, che non dipendono dalle singole nozioni, e che supponiamo immutabili nonostante lo sviluppo tecnologico.

³ In effetti questo è sempre accaduto, basti pensare alla geometria della riga e del compasso.

allo sviluppo della tecnologia. La cultura dell'uomo si evolve sia grazie alla crescita della matematica che della tecnologia. Ognuna delle tre componenti del triangolo contribuisce all'evoluzione di entrambe le altre, ma i contributi sono di natura diversa. Nel nostro caso specifico è rilevante osservare che la tecnologia, attualmente, contribuisce alla ricerca matematica essenzialmente fornendo strumenti di calcolo (in senso lato) e micromondi che facilitano l'approccio sperimentale alla ricerca (ad esempio programmi per visualizzare particolari oggetti geometrici). In entrambe i casi il ruolo fondamentale, come strumento tecnologico, lo gioca il computer che può espletare entrambe le funzionalità a seconda dei softwares a disposizione.

Se da un lato la tecnologia mette a disposizione i mezzi di cui sopra, è anche vero che è l'uomo a crearli ed usarli, il loro apporto allo sviluppo della matematica è quindi passivo, o comunque subordinato all'azione umana. Essi infatti mancano, almeno per il momento, delle capacità di inventare e di creare nuovo sapere autonomamente. In questo appare per ora lontano il momento in cui i computers potranno competere con l'uomo.

Cos'è allora che deve sapere fare l'uomo per far rendere al meglio l'interazione con la tecnologia nella ricerca matematica? A mio avviso deve essere in grado di utilizzare al meglio i mezzi a disposizione, ma anche in grado di produrne di nuovi. In questa ottica gli obiettivi dell'insegnamento devono necessariamente cambiare, sembra quindi opportuno insegnare agli studenti cosa è la matematica, cosa vuol dire fare la matematica, a scapito di abilità e nozioni ormai non più fondamentali.

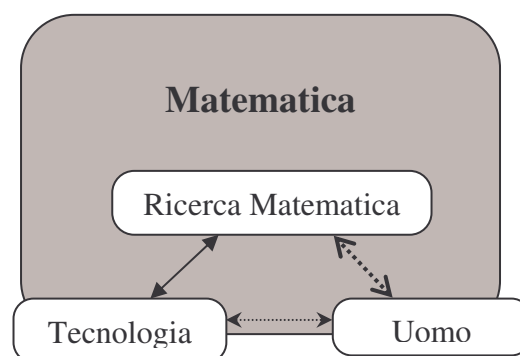


Fig. 1. Il rapporto uomo-matematica-tecnologia

Insegnare a *fare* la matematica

Se il nostro scopo è insegnare a fare la matematica allora dobbiamo individuare una griglia di capacità e concetti che riteniamo parte essenziale dell'attività in questione. Questo in generale sarà un compito arduo. Per la nostra ricerca abbiamo individuato alcuni aspetti che fanno sicuramente parte del cuore della matematica e sui quali abbiamo ritenuto opportuno basare il nostro intervento didattico:

- *l'idea di teoria;*
- *costruzione sociale del sapere;*
- *convenzionalità della matematica.*

Questi aspetti della matematica, a mio avviso, difficilmente possono sparire, o cambiare fortemente in un prossimo futuro, ed è quindi su questi che bisogna basare l'insegnamento.

L'algebra come sistema teorico

L'algebra è un sistema teorico che si basa su un numero piuttosto limitato e ben preciso di assiomi (in questo quindi si differenzia dalla geometria euclidea l'unico sistema assiomatico generalmente insegnato a scuola⁴) nel quale risulta piuttosto facile distinguere bene quali possono essere gli assiomi e quali possono essere i teoremi. In particolare in questo sistema teorico è anche possibile affrontare dimostrazioni piuttosto semplici, sembra quindi adatto per introdurre gli studenti all'idea di teoria e di dimostrazione.

1.2 Il problema didattico

Il problema didattico che affrontiamo riguarda:

l'introduzione degli allievi alla manipolazione simbolica come applicazione di regole (proprietà) sintattiche.

Il problema nasce dall'analisi critica, sviluppata da molti autori (Kieran, 1990, 1992, Sfard, 1991, Arzarello et al. 1992, ecc.), dell'impostazione classica che tende

⁴ Osserviamo anche che la geometria si porta con sé la *sporcizia* della percezione reale, ovvero tutti i preconcetti e le intuizioni legate al modo in cui lo studente vede il mondo che lo circonda.

ad introdurre la manipolazione simbolica come una serie di comportamenti procedurali da mettere in atto nel caso di determinati input operativi: "calcolare", "scomporre in fattori", "raccogliere a fattor comune". In particolare le proprietà delle operazioni sono in genere presentate in termini operativi (procedurali), ovvero come qualcosa che permettere di svolgere un particolare calcolo. Ad esempio la proprietà commutativa è espressa di solito con formulazioni più o meno simili alla seguente: *"cambiando l'ordine degli addendi il risultato non cambia"*». La proprietà è quindi espressa in termini di risultato di una o più operazioni e non in termini di uguaglianza tra espressioni.

Il "fruttuoso rapporto" tra la polarità procedurale e quella relazionale (Arzarelo et al., 1992) viene così inibito, subordinando l'utilizzazione di manipolazioni simboliche a specifiche richieste di carattere operativo e rendendo molto difficile l'uso autonomo di trasformazioni di espressioni per fini diversi da quelli espressamente riconosciuti dalla pratica scolastica.

Al fine di costruire un significato corretto e produttivo della manipolazione simbolica due sembrano i nodi da affrontare.

Da un lato introdurre le proprietà delle operazioni in termini relazionali, dall'altro subordinare il significato operativo ad un esplicito contesto problematico, nel quale entrambi i versi delle uguaglianze trovino ragione di essere interpretati ed utilizzati rispetto al fine specifico di risolvere il problema.

Dunque trasformare un'espressione in un'altra sarà attività consentita attraverso l'uso di proprietà e sarà attività diretta di volta in volta da fini specifici dati dal problema.

1.3 Quadro di riferimento

Dal punto di vista teorico la nostra ricerca si colloca tra più aree tematiche della ricerca didattica, tutte legate all'utilizzo del software da noi prodotto, come abbiamo rappresentato nel diagramma in Fig. 2.

A partire dai risultati ottenuti nel campo della geometria il progetto intende seguire le idee guida già utilizzate nella ricerca riguardante il ruolo del micromondo

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

Cabri nell'introduzione degli allievi all'idea di teoria, adattandole al contesto specifico dell'algebra (Mariotti 1997, Mariotti et Al.1998).

Da un lato l'idea di micromondo e della sua utilizzazione in ambito educativo (Noss & Hoyles, 1997) ha ispirato il progetto di costruzione di un software per l'algebra, che similmente a quanto accade con Cabri, fornisca come sue primitive le proprietà base (gli assiomi) dell'algebra.

Dall'altro l'idea di mediazione semiotica (Vygotsky, 1978) come elemento caratterizzante il processo di apprendimento-insegnamento all'interno di attività centrate sull'utilizzazione del software. Per quanto riguarda l'idea di mediazione ci siamo riferiti soprattutto alla teoria dell'object-mediation (Sfard, 1997), di cui parleremo dettagliatamente in un capitolo dedicato. Questa teoria, in particolare, ha guidato la progettazione e realizzazione di vari elementi caratterizzanti del prototipo,

Infine per quanto riguarda l'insegnamento dell'algebra abbiamo scelto di agire, nel nostro intervento didattico, sull'asse *procedurale-relazionale* riferendoci ad Arzarello - Bazzini - Chiappini, 1994.

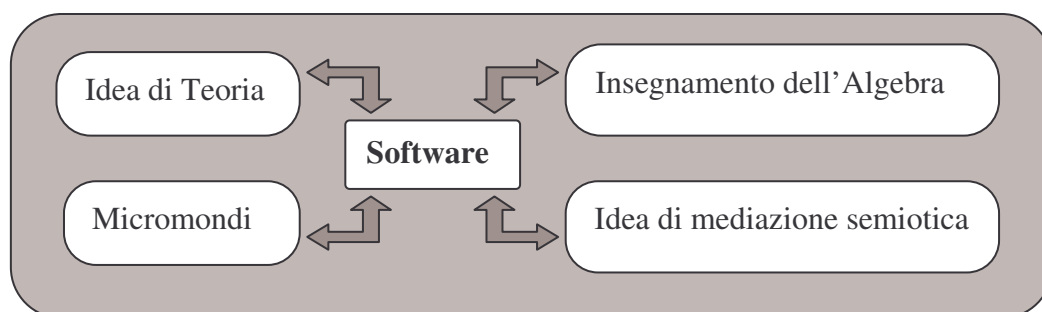


Fig. 2. Aree tematiche della ricerca

1.4 Presentazione del progetto

Descrizione

Al fine di introdurre gli allievi al significato teorico della manipolazione sintattica si è pensato di costruire un ambiente software, un micromondo, per la manipolazione simbolica. La progettazione del software è stata accompagnata dalla messa a punto di un percorso didattico per la sua utilizzazione.

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

La verifica del funzionamento del software nell'ambito del percorso è stata il principale obiettivo della ricerca; questo prevedeva una messa a punto del software 'in itinere'. Correzioni o miglioramenti necessari sono emersi in questa prima fase di verifica .

Un secondo obiettivo della ricerca era l'analisi fine del funzionamento degli strumenti di mediazione offerti dal software, insieme con l'individuazione delle modalità didattiche disponibili all'insegnante per attuare il processo di mediazione semiotica ipotizzato.

In particolare abbiamo cercato di mettere in luce aspetti chiave delle interazioni utente–macchina che forniscano un paradigma del software *migliore*. Abbiamo, quindi, cercato di stabilire dei criteri generali per la progettazione, e la valutazione, di un software una volta che un certo problema didattico è stato stabilito.

Metodologia

Il progetto è stato condotto con la collaborazione di un insegnante e di una classe di prima liceo scientifico.

2 La teoria dell'*object mediation*

In questo capitolo presenteremo una sintesi della tesi esposta da Sfard (Sfard 1997) riguardante il processo di *object-mediation*. Analizzeremo questa teoria cercando di chiarirne i punti fondamentali e mostrando, nei capitoli successivi, come la nostra ricerca si basa su di essa.

2.1 Cosa significa *oggettualizzare*

Nella tesi di Sfard sono distinti due tipi di *discorso*, quello di tipo *AR* (Actual Reality discourse) e quello di tipo *VR* (Virtual Reality discourse).

I discorsi di tipo *AR* sono quelli della vita comune, e sono caratterizzati essenzialmente dal fatto che gli oggetti di tali discorsi sono percettibili, nel senso che sono presenti fisicamente, oppure è come se fossero presenti poiché se ne ha una corrispondente immagine mentale. Con discorsi *VR*, invece, si intende quelli in cui manca la percezione (o pseudo percezione) degli oggetti:

“To put it in a somewhat simplistic way, Actual Reality communication, unlike communication carried out within VR discourse, may be perceptually mediated.” (Sfard 1997).

Un esempio di discorso di tipo *VR* è quello matematico, almeno per soggetti non esperti in materia. I due tipi di discorsi si distinguono per il diverso supporto offerto dalla percezione, ma possono avvalersi delle stesse forme linguistiche, ed, in effetti, questo accade spesso in ambito matematico⁵. È sufficiente osservare una qualsiasi conversazione tra matematici, ed il modo in cui gesticolano, per rendersi conto di come per loro le entità matematiche siano veri e propri oggetti; in particolare potrà

accadere che essi usino termini come “prendi”, “sposta”, “tira su”, “solleva”, riferiti magari ad una funzione continua. In sostanza quello che fanno i matematici è oggettualizzare le entità, di cui parlano, facendo un uso object-mediated dei simboli che le rappresentano; in questo modo essi conferiscono al discorso matematico (che è VR) le caratteristiche tipiche dei discorsi AR:

“Object-mediated use of mathematical symbols is the use which renders the VR discourse the leading characteristics of object-mediated AR discourse. This simple statement has many immediate entailments. It implies that object-mediated mathematical discourse is conducted as if “with an eye to” a certain entity which keeps the discourse *in focus* and *integrated*. The expression “in focus” refers to a discourse in which the participant has clear sense of knowing what his or her utterances are all about; a discourse is “integrated” if there is a general agreement between interlocutors about the focus of the conversation. It is thanks to object-mediation, namely the perceptual accessibility of concrete objects to which the participants refer, that both these features are attained in AR discourse without particular difficulty. [...]. Focus and integration are much more difficult to attain in mathematical discourse, where objects-names and symbols bring almost no perceptual (or as-if perceptual) hints about the relevant aspects of the situation – about those aspects which should be attended to in the conversation.” (Sfard 1997).

Secondo Sfard si fa un uso object-mediated dei simboli matematici quando questi vengono usati in un discorso di tipo AR. Questo può avvenire in particolare dando in qualche modo una consistenza percettiva agli oggetti stessi. In questo modo si fa sì che il discorso in questione sia *in focus* ed *integrated*, ovvero che i partecipanti sappiano costantemente di cosa stanno discutendo, e siano d'accordo sull'oggetto su cui è focalizzata la discussione. Questo può avvenire quando tutti gli interlocutori hanno un comune riferimento “percettivo” per ogni oggetto. In ambito scolastico può

⁵ “Indeed, in descriptions of mathematical processes and objects the same linguistic forms are used as in AR narrative concerned with physical procedures performed on concrete objects” (Sfard, 1997).

accadere che il discorso matematico non sia *in focus* e *integrated*, si può quindi creare una frattura tra insegnanti e studenti: può accadere che i primi dispongano di riferimenti percettivi, riguardanti le entità matematiche, mentre i secondi no, in quanto per loro il discorso matematico è di tipo VR. Uno degli scopi della nostra ricerca è costruire un ambiente in cui le entità matematiche possano avere una consistenza percettiva, in modo che, il discorso matematico, possa essere di tipo AR anche per gli studenti.

Sfard sostiene che l'abilità di oggettualizzare simboli matematici, ovvero farne un uso object-mediated, si traduce in un senso di comprensione in quanto i partecipanti possono accedere (direttamente o grazie all'immaginazione) ad entità differenti dai simboli usati nel discorso: in altri termini sono disponibili "oggetti" di riferimento. Il senso di comprensione in discorsi di tipo VR è molto più difficile da raggiungere proprio perché i riferimenti fisici, che nella vita reale sono già pronti ed accessibili, in questo caso devono essere appositamente costruiti dal soggetto⁶. Il passaggio da un tipo di discorso all'altro è reso possibile dalle analogie linguistiche, cui abbiamo già accennato, ed in particolare dal modo object-mediated in cui possono essere utilizzati i simboli in entrambi i casi:

"Another way to characterize similarity between AR and VR discourses is to say that in both, the use of symbols is often *mediated by objects*. It is very important to understand that in this expression, *the word "objects" does not stand alone and does not signal an existence of special entities which regulate the discourse*. It only has sense within certain phrases and, as will be explained

⁶ "It is important to understand that object-mediated use is not the only possibility in mathematics. In fact, not many students act with symbols in an object-mediated way. More often than not they use names, letters, algebraic formulae and graphs as things in themselves, not standing for anything else (Sfard 1991, 1992). Usually this is not a desirable situation. The ability to make an object-mediated use of mathematical symbols translates into a sense of understanding. Meaningful participation in a discourse is usually the one which is supported with an actual (e.g. made possible by seeing) or potential (e.g. made possible by imaging) access to entities different from the symbols themselves. Clearly this explains why the sense of meaningfulness is so much more difficult to achieve in the VR discourse than in AR: what is readily available through senses in the latter kind of discourse, must be reflectively constructed in the former. To sum up the similarity between AR and VR discourse expresses itself not only in structural linguistic features of these discourses, but also in the actions and experiences of their participants", (Sfard 1997).

later, its use is essentially metaphorical. Within our present discussion, in which a psychological perspective has been adopted, object-mediation means a certain distinct way of manipulating symbols, of solving problems and communicating -- a mode which reminds us of what can be observed when people talk about physical objects, whether actually present or only recalled. Thus, the term "object-mediation" refers (metaphorically!) to a certain set of discursive competencies and underlying psychological processes, and by no means does it try to imply an answer to the question of the existence and nature of the special entities supposed to mediate these processes." (Sfard 1997).

L'uso di un simbolo è mediato da un oggetto (cioè è di tipo object-mediated) quando lo stesso è utilizzato come se fosse un oggetto, o come se rappresentasse un oggetto realmente esistente. Ad esempio quando diciamo "dato che vale la proprietà commutativa, *sposto* il primo termine di una sommatoria in *fondo* alla stessa" trattiamo il *termine* e la *sommatoria* come oggetti, il primo può essere *spostato e portato in fondo* alla seconda, proprio come una sedia può essere spostata in fondo ad un tavolino. La stessa operazione, vista da un punto di vista formale avrebbe tutt'altro aspetto: "Poiché vale la proprietà commutativa, le due sommatorie sono equivalenti, posso perciò scegliere sia l'una che l'altra come rappresentante della classe di equivalenza a cui appartengono, quindi scelgo la seconda".

Oggettualizzare un'entità significa perciò utilizzare il simbolo che la rappresenta in maniera object-mediated, ovvero utilizzarlo *come se realmente esistesse un oggetto*, un riferimento percettivo per l'entità in questione. Bisogna però stare attenti a non lasciarsi trascinare da questioni ontologiche: conformemente alla tesi della Sfard noi non ci occupiamo dell'esistenza o meno di un'entità e quando parliamo di *oggetto matematico* non intendiamo necessariamente qualcosa che esiste in natura; la frase "oggettualizzare un'entità" si riferisce quindi solo ed esclusivamente al manifestarsi di certe forme linguistiche e processi psicologici tipici dei discorsi AR.

2.2 La costruzione dei significati matematici

"The central thesis of this paper [...] is that mathematical discourse and its objects are mutually constitutive: it is the discursive activity, including its continuous production of symbols, which creates the need of mathematical objects; and these are mathematical objects (or rather the object-mediated use of symbols) which, in their turn, influence the discourse and push it into new directions." (Sfard 1997).

Secondo Sfard i significati sono costruiti tramite l'uso di simboli e di oggetti. La parola *uso* va intesa in senso lato, ad esempio si *usano* dei simboli quando questi fanno parte di un discorso. Anche la parola *discorso* ha un senso molto generale, esso in particolare può anche avvenire tra oggetti fisici, ad esempio può rappresentare l'interazione tra la chiave che svita il bullone ed il bullone. In questa interazione è in qualche modo condensato il significato di *chiave*. Se vogliamo interpretarlo in senso linguistico, nel linguaggio di interazione tra chiave e bullone, lo *svitamento* è un'azione, che può necessitare di essere rappresentata da un simbolo, ad esempio la parola *svitamento*, che abbiamo appena introdotto non sapendo come altro indicarla; grazie all'introduzione della parola, lo *svitamento* acquisisce lo status di oggetto in quanto viene *usato* (linguisticamente) come tale. In questo esempio si può notare come è difficile parlare di una qualsiasi entità in assenza di un simbolo che la rappresenti.

Sfard sostiene che la produzione di significati matematici possa avvenire su due piani distinti, quello intradiscorsivo e quello extradiscorsivo; il primo riguarda discorsi (sempre in senso lato) interni alla matematica stessa, mentre il secondo si riferisce a discorsi che collegano la matematica ad altri argomenti. Per non allontanarci dalle questioni trattate nella nostra tesi, in questa sede ci limiteremo ad analizzare solo la costruzione di significati tramite processi intradiscorsivi. A questo proposito Sfard parla di un *pump-mechanism* tra significati, simboli ed oggetti: i discorsi che coinvolgono simboli ed oggetti già esistenti generano nuovi concetti, i quali, a loro volta diventano oggetti rappresentati da simboli (che possono essere

visti, anch'essi, come oggetti), e quindi contribuiscono alla nascita di nuovi significati, nuovi simboli, nuovi oggetti. In sostanza il discorso matematico e la realtà matematica, entrambi di tipo virtuale, non sono mai in equilibrio ed è proprio questa la molla che sta alla base dello sviluppo, intradiscorsivo, della matematica⁷. Dal punto di vista didattico queste assunzioni hanno alcune conseguenze immediate. Appare fondamentale la possibilità di introdurre simboli anche quando non se ne conosca il significato, il quale, potrà essere costruito proprio utilizzando il simbolo stesso. Dal punto di vista didattico, non è affatto imperativo introdurre simboli solo una volta che l'oggetto è stato studiato, anzi, introducendo da subito il simbolo si dà modo allo studente di *parlare* (o partecipare a discorsi in senso generico) usando il simbolo, costruendone quindi il significato.

Un esempio (Sfard 1997), in proposito, tratto dalla storia matematica, è offerto dal termine *funzione* che è stata introdotto da Bernoulli prima che lui ed i suoi contemporanei potessero attribuirgli il significato odierno. Esso in origine aveva un significato piuttosto vago:

“John Bernoulli, 1718: One calls here Function a quantity composed in any manner whatever of this variable and of constants”.

Una funzione per Bernoulli era quindi una quantità *composta in qualche modo* a partire da una variabile e da alcune costanti; la definizione non aveva quindi niente a che vedere con quella attuale espressa in termini di insiemi e di relazioni. Ciò non dimeno, è stato proprio grazie all'introduzione del simbolo (la parola *funzione*), che questo ha potuto estendere il proprio significato fino ad indicare il concetto odierno di funzione. Addirittura le funzioni sono diventate esse stesse degli oggetti sulla base dei quali sono stati definiti ulteriori concetti (es. il concetto di spazio duale). È probabile che se nessuno si fosse preso la briga di dare un nome all'idea di

⁷ “The VR discourse with its special symbolism, and the *virtual reality of mathematics* with its unique experimental manifestations co-emerge in the process of ongoing interaction; in this interaction, the discourse and the ‘mathematical reality’ are never in the state of perfect equilibrium, and this is what makes their joint development possible.” (Sfard 1997).

“comporre una quantità con variabili e costanti” la storia del concetto di funzione sarebbe stata diversa.

2.3 Verso l'oggettualizzazione dei simboli

Sfard distingue due modi di utilizzare simboli, quello templates-driven e quello object mediated. Quando un nuovo simbolo viene introdotto lo studente, in genere, lo utilizza in maniera templates-driven⁸; in questa fase i nuovi simboli vengono usati semplicemente per riempire certe caselle vuote di *frasi* (in senso lato) già pronte. In sostanza vengono presi discorsi riferiti ad altri simboli e questi vengono sostituiti con i nuovi. In pratica la prima cosa che lo studente coglie di un simbolo è il ruolo che questo svolge in alcuni tipi di discorsi; è quindi in grado di riprodurli semplicemente inserendo il nuovo simbolo nel posto giusto, ma non è in grado di creare frasi completamente nuove. Un esempio piuttosto interessante di uso templates-driven di una parola è quello proposto da Sfard⁹ (Sfard 1997): supponiamo che il lettore non abbia mai sentito la parola *krasnal*, però legge la frase “*un krasnal si è svegliato e si è alzato dal letto*”; la frase non dice cosa sia un *krasnal*, ciò non dimeno il lettore potrebbe azzardare qualche frase utilizzando questa parola, ad esempio “*il krasnal dopo essersi alzato ha fatto colazione*”, ed inoltre sarà certamente in grado di dire quali delle seguenti frasi a proposito dei *krasnal* siano corrette:

Ieri un krasnal è andato al mercato.

Dividiamo un krasnal per tre ed eleviamolo al quadrato.

Alcuni krasnals cantano.

Un krasnal comincia la mattina alle 5.30.

⁸ “When structural symbols (symbols that refer to objects) are introduced for the first time, their use is mainly templates-driven, and it is only some time later that it becomes object-mediated. Moreover, I will claim that this order of things, far from being undesirable, is immanent in the logic of the process of symbolization.” (Sfard 1997).

⁹ La traduzione è stata resa riadattandola alla lingua italiana, quindi non è fedelissima.

La prima frase, quella in cui il krasnal si alza dal letto, indirizza il lettore verso un campo semantico tipico degli esseri umani suggerendo possibili frasi da utilizzare sostituendo la parola krasnal ad altre; ad esempio risulterà naturale che sia possibile sostituire le parole "mio cugino" con "quel krasnal" nella frase "mio cugino è simpatico" ed ottenere "quel krasnal è simpatico". Quello che si fa è proprio prendere una frase riferita ad un altro simbolo già familiare e sostituirlo con la nuova parola. In sostanza può bastare una frase esemplare per fornire informazioni sulla natura di un oggetto in maniera da permettere di creare nuovi discorsi contenenti il simbolo rappresentante l'oggetto stesso. La fase templates-driven in genere è molto breve proprio perché, una volta che iniziamo ad utilizzare un simbolo, in realtà ne stiamo già costruendo il significato; quando andiamo ad utilizzare un simbolo per la seconda volta, esso si porta dietro la prima esperienza di utilizzo, quindi inizia già ad avere un suo significato autonomo.

Successivamente alla fase templates-driven c'è un processo intermedio che può portare alla fase object-mediated, cioè quella in cui si oggettualizzano i simboli come abbiamo già descritto nel primo paragrafo. Il processo intermedio non può essere analizzato in quanto troppo vario, possiamo solo dare il suggerimento didattico di cercare di supportare lo studente con eventuali riferimenti fisici (o pseudo fisici) in maniera da favorire un uso dei simboli mediato dagli oggetti. In questa tesi illustreremo appunto un software progettato basandoci su questa idea, e descriveremo un intervento didattico condotto nello stesso spirito.

La fase object-mediated invece, secondo Sfard (Sfard 1997), è caratterizzata dai seguenti comportamenti:

- Flessibilità e generalità d'uso: il simbolo acquista una vita propria ed il soggetto diventa in grado di incorporarlo in proposizioni (frasi, discorsi) che lui/lei non ha mai sentito prima, nuove.

- Il simbolo inizia a funzionare come rappresentazione¹⁰: invece che essere esso stesso l'oggetto del discorso, il simbolo, viene ora usato come rappresentazione di un'altra entità.
- Il simbolo viene spesso inavvertitamente scambiato con altri simboli che hanno un significato in comune con esso: a seconda della situazione avviene un cambio di parole per usarne alcune che siano più adatte alla frase in questione. In sostanza ci si muove all'interno di una classe di simboli che hanno un comune oggetto di riferimento o che hanno un significato comune.
- Economia d'espressione: simbolo può assumere il ruolo di *rappresentante* di una classe di simboli; in questo modo proposizioni che utilizzano simboli diversi possono essere sostituite da una sola proposizione che utilizza il simbolo rappresentante. Se due simboli rappresentano una stessa cosa, quello che prima era detto due volte (una volta per ciascun simbolo), ora grazie all'oggetto che funziona da riferimento comune ai due simboli, può essere detto una volta sola.
- Visualizzazione: il soggetto riesce a visualizzare un'immagine mentale che funzioni da riferimento per il significato del simbolo in questione.

Dal punto di vista della ricerca didattica, la caratterizzazione della fase object-mediated che abbiamo appena dato, rappresenta una griglia di criteri per verificare se uno studente abbia effettivamente oggettualizzato un dato concetto oppure no.

¹⁰ Ricordiamo che nella fase *templates driven* il simbolo è in realtà usato *grammaticalmente*, nel senso che viene inserito correttamente in proposizioni dove sta bene dal punto di vista *linguistico*, o comunque per imitazione rispetto ad altre proposizioni note.

3 Presentazione del software

3.1 Idee di base

Coerentemente con le ipotesi precedentemente avanzate, elemento caratterizzante del progetto è:

- La mediazione semiotica (Vygotsky, 1978) offerta dall'ambiente de' L'Algebrista.

Particolari strumenti di mediazione offerti dal micromondo sono

- i comandi/bottoni che forzano a rendere esplicite le proprietà algebriche utilizzate, altrimenti nascoste nel "calcolo".
- la funzione evidenziare, che svolge una semplice funzione di controllo per l'uso corretto dei bottoni, diviene un segno esterno del controllo teorico sull'applicabilità di una proprietà.

L'idea di base de' L'Algebrista è di fornire all'utente un manipolatore simbolico che svolga su richiesta certi particolari "calcoli", ma che non sia troppo potente. Deve poter applicare, su richiesta dell'utente, ciascuna delle proprietà delle operazioni fermandosi però al primo passo, senza cioè reiterare automaticamente l'applicazione della proprietà all'espressione finché questo è possibile (come fanno la maggior parte dei manipolatori simbolici).

In questa maniera l'utente può scegliere passo per passo le proprietà da applicare, quindi come modificare l'espressione e quando fermarsi.

L'attività non è centrata nella riduzione dell'espressione a particolari forme predefinite, ma nella manipolazione della stessa, è quindi improntata all'assoluta libertà dell'utente, che ha controllo assoluto sulle scelte durante la manipolazione.

Al computer è lasciato il controllo della correttezza dell'esecuzione del comando nonché il controllo sulla applicabilità del comando alla espressione selezionata. Nel caso la proprietà non sia applicabile il computer si limiterà a non eseguire la trasformazione richiesta.

Il ruolo del computer è quindi ridotto allo sviluppo dei calcoli che l'utente gli chiede di svolgere, in sostanza il computer è il braccio e l'utente la mente.

Questo in un certo senso avviene anche nei normali manipolatori simbolici, solo che il potere dell'utente è molto ridotto, ci sono sempre *zone d'ombra* in cui è solo ed esclusivamente il computer a decidere come manipolare l'espressione ed in che forma presentarla e questo di volta in volta potrà dipendere dalla consegna data, dal problema posto. Essi in effetti sono in generale realizzati con obiettivi diversi dal nostro: sostanzialmente si comportano coerentemente ad idee di economia ed efficienza standardizzate nella pratica matematica che mascherano la comprensione.

In questo modo il software dotato dei comandi base corrispondenti alle proprietà fondamentali delle operazioni somma e prodotto, rappresenta un micromondo che incorpora la teoria algebrica delle espressioni letterali. La trasformazione di un'espressione in un'altra corrisponde alla dimostrazione dell'equivalenza tra espressioni all'interno della teoria, alla dimostrazione di un particolare "teorema".

Elenchiamo illustrando ulteriormente le idee fondamentali su cui abbiamo basato la progettazione, la costruzione e l'utilizzo del software:

- Un **manipolatore simbolico** che sia totalmente sotto il controllo dell'utente. L'idea è quella di un micromondo delle espressioni algebriche in cui l'utente può agire trasformando le espressioni stesse in base alle proprietà fondamentali delle operazioni che funzionano da assiomi della teoria in cui ci si muove.
- Gli assiomi sono rappresentati dai **bottoni delle proprietà** (quindi ne mediano il significato) i quali non devono avere nessun comportamento implicito, devono fare solo ed esclusivamente ciò che l'assioma che rappresentano dice, e lo devono applicare una sola volta, non ricorsivamente.
- Il manipolatore stesso, **non deve essere in grado di effettuare alcuna trasformazione extra** se non guidato dall'utente tramite l'uso dei bottoni di cui sopra. In questo modo l'utente è costretto ad esplicitare tutti i passaggi impliciti quando esegue delle trasformazioni *automatiche*¹¹. Questa caratteristica lo

¹¹ Ad esempio l'utente può trovarsi costretto ad insegnare al computer come cancellare b^2-b^2 o sommare termini simili, perché esso non sa eseguire, automaticamente, certe trasformazioni.

differenzia dai normali manipolatori simbolici che tendono nascondere il modo in cui trasformano le espressioni e lasciano poco controllo all'utente. Il nostro deve essere un ambiente in cui l'utente impari a **costruire** un manipolatore simbolico classico, non ad usarlo.

- I bottoni che rappresentino relazioni di equivalenza devono essere reversibili e devono contenere in sé i propri inversi. In questo modo si cerca di rendere esplicito il significato di **equivalenza** tra espressioni e di eliminare l'unidirezionalità del simbolo "uguale" ("=").
- I bottoni devono essere inizialmente un certo numero limitato (stabilito all'inizio), che appunto rappresenti gli assiomi. Ogni qualvolta gli alunni scoprono/dimostrano un teorema, questo viene aggiunto al sistema sotto forma di bottone di teorema (vedi l'esempio $\Delta - \Delta = 0$). In questo modo gli studenti **costruiscono il sistema teorico** rendendosi conto via via dei progressi che fanno. Osserviamo che quella di creare teoremi è l'attività fondamentale dei matematici, in questo modo i ragazzi hanno la possibilità di partecipare ad un'attività significativa per la matematica. In particolare ogni "scoperta" fatta deve essere discussa e ufficialmente accettata in classe.
- Alcuni bottoni conterranno o rappresenteranno delle scritture che richiamano convenzioni della matematica in generale, mentre altri rappresenteranno convenzioni della realtà locale dell'attività in questione. In questo modo si ha la possibilità di esplicitare gli **aspetti convenzionali della matematica**.
- L'interazione tra l'utente ed il software avviene tramite **manipolazione diretta**, ovvero tramite l'uso di bottoni da cliccare. Non è quindi necessario per l'utente imparare un particolare linguaggio di programmazione onde interagire con il sistema.

3.2 Descrizione del software

Il prototipo, che abbiamo chiamato "**L'Algebrista**", è un'applicazione del più noto programma *Mathematica*. Esso è costituito da una o più palette contenenti una serie di bottoni, che descriveremo più sotto, ed una finestra contenente un notebook in cui lavora l'utente (in realtà può usare un numero a piacere di diverse finestre). L'interazione utente computer avviene sostanzialmente inserendo un'espressione nel computer e poi manipolandola selezionandone delle sotto espressioni e cliccando sui bottoni presenti nelle palette. Tutti i bottoni, tranne uno, producono sempre espressioni equivalenti all'espressione originaria su cui essi agiscono.

Descriviamo ora i comandi principali ed altri particolari dell'interazione utente-macchina.

Inserisci Espressione

La funzione di questo comando è di inserire l'espressione selezionata dall'utente in un ambiente di lavoro particolare, ovvero una nuova cella¹² con le seguenti caratteristiche fondamentali:

- a) La cella è costituita da un titolo ed un ambiente di lavoro.

Il titolo è una zona colorata grigio scuro e normalmente contenente l'espressione inserita, ma scritta con un font piuttosto grande. In particolare è una zona di testo editabile, si può quindi cambiare il titolo, ma non è cancellabile. L'impossibilità di eliminare questa parte della cella è necessaria per evitare che in caso di errori o malfunzionamenti dei bottoni, venga cancellata qualche parte di notebook precedente la cella titolo.

La parte che sopra abbiamo chiamato ambiente di lavoro invece conterrà, in principio, un'altra cella con l'espressione inserita all'inizio (scritta in nero) con scritto accanto (a sinistra) *Inizio* con un font più piccolo ed in blu. Le nuove espressioni prodotte tramite l'interazione con i bottoni faranno parte di questo ambiente.

¹² Per il lettore non pratico di *Mathematica* una cella non è altro che un'area all'interno della finestra individuata da una specie di parentesi quadra sulla destra della finestra.

Infine, per cancellare l'intera cella (compreso il titolo) è necessario selezionare la cella, e poi dal menu *Format* la voce *Option Inspector*, all'interno del quale è necessario, tra le *Cell Properties*, cliccare accanto alla voce *Cell Deletable* in maniera che il suo valore passi da *False* a *True*. Una volta fatto ciò basta fare la normale procedura che si fa per cancellare una cella, ovvero selezionarla e pigiare *Canc* o *Del* o *CTRL+x* oppure *Cut* nel menu *Edit*, a seconda del sistema che si sta usando¹³.

b) Quando un'espressione viene inserita nell'ambiente di lavoro avvengono le seguenti trasformazioni:

- Ogni sottrazione viene trasformata in somma di opposti, ad esempio $a-2-b \rightarrow a+(-2)+(-1)\cdot b$.
- Uno spazio o un asterisco tra due termini viene interpretato come operatore prodotto e viene sostituito dal simbolo " \cdot ".

c) Ogni volta che si agisce su un'espressione questa viene duplicata in maniera che non vada persa la storia dei passaggi. La nuova copia dell'espressione, quella su cui avranno effetto le eventuali modifiche, viene riportata a capo, in una nuova cella.

d) **Il Selezionatore.**

Di ogni espressione selezionata in questo ambiente si possono selezionare solo sottoespressioni (dal punto di vista matematico), ad esempio se è data l'espressione $a+b\cdot c$ non si possono selezionare né $a+b$ né $a+$ o $b\cdot$, mentre si possono selezionare a o $b\cdot c$ (nonché b e c)¹⁴.

¹³ In effetti questa procedura è piuttosto complicata e da rivedere in futuro, ma è stata accettata per il momento considerando la mia presenza costante come assistente per gli utenti durante la sperimentazione. Questa come altre difficoltà dovute all'implementazione difettosa de' L'Algebrista sono state accettate e superate senza problemi dai ragazzi durante la sperimentazione.

¹⁴ In effetti è possibile selezionare anche $+b\cdot c$, questo è un effetto collaterale dovuto *Mathematica* che purtroppo non siamo ancora riusciti ad eliminare. In linea di massima il selezionatore funziona bene, solo che a volte dà dei problemi con le parentesi: se abbiamo ad esempio $(a+(b+c))$ lascia selezionare $(a+(b+c))$. Questo difetto può comportare che si selezionino espressioni non sintatticamente corrette, le cui conseguenze non sono comunque catastrofiche come vedremo tra poco.

Infine un'ultimo difetto è che a volte pare *incantarsi* e magari non lascia selezionare sottoespressioni di un'espressione che sarebbero matematicamente e sintatticamente corrette. Per ovviare a quest'ultimo problema abbiamo notato che basta far agire sull'espressione un comando qualsiasi che la lasci inalterata, la nuova copia in genere è perfettamente selezionabile in ogni sua parte.

- e) Qualora l'utente selezioni un'espressione non sintatticamente corretta e prema un qualsiasi bottone, il computer genererà una nuova sottocella contenente un messaggio di errore. Questa a dire il vero è una generica caratteristica di *Mathematica*, ed è sufficiente selezionare l'intera cella e poi la voce *Cut* nel menu *Edit* per eliminare il messaggio di errore. La stessa procedura serve per eliminare qualsiasi passaggio non voluto dall'utente. Ogni espressione prodotta può essere cancellata, ovvero si può sempre tornare allo stato precedente tramite la procedura appena indicata, mentre invece non funziona il normale *Undo*.
- f) Ogni volta che un bottone agisce su un'espressione modificandola, la nuova espressione viene scritta a capo ed è facilmente individuabile poiché si trova nel rigo immediatamente seguente la linea trasversale che indica la posizione del cursore che sarà automaticamente situato esattamente tra la vecchia e la nuova espressione.
- g) Ogni volta che un bottone produce una nuova espressione questa è accompagnata da una *label* scritta con un font più piccolo ed in blu e che riporta il bottone che è stato premuto e la parte di espressione che era stata selezionata. Questa opzione è utile per ricostruire i passaggi fatti.
- h) Le *labels* scritte accanto (o sopra a seconda della loro dimensione) ad ogni nuova espressione possono essere nascoste, in modo da rendere la sequenza dei passaggi leggermente più compatta e più leggibile; ma possono essere riportate alla luce ogni qualvolta l'utente lo voglia. Per fare quest'operazione è necessario selezionare l'intera cella (o le sottocelle su cui vogliamo agire) e dal menù *format* la voce *Option Inspector*. All'interno dell'*option inspector* è necessario selezionare *Cell Labels* e cliccare sul quadratino accanto alla voce *Hide Labels*, in modo da passare da *True* a *False* e viceversa ovvero nascondere o meno le labels¹⁵.

Proprietà associativa

Per quanto riguarda questa proprietà abbiamo sostanzialmente 3 bottoni, tutti scritti in rosso.

- Il primo è rappresentato dalla scrittura " $a \backslash (b \backslash c) \Leftrightarrow (a \backslash b) \backslash c$ ". Come operatore è stato usato un simbolo che non rappresenta né la somma né la moltiplicazione (che sono le due operazioni con le quali lavora il prototipo) in modo che le possa rappresentare entrambe senza dover usare due diversi bottoni per una stessa proprietà. Questa caratteristica va chiaramente esplicitata con gli alunni e dovrebbe aiutare a mettere in luce il fatto che la proprietà è la stessa per le due diverse operazioni. Lo stesso discorso vale per gli altri due bottoni. La doppia freccia sta ad indicare che il bottone fa passare da una scrittura all'altra (tra le due equivalenti) senza nessuna particolare priorità, nel caso in cui trovi una scrittura, all'ora produce l'altra equivalente, senza discriminazioni.
- Il bottone " $\Delta \circ \square \square \square (\Delta \circ \square)$ " ha la funzione di aggiungere delle parentesi esterne alla sottoespressione selezionata.
- Il **Bottone a Rischio** rappresentato anche dalla scrittura " $(\Delta \circ \square) \square \Delta \circ \square$ ", serve a togliere le parentesi esterne ad un termine, ma anche tutte quelle interne rese *inutili* dalla proprietà associativa.

Gli ultimi due bottoni ed il precedente sono chiaramente giustificati dalla Proprietà Associativa poiché è questa che garantisce che togliendo e/o aggiungendo parentesi nei punti giusti l'espressione rimane equivalente alla prima. In particolare è la proprietà associativa che permette di passare dall'originaria scrittura corretta per le operazioni binarie $a \backslash (b \backslash c)$ alla scrittura $a \backslash b \backslash c$ che ha senso solo perché vale tale proprietà. Inoltre per quanto riguarda questi due bottoni osserviamo che la freccia ha un solo verso in quanto non sono univocamente reversibili. Ed è per questo che non sono riuniti in un unico tasto; in effetti, se consideriamo le espressioni $(a+b)+c$ e $a+(b+c)$, esse, possono diventare entrambe $a+b+c$, quindi è necessario l'intervento dell'utente per tornare indietro, il quale può scegliere dove aggiungere le parentesi.

¹⁵ La possibilità di nascondere le labels offre la possibilità di fare attività in cui all'utente viene presentata una serie di passaggi e lui/lei deve ricostruire la sequenza di comandi che è necessario utilizzare per riprodurre tale serie di uguaglianze tra espressioni.

Notare che, grazie alle caratteristiche della cella di lavoro, la sotto-espressione selezionata dall'utente per aggiungere delle parentesi non può non essere una sotto-espressione dell'espressione originaria, quindi l'operazione di aggiungere una coppia di parentesi risulterà sempre matematicamente corretta. Lo stesso discorso **non** vale per il bottone per togliere le parentesi, che infatti è chiamato **Bottone a Rischio**. Esso in effetti può generare degli "errori", ad esempio data $a \bullet (b+c)$ se si seleziona $(b+c)$ viene trasformata in $a \bullet b+c$ che è chiaramente non equivalente all'espressione originaria. Questo bottone è messo in disparte e deve essere perfettamente chiaro per l'utente il tipo di errore che può generare e soprattutto *perché* quello è un errore. L'utente che lo usa si prende la responsabilità di usarlo e di garantire che produrrà una scrittura equivalente a quella originaria.

Infine osserviamo che, sia per il Bottone a Rischio che per quello che applica la Proprietà Associativa, il computer sceglie se applicarlo ai prodotti o alle somme a seconda che la testa dell'espressione selezionata sia un prodotto o una somma, ad esempio il primo dei due applicato all'espressione $a+(b+c)+b \bullet (c \bullet d)$ produrrebbe $a+b+c+b \bullet (c \bullet d)$. Per togliere le parentesi del prodotto bisognerebbe in tal caso selezionare $b \bullet (c \bullet d)$.

Proprietà distributiva

Per quanto riguarda questa proprietà è presente il solo bottone rappresentato dalla scritta, in nero, " $a \bullet (b+c) \Leftrightarrow a \bullet b+a \bullet c$ ". Anche in questo caso la doppia freccia indica che il bottone serve a passare da un tipo di scrittura all'altra. Più precisamente ogni volta che trova un'espressione la cui testa sia una somma cerca di "*raccogliere*", mentre in caso trovi un'espressione la cui testa è una moltiplicazione allora cerca di "*svolgere*". È quindi responsabilità dell'utente scegliere quando usarla e che parte di espressione selezionare per ottenere le modifiche desiderate. Osserviamo che ogni volta che il bottone *svolge* racchiude la sotto-espressione prodotta tra due parentesi onde evitare errori del tipo: $a \bullet (b+c) \bullet d \nRightarrow a \bullet b+a \bullet c \bullet d$. Osserviamo anche che questo bottone e la proprietà che rappresenta svolgono un ruolo fondamentale nella manipolazione di espressioni, e soprattutto osserviamo che, condensando in un'unico tasto le due procedure note come *Svolgi* e *Raccogli*; scompare la distinzione, e resta

la *Proprietà Distributiva*, una relazione di equivalenza di cui esse sono figlie e che spesso si tende a dimenticare. In particolare così facendo poniamo allo stesso livello esercizi ed attività come portare un'espressione in forma canonica o scomporla in fattori che in genere vengono affrontate in contesti diversi ed a molti mesi di distanza l'una dall'altra.

Proprietà commutativa

Il bottone riguardante questa proprietà è rappresentato, in blu, dalla scritta " $a \cdot b \Leftrightarrow b \cdot a$ " ed agisce scambiando solo le somme o solo i prodotti a seconda che la testa dell'espressione sia la somma o la moltiplicazione.

Un'osservazione sul comportamento del bottone. Quando si deve commutare due termini, il bottone, li permuta nell'unico modo possibile. Ma quando si voglia commutare una somma di tre o più termini le possibili permutazioni sono molte, ma il bottone ne produce una particolare, che magari non è necessariamente quella voluta dall'utente. Si crea quindi il problema di come ottenere una particolare permutazione; si può osservare come, selezionando sempre solo due termini si possono ottenere tutte permutazioni desiderate, o meglio ogni permutazione è generata da un numero finito di permutazioni di due elementi.

Il calcolo

I bottoni in questione sono due, uno per la somma ed uno per il prodotto, rappresentati rispettivamente da " $3 \Leftrightarrow 1+1+1$ " e " $2 \cdot 3 \Leftrightarrow 6$ ". Il primo dei due esegue la somma tra numeri, o nel caso sia selezionato un solo numero, lo trasforma in " $1+1+1+\dots+1$ ", il secondo invece esegue il prodotto tra numeri, o alternativamente ne produce la scomposizione in fattori primi. Entrambe funzionano solo ed esclusivamente su numeri, qualora, nell'espressione selezionata, vi sia un qualsiasi termine che non è un numero, essa rimane invariata.

Proprietà degli elementi neutro

Questi sono: " $1 \cdot \square \Leftrightarrow \square$ ", " $\square \cdot 0 \square$ " e " $\square + 0 \square$ ". Essi fanno esattamente ciò che ci si aspetta facciano, è da notare però che il simbolo usato, il " \square " sta a rappresentare un qualsiasi termine, anche magari un'espressione complicatissima.

Le potenze

Per quanto riguarda le potenze abbiamo un bottone, rappresentato da " $\Delta^3 \Leftrightarrow \Delta \cdot \Delta \cdot \Delta$ ". Esso è utile in quanto L'Algebrista non sa interpretare le potenze automaticamente; ad esempio, se vogliamo applicare la proprietà distributiva all'espressione $a^2 + a \cdot b$, per ottenere $a \cdot (a + b)$, dobbiamo prima trasformare a^2 in $a \cdot a$, altrimenti L'Algebrista interpreta a^2 , come se fosse una scrittura indipendente dal simbolo a .

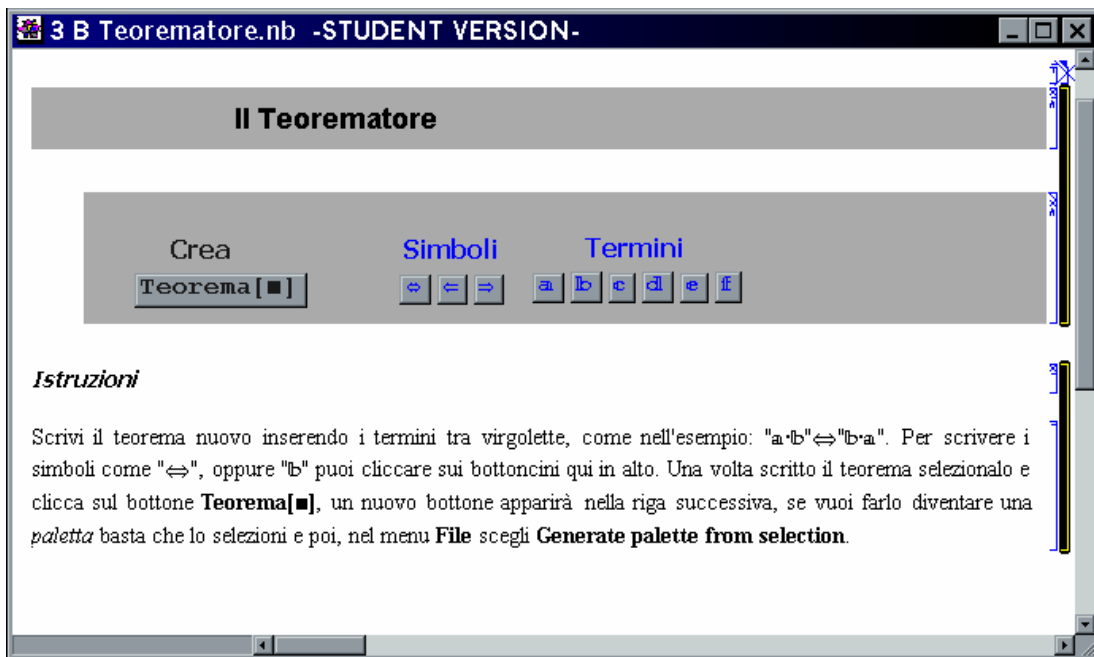
I teoremi

I bottoni finora elencati rappresentano la base su cui lavorare, gli assiomi della teoria incorporata e rappresentata da L'Algebrista, grazie a questi è possibile dimostrare dei teoremi che, quando possibile, vengono a fare parte della teoria tramite l'introduzione di bottoni che li rappresentano. Alcuni esempi di *bottoni teoremi* possono essere:

- $\Delta - \Delta \square 0$
- $2 \cdot \Delta + 3 \cdot \Delta \square 5 \cdot \Delta$
- $\Delta^2 - \square^2 \Leftrightarrow (\Delta - \square)(\Delta + \square)$

Il Teorematore

Il Teorematore è un notebook contenente un bottone che permette allo studente di creare facilmente un qualsiasi bottone teorema a suo piacimento. Osserviamo che, per lo meno originalmente, il Teorematore non esegue alcun controllo sulla veridicità o meno del teorema in questione, tanto meno verifica che sia coerente con gli assiomi a disposizione. Di conseguenza può in realtà essere utilizzato anche per costruire nuove *teorie*. Riportiamo per intero il notebook in questione che contiene anche le istruzioni per l'utente.



3.3 Perché non usare i normali comandi di *Mathematica*?

Riteniamo che l'uso di *Mathematica* sia consigliabile solo in caso di utenti in qualche maniera esperti dal punto di vista matematico. In effetti ci sembra più opportuno l'utilizzo di micromondi come il nostro (eventualmente ed opportunamente arricchito) più adeguato nel caso, ad esempio di interazioni con studenti più piccoli o poco esperti matematicamente. Il punto cruciale sta nel

linguaggio di interazione tra macchina ed utente: nel caso di *Mathematica* è necessario usare scritture poco leggibili e termini (per altro in inglese) poco chiari, la manipolazione non è diretta, bensì si danno al computer gli ordini da eseguire (uno alla volta o tutti insieme) per via testuale. L'apprendimento del particolare *linguaggio*, che è diverso sia da quello matematico che da quello verbale, rischia di assorbire gran parte dell'attenzione e degli sforzi dell'utente. Qualora invece il linguaggio che permette di interagire con il computer sia quello matematico, o quello verbale, o una via di mezzo, abbiamo che gli sforzi (nel caso che ci siano) per apprenderlo non risultano inutili, ed al limite sono parte dell'attività stessa dell'apprendimento della matematica. In particolare nel nostro prototipo l'utente non interagisce mai con la macchina impartendo ordini testuali, ma cliccando su appositi bottoni, quindi scompare la difficoltà della traduzione in *computerese* anche perché le funzioni dei bottoni sono indicate in *matematichese*, quindi lo studente deve, al limite tradurre in quest'ultimo linguaggio.

Infine rileviamo che *Mathematica* ogni volta che manipola una qualsiasi espressione algebrica la trasforma automaticamente nella forma che ritiene più semplice¹⁶. Nel fare ciò utilizza, implicitamente, le varie proprietà delle operazioni senza informare l'utente di ciò che sta facendo. Questa caratteristica del software si rivela ovviamente scomoda se la nostra impostazione didattica si basa sul totale controllo, da parte dell'allievo, sulla manipolazione delle espressioni. In particolare nel normale ambiente *Mathematica* sfugge completamente il ruolo giocato dalla proprietà fondamentali delle operazioni (commutativa, distributiva, associativa) nel passare da un'espressione ad un'altra equivalente.

¹⁶ Cosa significa *semplice*? Questa parola in algebra ha un significato decisamente relativo. *Mathematica* tende a trasformare automaticamente ciascuna espressione in forma canonica, o a fattorizzarla, a seconda dei calcoli che sta svolgendo. Ma sono gli autori stessi del software a rilevare questa caratteristica ed a notare che non necessariamente queste due forme sono sempre le più utili o più semplici (Man 305-7). In particolare fanno notare che la forma canonica è comoda per confrontare due espressioni, mentre ridurre a fattori è utile quando si vuole dividere un'espressione.

La sperimentazione

La sperimentazione di L'Algebrista è nata con lo scopo di verificare quanto il software fosse effettivamente utilizzabile in classe; la sua natura di prototipo non ne garantiva il perfetto funzionamento e lasciava prevedere che dall'esperienza con gli studenti avremmo tratto indicazioni per modificarlo strada facendo. Per questi motivi la sperimentazione è stata svolta con la mia costante presenza in classe in qualità di *tecnico* ed organizzatore delle attività; di conseguenza l'ambiente di lavoro non è stato quello della normale interazione alunni insegnante (eravamo due insegnanti), quindi dal punto di vista della ricerca didattica la sperimentazione costituisce un primo livello di validazione del prototipo. D'altro canto il carattere innovativo del software non permetteva di determinare dei problemi didattici precisi da studiare a fondo. Avevamo solo ipotesi generali, ed ipotesi specifiche, riguardo alcune tematiche che analizzeremo in questo capitolo. L'esposizione prosegue con:

- descrizione del percorso;
- analisi del test per quanto riguarda aspetti rilevanti per la sperimentazione;
- individuazione e correzione di misconcezioni degli studenti, dovute ai loro studi passati;
- costruzione di concetti nuovi per gli studenti;
- studio dell'interazione software studente in termini di object-mediation.

Il percorso pianificato e le nostre aspettative

Il percorso pianificato è costituito da tre fasi che andiamo ora a descrivere:

Fase preliminare: questa fase prevedeva la somministrazione agli studenti di un test onde capire che tipo di approccio avessero alle espressioni numeriche, distinguendo sostanzialmente tra approcci *procedurali* ed approcci *relazionali*. Sulla base di questo test avremmo pianificato la prima parte dell'intervento didattico. Con

approccio procedurale (o operativo), ci riferiamo all'interpretazione delle espressioni come entità che hanno senso solo in quanto "vanno calcolate". Secondo un approccio relazionale invece le espressioni possono essere anche confrontate tra di loro, senza necessariamente passare attraverso il calcolo del risultato. Un possibile modo per confrontarle è analizzarne la struttura, in questo caso parleremo di approccio *strutturale*.

Fase delle espressioni numeriche: in seguito ai risultati del test abbiamo stabilito di partire dall'idea di espressioni numeriche come procedimenti di calcolo, puntando ad attività di confronto di espressioni, onde ottenere un'evoluzione, degli studenti, dal procedurale al relazionale. Lo scopo di questa fase era fornire strumenti di calcolo e di pensiero per affrontare le espressioni letterarie. In particolare bisognava affrontare le seguenti tematiche:

- Relazione di equivalenza tra espressioni (procedimenti di calcolo): due espressioni sono equivalenti \Downarrow danno lo stesso risultato \Downarrow è possibile trasformare l'una nell'altra utilizzando le proprietà fondamentali delle operazioni. Prevedevamo di affrontare questa tematica basandoci su attività di confronto tra espressioni.
- Distinguere, e quindi definire, "*verificare come calcolare*" e "*dimostrare come trasformare utilizzando le proprietà*".
- Discutere alcune particolarità del software di rilevanza matematica, ad esempio: perché $a - b$ viene automaticamente trasformata in $a + (-b)$.
- Affrontare e discutere ciascuna delle proprietà delle operazioni.
- Discussioni e produzione di un cartellone che sintetizzasse le scoperte fatte.
- Esercizi paralleli del tipo *aree e pensa un numero*¹⁷ per dare senso al confronto di espressioni e per introdurre gradualmente le lettere.

Fase delle espressioni letterarie

Obiettivo di questa fase è introdurre gli alunni al calcolo letterale, ed all'idea di teoria, in particolare prevede:

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

- Introduzione delle espressioni letterarie come procedimenti di calcolo.
- Costruzione del significato dei termini uguali/equivalenti/diverse, per quanto riguarda le espressioni letterarie. Gli studenti dovranno affrontare diversi modi di stabilire l'equivalenza, o meno, tra espressioni.
- Costruzione ed utilizzo di teoremi, in particolare i prodotti notevoli.

Aspettative

Da questo percorso ci attendevamo che:

- L'approccio degli studenti alle espressioni progredisce da quello procedurale a quello relazionale secondo il percorso: procedurale \rightarrow strutturale \rightarrow relazionale.
- Si appropriassero dell'embrione delle idee di: equivalenza di espressioni; dimostrazione; teoria.
- Acquisissero la capacità di (provare a) scegliere tra le diverse espressioni equivalenti a seconda della situazione.
- Riuscissero ad esplicitare passaggi scontati durante la manipolazione di espressioni.
- Acquisissero la capacità di giudicare il proprio lavoro passando attraverso la capacità di giudicare il lavoro altrui.

Inoltre era presumibile attenderci:

- La formazione del germe del concetto di equazione tramite un approccio anche funzionale alle espressioni.
- Non elevata capacità di calcolo rapido, ma buona capacità di fare e giustificare trasformazioni mirate.

¹⁷ Sono presenti alcuni esempi nella sezione Parco Giochi

Il percorso effettivo

Il percorso effettivo ha seguito sostanzialmente quello pianificato, ma ha ovviamente subito delle variazioni legate alle reazioni della classe. In particolare, non le discussioni svolte in classe non sono state approfondite quanto avremmo voluto; inoltre è da rilevare che le previste attività parallele da svolgere nelle normali lezioni in classe non sono state svolte. Questo significa che in pratica la sperimentazione è avvenuta solo e solamente in ambito laboratorio di informatica.

Dopo il test iniziale (la cui trattazione tralasciamo per il momento) abbiamo fatto una lezione preliminare in cui i ragazzi prendevano confidenza con il programma.

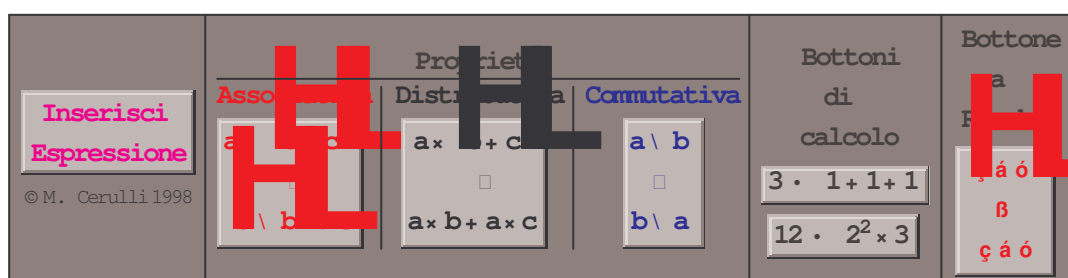


Fig. 1: la paletta iniziale

Durante la stessa è stata discussa la questione delle sottrazioni che venivano trasformate in somme dal programma quando veniva usato il bottone *inserisci espressione*.

Durante la prima fase della sperimentazione i ragazzi hanno lavorato con una paletta (Fig. 1) contenente solo le proprietà di base ed i due bottoni di calcolo. Questi ultimi due bottoni sono stati introdotti durante la prima lezione, in cui abbiamo chiesto ai ragazzi di *calcolare* un'espressione (vedi CL 1. Appendice I¹⁸).

Successivamente sono state introdotte una ad una tutte e tre le proprietà, sempre avvalendoci di consegne in cui i ragazzi dovevano confrontare due o più espressioni (CL 2–6.). Parallelamente somministravamo loro degli esercizi per casa di due tipi: il confronto di espressioni (CS 2,3) e la ricostruzione di una serie di trasformazioni

¹⁸ Tutte le attività svolte sono raccolte nell'appendice in questione, d'ora in avanti scriveremo solo la sigla che rappresenta l'attività.

effettuate su un'espressione di partenza (CS 1,4). In entrambe i casi ai ragazzi era richiesto di esprimersi in termini dei bottoni o delle proprietà usate¹⁹.

Osserviamo che, come è reso anche dalla grafica²⁰, i bottoni di calcolo e quelli delle proprietà hanno uno status completamente diverso; anche nelle attività svolte, quelli delle proprietà sono diventati i bottoni *per dimostrare*, mentre quelli di calcolo sono diventati i bottoni *per verificare*.

Rispetto alle nostre aspettative, purtroppo è saltata la creazione di un cartellone che ufficializzasse le scoperte fatte in classe.

Dopo alcune settimane i ragazzi erano in grado di usare le proprietà per trasformare espressioni, e, almeno a parole, sapevano distinguere i vari passaggi di una trasformazione. A questo punto siamo passati alle espressioni letterali.

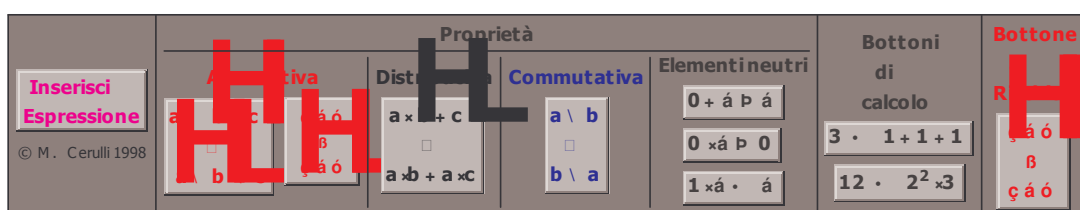


Fig. 2: seconda paletta

Ai ragazzi è stata presentata una nuova paletta (Fig. 2) contenente anche le proprietà degli elementi neutri; i bottoni in questione sono raggruppati, come si può notare (fig. 2), insieme alle proprietà poiché vanno, insieme ad esse, a costituire gli assiomi della teoria su cui lavorare.

La prima consegna (CL 9.) riguardante le espressioni letterarie ne presenta tre come procedimenti di calcolo per ottenere una stessa area; ai ragazzi era richiesto di investigarne la correttezza. L'approccio dei ragazzi a questo esercizio è stato vario, alcuni si sono appoggiati a ragionamenti geometrici, altri hanno sostituito dei numeri alle lettere, ed altri ancora si sono avvalsi subito delle proprietà cercando di trasformare le espressioni le une nelle altre (alcuni di loro lo hanno fatto a mano). Tutti quanti però, per dare una risposta definitiva, si sono avvalsi del computer onde produrre una *dimostrazione* basata sulle proprietà. Durante questa ultima fase i

¹⁹ Gli esercizi in questione sono tutti riportati nel Parco Giochi

²⁰ Vedi capitolo 3.

ragazzi sono incappati nella necessità di insegnare al computer che $b \cdot b - b \cdot b = 0$ dimostrando così il loro primo teorema.

Nella lezione successiva è stato discusso in classe cosa significa cosa significa che due espressioni letterarie sono equivalenti, ed infine, è stato introdotto un nuovo bottone (fig. 3) che facesse ciò che i ragazzi avevano *dimostrato*²¹ la volta precedente. Osserviamo che l'idea di aver dimostrato un teorema sembra aver gratificato molto i ragazzi.



Fig. 3: la paletta dei teoremi

La sperimentazione è proseguita con una serie di attività incentrate sulla costruzione di un sistema teorico avente i suoi assiomi, rappresentati dalla paletta delle proprietà, ed i suoi teoremi, rappresentati dalla paletta dei teoremi e dai nuovi bottoni prodotti con il Teorematore. In particolare, sono stati prodotti bottoni rappresentanti i prodotti notevoli, i quali sono stati a loro volta utilizzati in attività tese all'introduzione dell'idea di fattorizzazione.

²¹ In questo caso con *dimostrato* intendiamo che hanno prodotto delle catene di espressioni equivalenti giustificando ogni passaggio in base alle proprietà delle operazioni, ovvero in base agli assiomi.

Osservazioni sulla sperimentazione

Sin dal test preliminare abbiamo potuto osservare una certa confusione per quanto riguarda le proprietà delle operazioni. Tale confusione è sia a livello verbale che concettuale ed è dovuta soprattutto all'ambiguità di certi termini e delle relative definizioni. Inizieremo questo capitolo andando ad analizzare le situazioni riscontrate per ciascuna delle proprietà.

Nella seconda parte del capitolo riporteremo un'interpretazione in termini di "*object mediation*" di vari tipi di progressi che abbiamo potuto riscontrare nei protocolli prodotti dagli studenti. In particolare cercheremo di distinguere e mettere in evidenza il ruolo svolto da L'Algebrista e dalle peculiari attività svolte.

Confusione a livello puramente verbale per quanto riguarda le proprietà fondamentali delle operazioni

Proprietà commutativa e proprietà associativa

La proprietà commutativa viene in genere presentata senza formule, ma "a parole": "*scambiando l'ordine ... il risultato non cambia*". Questo sembra generare una certa confusione in quanto, alcuni studenti, al posto dei puntini, inseriscono, talvolta inconsciamente, le parole "*delle operazioni*" invece di "*degli addendi (fattori)*", ottenendo la frase: "*scambiando l'ordine delle operazioni il risultato non cambia*". Di conseguenza, talvolta, i ragazzi si appellano alla proprietà commutativa, ma in realtà applicano la proprietà associativa. In effetti la scrittura $2+(3+4)$, vista come procedimento di calcolo, risulta equivalente alla scrittura $(2+3)+4$ in quanto "*scambiando l'ordine delle operazioni il risultato non cambia*": nel primo caso si esegue prima la somma $3+4$ e poi $2+7$, mentre nel secondo caso prima si esegue la somma $2+3$ e poi $5+4$. Riportiamo alcune risposte date dai ragazzi relativamente alla prima parte del test preliminare:

Protocollo n. 23:

$$17 + (6 + 9) = (17 + 6) + 9$$

È corretta perché cambiando l'ordine delle operazioni il risultato non cambia

$$6 + 2 \cdot (4 \cdot 5) = 6 + (2 \cdot 4) \cdot 5$$

È corretta perché se non si cambiano i segni e i numeri è indifferente fare prima una moltiplicazione o l'altra

Protocollo n.15:

$$17 + (6 + 9) = (17 + 6) + 9$$

È giusta perché l'addizione gode della proprietà commutativa, quindi si può tranquillamente cambiare l'ordine dei numeri. In questo caso, perciò le parentesi non hanno importanza

$$6 + 2 \cdot (4 \cdot 5) = 6 + (2 \cdot 4) \cdot 5$$

Come nel precedente è giusta perché anche la moltiplicazione gode della proprietà commutativa. Basta lasciare invariate le operazioni diverse (in questo caso le addizioni)

Protocollo n. 10:

$$17 + (6 + 9) = (17 + 6) + 9$$

CORRETTA perché l'addizione gode della proprietà associativa

$$6 + 2 \cdot (4 \cdot 5) = 6 + (2 \cdot 4) \cdot 5$$

CORRETTA perché l'addizione e la moltiplicazione godono della proprietà commutativa

Questo comportamento può però trovare delle contraddizioni quando ci sono, nella stessa espressione, sia moltiplicazioni che addizioni, in tale caso il ragionamento entra in crisi dato che *scambiare l'ordine delle operazioni* significherebbe scambiare la somma con la moltiplicazione. Di conseguenza può capitare che scritture giuste vengano considerate sbagliate:

Protocollo n. 25:

$$17 + (6 + 9) = (17 + 6) + 9$$

CORRETTA perché cambiando l'ordine degli addendi il risultato non cambia

$$6 + 2 \cdot (4 \cdot 5) = 6 + (2 \cdot 4) \cdot 5$$

SBAGLIATA perché vengono cambiate le cifre che fanno parte e della moltiplicazione e quelle che fanno parte dell'addizione

Proprietà associativa e proprietà distributiva

La parola *associare* porta con sé, dal linguaggio naturale, il significato di “*mettere insieme, accoppiare*”. Si crea perciò una situazione di ambiguità tra linguaggio naturale e linguaggio matematico, che risulta difficile da gestire, soprattutto quando le spiegazioni dell'insegnante si muovono a livello operativo, descrivendo a parole

ciò che si deve fare. Questa ambiguità, in ambito matematico, può far sì che la proprietà associativa e quella distributiva vengano confuse in quanto il passaggio $3 \cdot 4 + 3 \cdot 5 \rightarrow 3 \cdot (4 + 5)$ può essere interpretato "associa il 4 ed il 5", mentre, viceversa, il passaggio $3 \cdot (4 + 5) \rightarrow 3 \cdot 4 + 3 \cdot 5$ può essere interpretato "associa il 3 al 4 ed al 5". Abbiamo potuto constatare sia nel test preliminare che durante le prime settimane di sperimentazione, che questo tipo di fraintendimento è piuttosto frequente e più difficile da debellare rispetto a quello tra le proprietà associativa e commutativa.

Protocollo n. 21:

$$17 + (6 + 9) = (17 + 6) + 9$$

Corretta perché è applicata la distributiva

$$6 + 2 \cdot (4 \cdot 5) = 6 + (2 \cdot 4) \cdot 5$$

Corretta perché è applicata la distributiva

Protocollo n. 6:

$$8 + 9 \cdot (3 + 2) - 17 = 8 + 27 + 18 - 17$$

Si è giusta perché vale la proprietà associativa

$$3 + 6 \cdot 73 + 6 \cdot 8 + 13 = 3 + 6 \cdot (73 + 8) + 13$$

Si è uguale perché vale la proprietà associativa

Confusione a livello concettuale per quanto riguarda le proprietà delle operazioni

La proprietà associativa

Aldilà della confusione a livello verbale appena descritta, è stato possibile osservare che la proprietà associativa (che sia chiamata con il suo vero nome oppure no) sembra essere facilmente riconosciuta nel caso dell'addizione, ma non nel caso della moltiplicazione; Il seguente protocollo (sempre relativo al test) mostra un esempio di questo fenomeno:

Protocollo n. 6:

$$17 + (6 + 9) = (17 + 6) + 9$$

Si è uguale perché vale la proprietà associativa

$$6 + 2 \cdot (4 \cdot 5) = 6 + (2 \cdot 4) \cdot 5$$

No, non è giusta

Si può notare come nel caso della moltiplicazione la proprietà associativa non viene riconosciuta.

Proprietà distributiva

Per quanto riguarda la proprietà distributiva la situazione è più complessa, essa raramente ha i caratteri della relazione di equivalenza, piuttosto assume due diverse interpretazioni, entrambe operative, una che permettere di *distribuire* e l'altra che permette di *raccogliere*. Tale scissione è generalmente rafforzata, e comunque caratterizzata, dall'uso di due termini distinti per indicare i due *versi* della proprietà in questione. Questo è perfettamente in linea con quanto si trova nei libri di testo, dove troviamo anche parole come "proprietà invariantiva". Come esempio riportiamo il protocollo 15, relativo al test (consegna T 1):

Protocollo n. 15:

$$15 + 6 \cdot 4 + 19 \cdot 4 + 11 = 15 + (6 + 19) \cdot 4 + 11$$

È giusta perché è stata applicata la proprietà invariantiva

$$8 + 9 \cdot (3+2) - 17 = 8 + 27 + 18 - 17$$

È giusta perché alla seconda parte dell'uguaglianza è stata applicata la proprietà distributiva

$$3 + 6 \cdot 73 + 6 \cdot 8 + 13 = 3 + 6 \cdot (73 + 8) + 13$$

È sbagliata

In altri casi invece sembra essere riconosciuto un solo *verso* della proprietà, anche a seconda che si tratti di distributiva destra o sinistra:

Protocollo n. 7:

$$15 + 6 \cdot 4 + 19 \cdot 4 + 11 = 15 + (6 + 19) \cdot 4 + 11$$

È sbagliata

$$8 + 9 \cdot (3+2) - 17 = 8 + 27 + 18 - 17$$

È corretta per la proprietà distributiva

$$3 + 6 \cdot 73 + 6 \cdot 8 + 13 = 3 + 6 \cdot (73 + 8) + 13$$

Si è corretta

Protocollo n. 3:

$$15 + 6 \cdot 4 + 19 \cdot 4 + 11 = 15 + (6 + 19) \cdot 4 + 11$$

È sbagliata perché si moltiplica per 4 sia il 6 che il 19 e non il 6+19

$$8 + 9 \cdot (3+2) - 17 = 8 + 27 + 18 - 17$$

È sbagliata perché prima va calcolata la parentesi

$$3 + 6 \cdot 73 + 6 \cdot 8 + 13 = 3 + 6 \cdot (73 + 8) + 13$$

È corretta se si applica la proprietà distributiva

Protocollo n. 8:

$$15 + 6 \cdot 4 + 19 \cdot 4 + 11 = 15 + (6 + 19) \cdot 4 + 11$$

È sbagliata perché secondo me il procedimento è il seguente:

$$15 + 24 + 76 + 11 = 126$$

$$8 + 9 \cdot (3+2) - 17 = 8 + 27 + 18 - 17$$

Secondo me è giusta perché si applica la proprietà distributiva

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

$$3 + 6 \cdot 73 + 6 \cdot 8 + 13 = 3 + 6 \cdot (73 + 8) + 13$$

Secondo me è sbagliata, perché il procedimento dovrebbe essere

$$3+438+48+13$$

Evoluzione del concetto di proprietà: alcuni protocolli esemplari

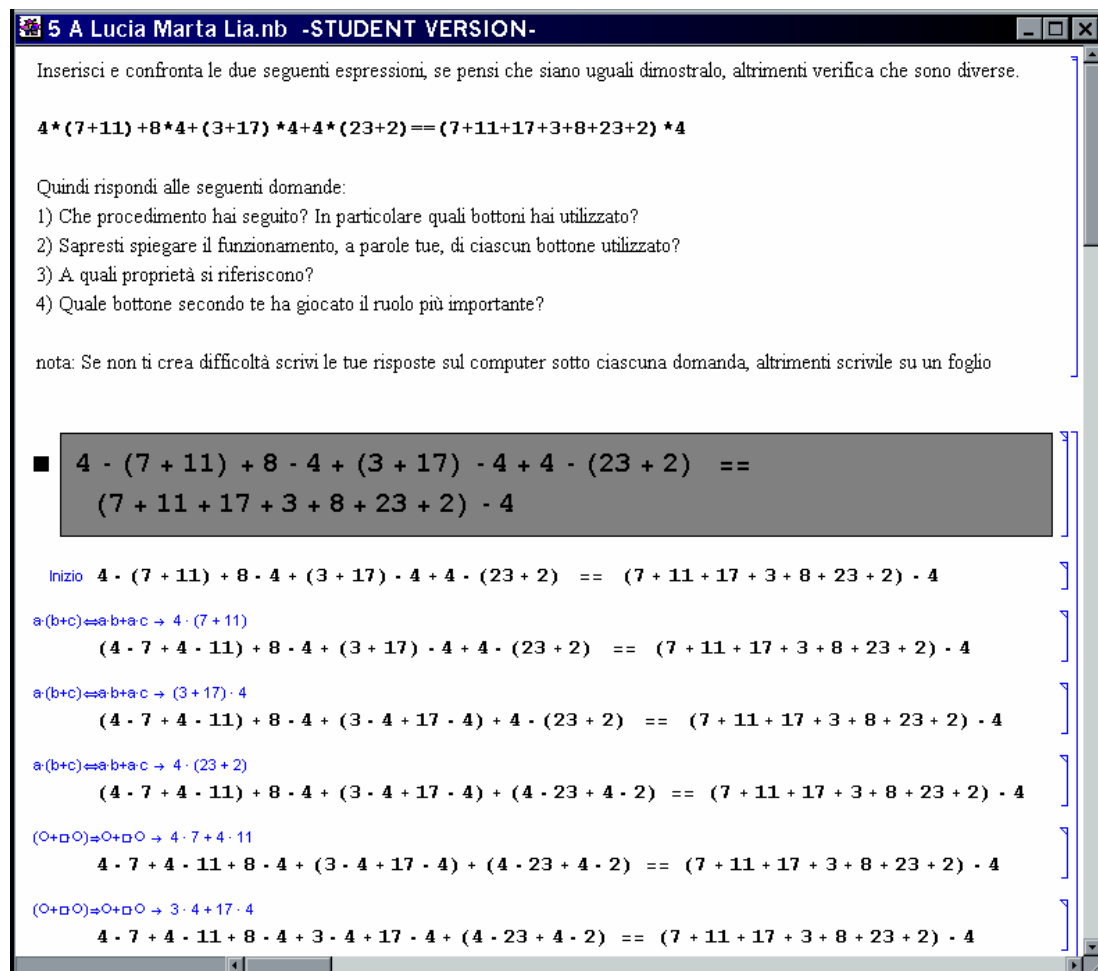
La confusione verbale e quella concettuale, di cui abbiamo discusso in queste ultime due sezioni, sono, in effetti, scomparse dopo qualche settimana di sperimentazione. La nostra ipotesi è che questo sia avvenuto grazie all'uso di L'Algebrista e dei suoi bottoni, grazie ai quali ogni proprietà risulta ben distinta e ben rappresentata grazie alla doppia simbologia usata, quella matematica e quella verbale. Questa osservazione, come molte delle altre che riporteremo in questo capitolo, va interpretata come indicazione per eventuali ricerche future.

Analizziamo ora alcuni protocolli che sono esemplari sia delle attività proposte che dei comportamenti che possiamo attenderci. I protocolli sono relativi all'attività CL 5, svolta in laboratorio con L'Algebrista, quindi ciascuno di essi è stato prodotto dalla collaborazione di due o tre studenti. Ogni protocollo è costituito da una parte scritta nella modalità *editore di testo*, che spesso contiene affermazioni degli studenti intrecciate con la consegna, e da una parte realizzata nell'ambiente algebrico di L'Algebrista, costituita da una serie di trasformazioni algebriche.

Cominciamo riportando l'analisi di ciascun protocollo, seguita dagli stessi.

Lucia Marta e Lia

In una fase preliminare riescono ad arrivare, per tentativi, a trasformare la prima espressione in un'espressione senza parentesi; seguono ancora dei tentativi per ripristinare le parentesi al posto giusto. Di fronte alla difficoltà c'è un brusco cambiamento di strategia, che riporta verso un metodo più familiare, ovvero si usano i bottoni di calcolo e si verifica l'uguaglianza dei risultati. Il livello di consapevolezza in questo caso è ancora basso: mancano i commenti e/o le risposte alle domande 1), 2), 3), 4).



Inserisci e confronta le due seguenti espressioni, se pensi che siano uguali dimostralolo, altrimenti verifica che sono diverse.

$$4 * (7 + 11) + 8 * 4 + (3 + 17) * 4 + 4 * (23 + 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) * 4$$

Quindi rispondi alle seguenti domande:

- 1) Che procedimento hai seguito? In particolare quali bottoni hai utilizzato?
- 2) Sapresti spiegare il funzionamento, a parole tue, di ciascun bottone utilizzato?
- 3) A quali proprietà si riferiscono?
- 4) Quale bottone secondo te ha giocato il ruolo più importante?

nota: Se non ti crea difficoltà scrivi le tue risposte sul computer sotto ciascuna domanda, altrimenti scrivile su un foglio

$$4 * (7 + 11) + 8 * 4 + (3 + 17) * 4 + 4 * (23 + 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) * 4$$

Inizio $4 * (7 + 11) + 8 * 4 + (3 + 17) * 4 + 4 * (23 + 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) * 4$

$a(b+c) \Leftrightarrow a*b+a*c \rightarrow 4 * (7 + 11)$

$$(4 * 7 + 4 * 11) + 8 * 4 + (3 + 17) * 4 + 4 * (23 + 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) * 4$$

$a(b+c) \Leftrightarrow a*b+a*c \rightarrow (3 + 17) * 4$

$$(4 * 7 + 4 * 11) + 8 * 4 + (3 * 4 + 17 * 4) + 4 * (23 + 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) * 4$$

$a(b+c) \Leftrightarrow a*b+a*c \rightarrow 4 * (23 + 2)$

$$(4 * 7 + 4 * 11) + 8 * 4 + (3 * 4 + 17 * 4) + (4 * 23 + 4 * 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) * 4$$

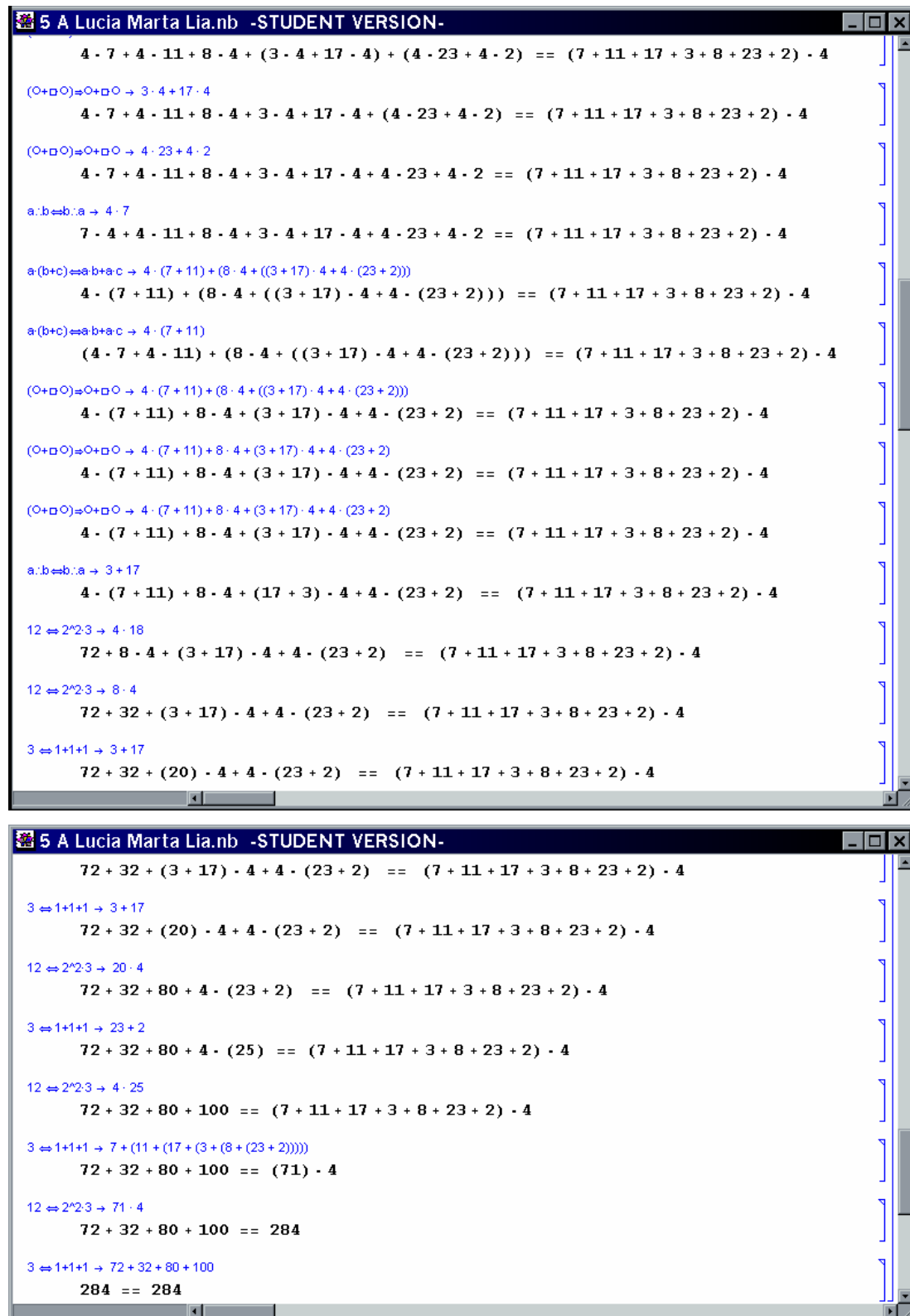
$(O+O) \Rightarrow O+O \rightarrow 4 * 7 + 4 * 11$

$$4 * 7 + 4 * 11 + 8 * 4 + (3 * 4 + 17 * 4) + (4 * 23 + 4 * 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) * 4$$

$(O+O) \Rightarrow O+O \rightarrow 3 * 4 + 17 * 4$

$$4 * 7 + 4 * 11 + 8 * 4 + 3 * 4 + 17 * 4 + (4 * 23 + 4 * 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) * 4$$

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.



Alessandro e Filippo

Notiamo un solo passaggio, in cui si applica la proprietà distributiva, poi una serie di calcoli.

Dai commenti si può pensare che il significato delle proprietà sia chiaro, ma dalla soluzione proposta è chiaro che il senso delle proprietà, nel quadro delle trasformazioni di un'espressione, non è stato colto.

5 B AlessandroFilippo2.nb -STUDENT VERSION-

Inserisci e confronta le due seguenti espressioni, se pensi che siano uguali dimostrarlo, altrimenti verifica che sono diverse.

$$4*(7+11) + 8*4 + (3+17) * 4 + 4*(23+2) == (7+11+17+3+8+23+2) * 4$$

Quindi rispondi alle seguenti domande:

- 1) Che procedimento hai seguito? In particolare quali bottoni hai utilizzato?
-Abbiamo lavorato su questa espressione usando i bottoni di calcolo e quelli delle proprietà.
- 2) Sapresti spiegare il funzionamento, a parole tue, di ciascun bottone utilizzato?
-Uno dei bottoni che abbiamo usato è quella della PROPR.DISTRIBUTIVA: questa proprietà serve per visualizzare in maniera più rapida la moltiplicazione tra un numero intero e la somma di due numeri interi.
La seconda è quella della PROPR. COMMUTATIVA: questa ha la funzione che, scambiando l'ordine degli addendi o dei prodotti il risultato non cambia.
L'ultima è quella della PROPR. ASSOCIATIVA: questa serve per cambiare l'ordine di due prodotti, uno tra parentesi ed uno all'esterno.
Infine, i bottoni di calcolo servono, o per sommare più numeri o per scomporre in fattori primi un numero
- 3) A quali proprietà si riferiscono?
-A quella ASSOCIATIVA, DISTRIBUTIVA e COMMUTATIVA.
- 4) Quale bottone secondo te ha giocato il ruolo più importante?
-I due bottoni di calcolo.

Nota: Se non ti crea difficoltà scrivi le tue risposte sul computer sotto ciascuna domanda, altrimenti scrivile su un foglio

5 B AlessandroFilippo2.nb -STUDENT VERSION-

■ $4 - (7 + 11) + 8 - 4 + (3 + 17) - 4 + 4 - (23 + 2) ==$
 $(7 + 11 + 17 + 3 + 8 + 23 + 2) - 4$

$\ln[1]= 4 \cdot (7 + 11) + 8 \cdot 4 + (3 + 17) \cdot 4 + 4 \cdot (23 + 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) \cdot 4$

$a(b+c) \Rightarrow a \cdot b + a \cdot c \rightarrow 4 \cdot (7 + 11)$
 $(4 \cdot 7 + 4 \cdot 11) + 8 \cdot 4 + (3 + 17) \cdot 4 + 4 \cdot (23 + 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) \cdot 4$

$3 \Rightarrow 1+1+1 \rightarrow 7+11$
 $4 \cdot (18) + 8 \cdot 4 + (3 + 17) \cdot 4 + 4 \cdot (23 + 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) \cdot 4$

$3 \Rightarrow 1+1+1 \rightarrow 3+17$
 $4 \cdot (18) + 8 \cdot 4 + (20) \cdot 4 + 4 \cdot (23 + 2) == (7 + 11 + 17 + 3 + 8 + 23 + 2) \cdot 4$

$3 \Rightarrow 1+1+1 \rightarrow 23+2$
 $4 \cdot (18) + 8 \cdot 4 + (20) \cdot 4 + 4 \cdot (25) == (7 + 11 + 17 + 3 + 8 + 23 + 2) \cdot 4$

$3 \Rightarrow 1+1+1 \rightarrow 7+(11+(17+(3+(8+(23+2))))))$
 $4 \cdot (18) + 8 \cdot 4 + (20) \cdot 4 + 4 \cdot (25) == (71) \cdot 4$

$12 \Rightarrow 2^2 \cdot 3 \rightarrow 4 \cdot 18$
 $72 + 8 \cdot 4 + (20) \cdot 4 + 4 \cdot (25) == (71) \cdot 4$

$12 \Rightarrow 2^2 \cdot 3 \rightarrow 8 \cdot 4$
 $72 + 32 + (20) \cdot 4 + 4 \cdot (25) == (71) \cdot 4$

$(0+\square \square) \Rightarrow 0+\square \square \rightarrow 20$
 $72 + 32 + 20 \cdot 4 + 4 \cdot (25) == (71) \cdot 4$

$12 \Rightarrow 2^2 \cdot 3 \rightarrow 20 \cdot 4$
 $72 + 32 + 80 + 4 \cdot (25) == (71) \cdot 4$

Anna

La soluzione è sostanzialmente corretta. È possibile, qui, vedere l'evoluzione rispetto al test: le motivazioni della correttezza dei passaggi sono adesso date in termini di proprietà.

Da notare il permanere di "vecchi schemi": *"proprietà distributiva: il numero che moltiplica viene distribuito ..."*.

5 C anna.nb -STUDENT VERSION-

Inserisci e confronta le due seguenti espressioni, se pensi che siano uguali dimostralolo, altrimenti verifica che sono diverse.

$$4 \cdot (7+11) + 8 \cdot 4 + (3+17) \cdot 4 + 4 \cdot (23+2) == (7+11+17+3+8+23+2) \cdot 4$$

Quindi rispondi alle seguenti domande:

- 1) Che procedimento hai seguito? In particolare quali bottoni hai utilizzato?
- 2) Sapresti spiegare il funzionamento, a parole tue, di ciascun bottone utilizzato?
- 3) A quali proprietà si riferiscono?
- 4) Quale bottone secondo te ha giocato il ruolo più importante?

nota: Se non ti crea difficoltà scrivi le tue risposte sul computer sotto ciascuna domanda, altrimenti scrivile su un foglio

■ $4 \cdot (7 + 11) + 8 \cdot 4 + (3 + 17) \cdot 4 + 4 \cdot (23 + 2)$

Inizio $4 \cdot (7 + 11) + 8 \cdot 4 + (3 + 17) \cdot 4 + 4 \cdot (23 + 2)$

$a(b+c) \Leftrightarrow a \cdot b + a \cdot c \rightarrow 4 \cdot (7 + 11)$
 $(4 \cdot 7 + 4 \cdot 11) + 8 \cdot 4 + (3 + 17) \cdot 4 + 4 \cdot (23 + 2)$

$a(b+c) \Leftrightarrow a \cdot b + a \cdot c \rightarrow (3 + 17) \cdot 4$
 $(4 \cdot 7 + 4 \cdot 11) + 8 \cdot 4 + (3 \cdot 4 + 17 \cdot 4) + 4 \cdot (23 + 2)$

$a(b+c) \Leftrightarrow a \cdot b + a \cdot c \rightarrow 4 \cdot (23 + 2)$
 $(4 \cdot 7 + 4 \cdot 11) + 8 \cdot 4 + (3 \cdot 4 + 17 \cdot 4) + (4 \cdot 23 + 4 \cdot 2)$

$(O+\square O) \Rightarrow O+\square O \rightarrow 4 \cdot 7 + 4 \cdot 11 + 8 \cdot 4 + (3 \cdot 4 + 17 \cdot 4) + (4 \cdot 23 + 4 \cdot 2)$
 $4 \cdot 7 + 4 \cdot 11 + 8 \cdot 4 + 3 \cdot 4 + 17 \cdot 4 + 4 \cdot 23 + 4 \cdot 2$

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

```

5 C anna.nb -STUDENT VERSION-

(O+O)O=O+O → 4·7+4·11+8·4+(3·4+17·4)+(4·23+4·2)
      4·7+4·11+8·4+3·4+17·4+4·23+4·2

a:b=b:a → 4·7+4·11
      (4·11+4·7)+8·4+(3·4+17·4)+(4·23+4·2)

(O+O)O=O+O → 4·11+4·7+8·4+(3·4+17·4)+(4·23+4·2)
      4·11+4·7+8·4+3·4+17·4+4·23+4·2

a:b=b:a → 3·4+17·4
      (4·11+4·7)+8·4+(17·4+3·4)+(4·23+4·2)

(O+O)O=O+O → 4·11+4·7+8·4+(17·4+3·4)+(4·23+4·2)
      4·11+4·7+8·4+17·4+3·4+4·23+4·2

a(b+c)=a(b+a) → 4·7+8·4
      4·11+4·7+8·4+17·4+3·4+4·23+4·2

a:b=b:a → 4·11
      11·4+4·7+8·4+17·4+3·4+4·23+4·2

a:b=b:a → 4·7
      11·4+7·4+8·4+17·4+3·4+4·23+4·2

a:b=b:a → 4·23
      11·4+7·4+8·4+17·4+3·4+23·4+4·2

a:b=b:a → 4·2
      11·4+7·4+8·4+17·4+3·4+23·4+2·4

a(b+c)=a(b+a) → 11·4+7·4+8·4+17·4+3·4+23·4+2·4
      (11+(7+(8+(17+(3+(23+2))))))·4

```

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

```

5 C anna.nb -STUDENT VERSION-
a(b+c)⇔a+b+a·c → 11·4+7·4+8·4+17·4+3·4+23·4+2·4
(11 + (7 + (8 + (17 + (3 + (23 + 2)))))) · 4

(O+□O)⇔O+□O → (11+(7+(8+(17+(3+(23+2))))))·4
(11 + (7 + (8 + (17 + (3 + (23 + 2)))))) · 4

(O+□O)⇔O+□O → 7+(8+(17+(3+(23+2))))
(11 + 7 + 8 + 17 + 3 + 23 + 2) · 4

■ (7 + 11 + 17 + 3 + 8 + 23 + 2) · 4

Inizio (7 + 11 + 17 + 3 + 8 + 23 + 2) · 4

3 ⇔ 1+1+1 → 7+(11+(17+(3+(8+(23+2))))))
(71) · 4

(O+□O)⇔O+□O → 71
71 · 4

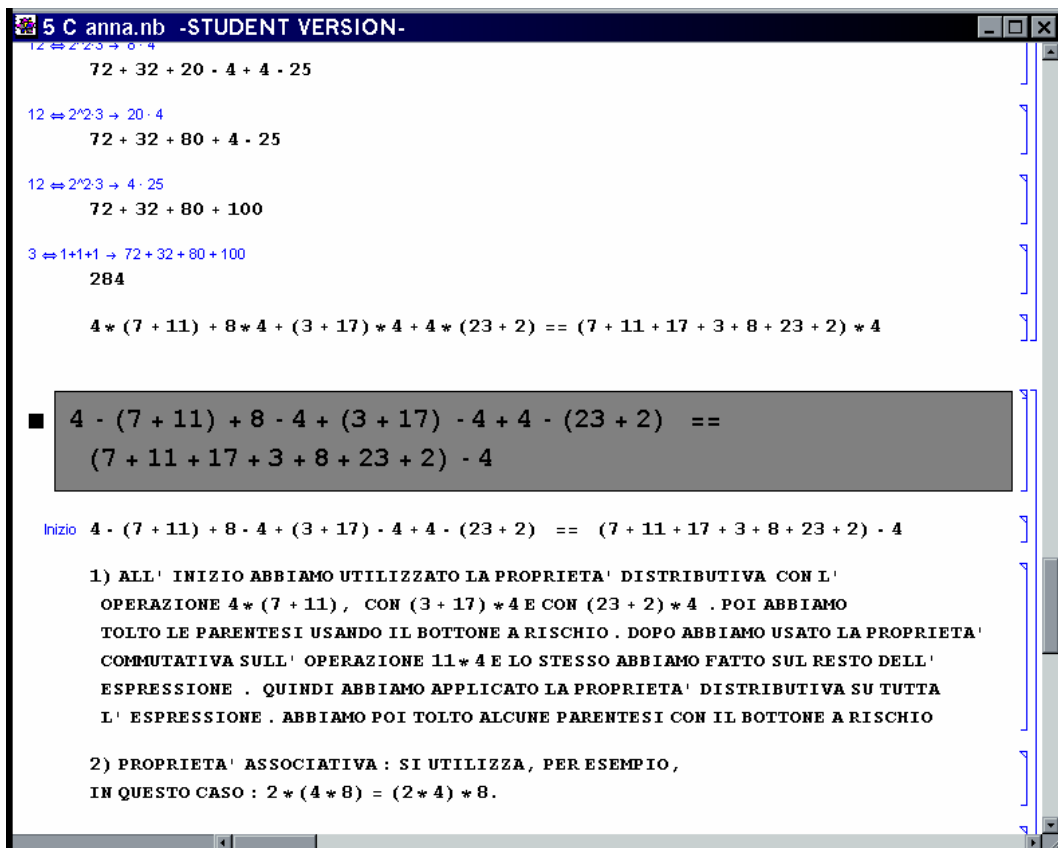
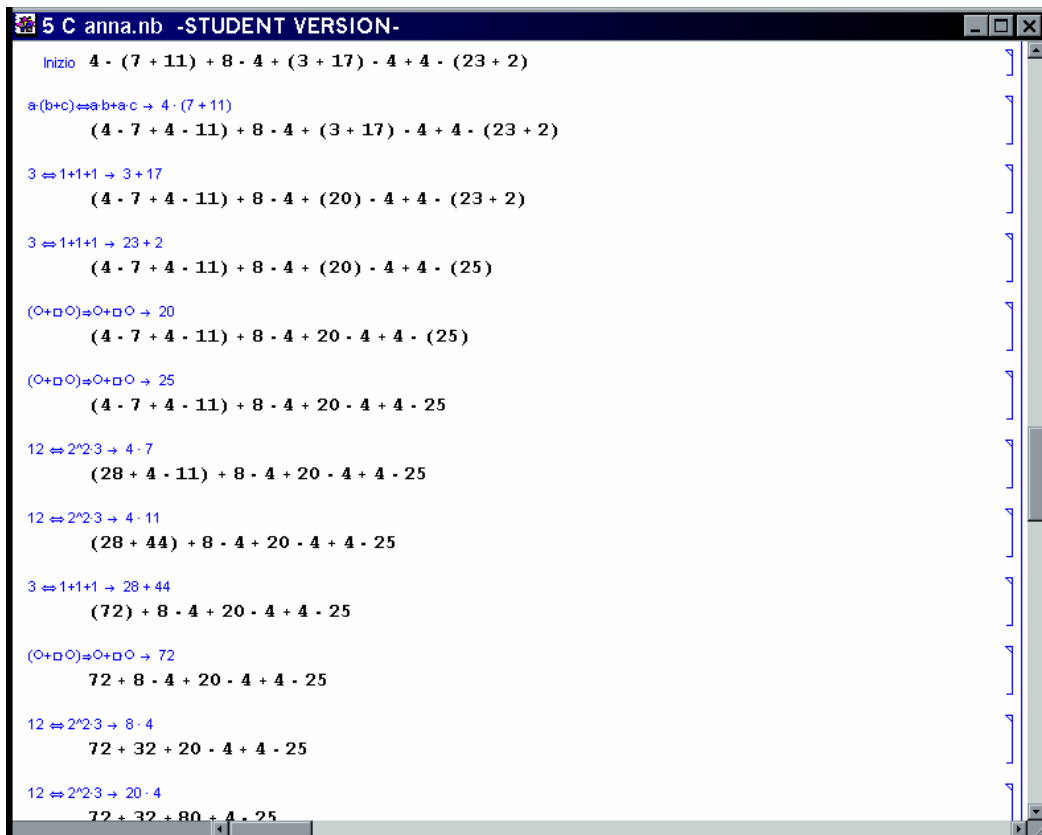
12 ⇔ 2^2·3 → 71 · 4
284

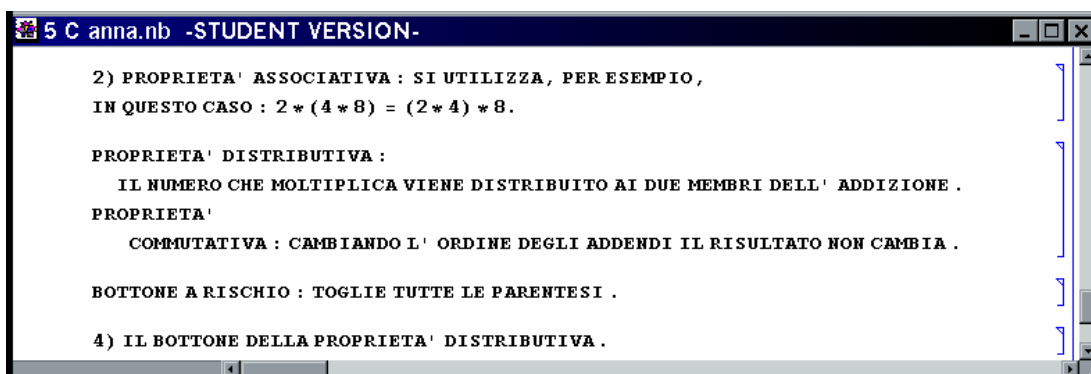
■ 4 - (7 + 11) + 8 - 4 + (3 + 17) - 4 + 4 - (23 + 2)

Inizio 4 - (7 + 11) + 8 - 4 + (3 + 17) - 4 + 4 - (23 + 2)

a(b+c)⇔a+b+a·c → 4·(7+11)
(4·7 + 4·11) + 8 - 4 + (3 + 17) - 4 + 4 - (23 + 2)

```

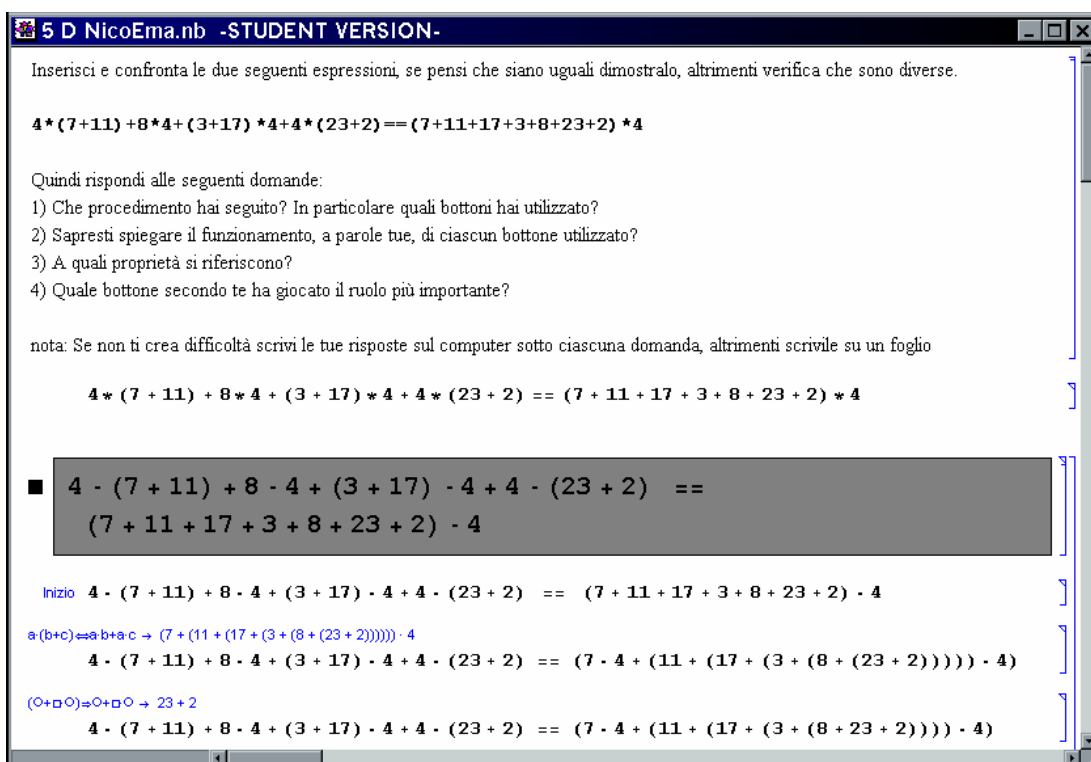




Nicola e Emanuele

Interessante perché trasforma tutte e due le espressioni e le porta ad una forma che non p quella canonica.

Anche le risposte alle domande 1), 2), 3), 4), sono di buon livello. Notare come viene sottovalutata la proprietà commutativa. In generale la proprietà che si ritiene più importante è la distributiva.



Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

```

5 D NicoEma.nb -STUDENT VERSION-
(O+□O)⇒O+□O ⇒ 23+2
4 · (7 + 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == (7 · 4 + (11 + (17 + (3 + (8 + 23 + 2)))) · 4)

(O+□O)⇒O+□O ⇒ 8+23+2
4 · (7 + 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == (7 · 4 + (11 + (17 + (3 + 8 + 23 + 2)))) · 4)

(O+□O)⇒O+□O ⇒ 3+8+23+2
4 · (7 + 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == (7 · 4 + (11 + (17 + 3 + 8 + 23 + 2)))) · 4)

(O+□O)⇒O+□O ⇒ 17+3+8+23+2
4 · (7 + 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == (7 · 4 + (11 + 17 + 3 + 8 + 23 + 2) · 4)

a:b⇒b:a ⇒ 7 · 4
4 · (7 + 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == (4 · 7 + (11 + 17 + 3 + 8 + 23 + 2) · 4)

(O+□O)⇒O+□O ⇒ 4 · 7 + (11 + 17 + 3 + 8 + 23 + 2) · 4
4 · (7 + 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == 4 · 7 + (11 + 17 + 3 + 8 + 23 + 2) · 4

a(b+c)⇒a b+a c ⇒ (11 + (17 + (3 + (8 + (23 + 2))))) · 4
4 · (7 + 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == 4 · 7 + (11 · 4 + (17 + (3 + (8 + (23 + 2))))) · 4)

a(b+c)⇒a b+a c ⇒ 4 · (7 + 11)
(4 · 7 + 4 · 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) ==
4 · 7 + (11 · 4 + (17 + (3 + (8 + (23 + 2))))) · 4)

a:b⇒b:a ⇒ 11 · 4
(4 · 7 + 4 · 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) ==
4 · 7 + (4 · 11 + (17 + (3 + (8 + (23 + 2))))) · 4)

(O+□O)⇒O+□O ⇒ 4 · 7 + (4 · 11 + (17 + (3 + (8 + (23 + 2))))) · 4
(4 · 7 + 4 · 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == 4 · 7 + (4 · 11 + (17 + (3 + (8 + (23 + 2))))) · 4)

```

```

5 D NicoEma.nb -STUDENT VERSION-
(O+□O)⇒O+□O ⇒ 4 · 7 + (4 · 11 + (17 + (3 + (8 + (23 + 2))))) · 4
(4 · 7 + 4 · 11) + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == 4 · 7 + 4 · 11 + (17 + (3 + (8 + (23 + 2))))) · 4

(O+□O)⇒O+□O ⇒ 4 · 7 + 4 · 11
4 · 7 + 4 · 11 + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == 4 · 7 + 4 · 11 + (17 + (3 + (8 + (23 + 2))))) · 4

(O+□O)⇒O+□O ⇒ 17 + (3 + (8 + (23 + 2)))
4 · 7 + 4 · 11 + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) == 4 · 7 + 4 · 11 + (17 + 3 + 8 + 23 + 2) · 4

a(b+c)⇒a b+a c ⇒ (17 + 3 + 8 + 23 + 2) · 4
4 · 7 + 4 · 11 + 8 · 4 + (3 + 17) · 4 + 4 · (23 + 2) ==
4 · 7 + 4 · 11 + (17 · 4 + 3 · 4 + 8 · 4 + 23 · 4 + 2 · 4)

a(b+c)⇒a b+a c ⇒ (3 + 17) · 4
4 · 7 + 4 · 11 + 8 · 4 + (3 · 4 + 17 · 4) + 4 · (23 + 2) ==
4 · 7 + 4 · 11 + (17 · 4 + 3 · 4 + 8 · 4 + 23 · 4 + 2 · 4)

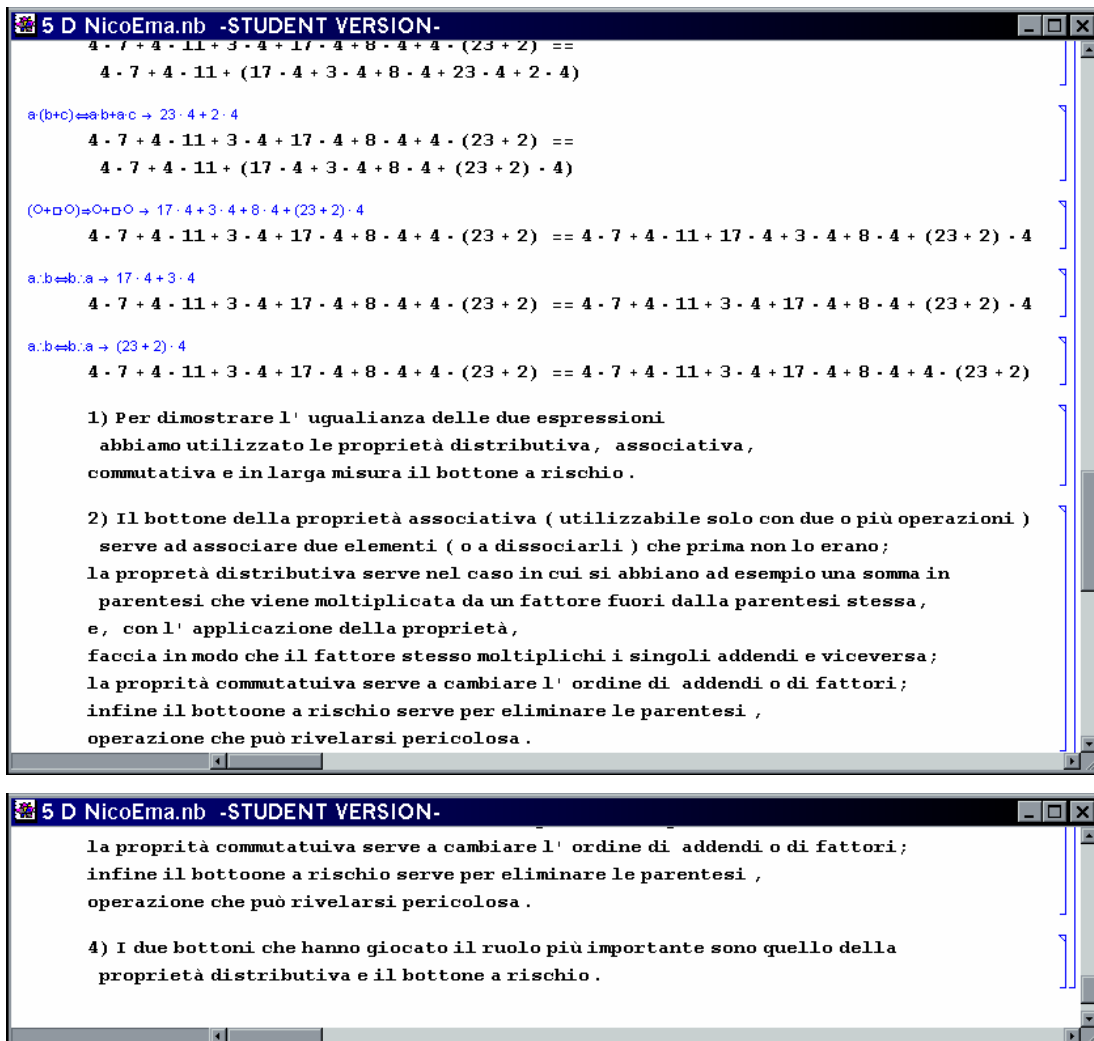
a:b⇒b:a ⇒ 3 · 4 + 17 · 4
4 · 7 + 4 · 11 + 8 · 4 + (17 · 4 + 3 · 4) + 4 · (23 + 2) ==
4 · 7 + 4 · 11 + (17 · 4 + 3 · 4 + 8 · 4 + 23 · 4 + 2 · 4)

a:b⇒b:a ⇒ 8 · 4 + (17 · 4 + 3 · 4)
4 · 7 + 4 · 11 + (3 · 4 + 17 · 4) + 8 · 4 + 4 · (23 + 2) ==
4 · 7 + 4 · 11 + (17 · 4 + 3 · 4 + 8 · 4 + 23 · 4 + 2 · 4)

(O+□O)⇒O+□O ⇒ 3 · 4 + 17 · 4
4 · 7 + 4 · 11 + 3 · 4 + 17 · 4 + 8 · 4 + 4 · (23 + 2) ==
4 · 7 + 4 · 11 + (17 · 4 + 3 · 4 + 8 · 4 + 23 · 4 + 2 · 4)

```

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.



Interpretazione della sperimentazione in termini di object-mediation

Dal procedurale al relazionale

All'inizio della sperimentazione, il test preliminare, ha rivelato che gli studenti conoscevano le proprietà delle operazioni in maniera piuttosto confusa, come abbiamo già mostrato nei paragrafi precedenti. Bisogna però rivelare che in generale ne avevano un'idea di tipo operativo; esse sono interpretate come "*Regole che servono per risolvere le espressioni*" o comunque regole che servono per modificare

le espressioni in maniera unidirezionale²². In effetti nelle loro precedenti esperienze scolastiche, nell'ambito particolare delle espressioni numeriche, gli studenti usavano le proprietà inconsciamente ed in maniera operativa per *calcolare* le espressioni. In particolare, questa era la situazione che ci aspettavamo poiché nelle scuole medie, in generale, l'unico tipo di attività che coinvolge le espressioni consiste nel calcolare le stesse. Per ovviare a questo inconveniente, come prima cosa, abbiamo dato ai ragazzi la possibilità di attribuire un nuovo ruolo alle espressioni, grazie ad attività in cui era richiesto di confrontarle invece che di calcolarle. Dal punto di vista della teoria dell'object-mediation abbiamo fornito agli studenti un nuovo tipo di discorso, riguardante le espressioni, nel quale in particolare è possibile *usare* le proprietà. Il tipo di discorso in questione è caratterizzato dal *template* (schema di discorso): "seleziono l'espressione e clicco il bottone in modo da modificarla per trasformarla in un'espressione equivalente". In questo particolare template il *bottone* rappresenta una proprietà. Di conseguenza, la parola *equivalenti*, che in origine aveva significato di *aventi lo stesso risultato*, acquisisce una nuova interpretazione, legata all'uso, al tipo di compito, ed al feedback che la macchina dà: *equivalenti vuol dire trasformabili l'una nell'altra tramite le proprietà*.

Il tipo di discorso in questione è reso possibile dalla peculiarità del software che rende percettibili, e soprattutto utilizzabili, le proprietà, per scopi diversi dal semplice *calcolo di un'espressione*. Inoltre osserviamo che, il particolare tipo di consegna, porta a superare l'interpretazione, del termine *trasformazione*, come prefissato ad ottenere espressioni di un certo tipo, ovvero in forma canonica. Il fatto che la macchina si carichi dell'esecuzione di passaggi pesanti, dal punto di vista della

²² Consegna T2.

Protocollo 21:

Proprietà associativa: *proprietà che associa tutti i numeri*

Proprietà distributiva: *proprietà che distribuisce tra i numeri di un'espressione*

Proprietà commutativa: *proprietà che cambia i numeri ma non il risultato*

Proprietà: *caratteristica di una formula*

Protocollo 13:

Proprietà distributiva: $2 \cdot (3+4) = 2 \cdot 3 + 2 \cdot 4 = 6 + 8 = 14$

Protocollo 23:

Proprietà distributiva: *scomporre un numero in altri più piccoli*

Protocollo 2:

Proprietà associativa: *serve per semplificare le addizioni* $2+4+3=2+(4+3)$

Proprietà distributiva: *permette di distribuire il fattore sui due addendi* $2 \cdot (4+1) = (2 \cdot 4) + (2 \cdot 1)$

scrittura, permette agli allievi di fare esperienze con scritture diverse. Quando si opera in ambiente carta e penna, invece, non è possibile fare esperimenti. Il feedback della macchina permette un uso controllato del bottone; la dialettica tra previsione e controllo consente l'emergere di un significato come prodotto di adattamenti successivi di interpretazioni.

La prima conseguenza dovuta al particolare tipo di attività svolto utilizzando L'Algebrista è che le proprietà hanno smesso di essere unidirezionali, acquistando lo status di relazioni di equivalenza. A questo proposito riportiamo alcuni protocolli riguardanti le risposte²³ date da alcuni studenti nel test preliminare ed in un compito per casa assegnato dopo qualche settimana di sperimentazione. In entrambe le consegne (T 2 e CS 7) allo studente era richiesto di parlare delle proprietà delle operazioni.

Alessandro:

T 2:

proprietà associativa *antisimmetrica*; proprietà distributiva *difficile*; proprietà commutativa *facile*; proprietà *commutativa*.

CS 7:

Commutativa = cambiando l'ordine degli addendi (o prodotti) il risultato non cambia $[a \setminus b \setminus c] \setminus a$;

Distributiva = una somma di numeri moltiplicata per un altro numero è equivalente alla somma delle moltiplicazioni tra il numero non sommato ed uno dei due addendi $[a \bullet (b+c)] \setminus a \bullet b + a \bullet c$

Associativa = in una moltiplicazione (o addizione) di tre numeri l'ordine delle parentesi non conta $[a \setminus (b \setminus c)] \setminus (a \setminus b) \setminus c$

Serena:

T 2:

la proprietà associativa è quando associo due numeri per facilitarmi il calcolo: $7+3+8+4+2+6 = \{7+3=10; 8+2=10; 6+4=10\} = 30$; La proprietà distributiva si ha quando un numero si distribuisce a tutti i membri di un'operazione; proprietà commutativa: quando si scambiano di posto gli addendi o gli operandi di una addizione o di una moltiplicazione; le proprietà sono come delle regole che vengono applicate alle operazioni.

CS 7:

(riporta prima uno schema che rappresenti la paletta dei bottoni).

Associativa: la proprietà associativa si può applicare solamente alle operazioni che hanno solamente o il segno \bullet o il segno $+$, es.:

²³ Riportiamo in italico tutto ciò che è stato scritto dagli studenti

$$3+5+6 = \text{Associativa } 14$$

$$3+(5+6) = \text{Associativa } 14$$

$$(3+5)+6 = 14$$

$$3 \cdot 5 \cdot 6 \text{ Associativa } =$$

$$(3 \cdot 5) \cdot 6 \text{ Associativa } =$$

$$3 \cdot (5 \cdot 6)$$

Distributiva: la proprietà distributiva deve essere utilizzata in questa situazione $a \cdot (b+c) = a \cdot b + a \cdot c$, anche con più numeri tra parentesi. Nel programma il numero che deve moltiplicare i numeri tra parentesi deve essere posto a destra o a sinistra, cioè o $a \cdot b + a \cdot c$ o $b \cdot a + c \cdot a$

Commutativa: è la proprietà più semplice, si utilizza quando si vogliono scambiare solamente gli ordini degli addendi tra i numeri della moltiplicazione e il risultato non cambia.

Emanuele:

T 2:

proprietà associativa $(3+2) \cdot 6 = 5 \cdot 6$; proprietà distributiva $13+15=13+10+15+10$;
proprietà commutativa $6+5+4=5+6+4$; proprietà commutativa (addizione).

CS 7:

Associativa

- il simbolo \backslash riassume il simbolo "+" e "•"
- servono almeno due operazioni per applicarla
- una volta applicata le parentesi attorno ai numeri non vengono spostate bensì aggiunte

Distributiva

- usabile solo con moltiplicazione ed addizione (che poi sono le uniche operazioni svolte dal programma).
- Questo bottone applica la proprietà in entrambe i modi $[a \cdot (b+c); a \cdot b + a \cdot c]$

Commutativa

- Bisogna fare attenzione e non evidenziare parentesi quando si applica

Lucia:

T 2:

proprietà associativa è una proprietà che dice che si possono cambiare ad esempio le parentesi in un'espressione. Es.: $17+(8+9)=(17+8)+9$; La proprietà distributiva ci dice che i numeri possono essere "scomposti" in un'espressione, ma il risultato non cambia. $(5+10) \cdot 3 = (5+5 \cdot 2) \cdot 3$; proprietà commutativa è una proprietà che hanno solo le addizioni e le sottrazioni e dice che cambiando l'ordine degli addendi (o dei moltiplicandi) il risultato non cambia; proprietà "ho usato la proprietà commutativa".

CS 7:

(riporta prima uno schema che rappresenti la paletta dei bottoni).

Associativa: questa proprietà si può applicare sia se troviamo un'addizione, sia se troviamo un prodotto, a patto però che i due segni (fuori e dentro le parentesi) siano uguali. Applicando questa proprietà, noi possiamo trasformare delle

operazioni così: $4+(5+3) \sqcap (4+5)+3$ oppure $4\bullet(5\bullet3) \sqcap (4\bullet5)\bullet3$ (le freccette per indicare che funziona anche all'inverso).

Distributiva: questa proprietà si può applicare soltanto se abbiamo un prodotto tra uno o più numeri per un'addizione (messa tra parentesi) di due o più numeri. Es.: $5\bullet(4+3)=5\bullet4+5\bullet3$. Però funziona anche con tre numeri es.: $5\bullet(4+3+2)=5\bullet4+5\bullet3+5\bullet2$. Anche questa proprietà funziona anche all'inverso. Praticamente essa distribuisce il numero moltiplicatore e lo fa moltiplicare per ognuno degli addendi fra parentesi, e viceversa.

Commutativa: questa proprietà funziona sia con un'addizione che con una moltiplicazione. Essa, praticamente scambia l'ordine degli addendi o dei moltiplicandi. Es.: $3\bullet4=4\bullet3$ oppure $3+4=4+3$.

Questi protocolli illustrano come ci sia stata un'evoluzione verso un'interpretazione dichiaratamente bidirezionale delle proprietà, che in principio sembravano avere un solo verso ben determinato. In effetti possiamo notare che in tutti i casi analizzati gli studenti specificano che le proprietà funzionano sia in un verso che nell'opposto. Tale cambiamento è piuttosto evidente per quanto riguarda la proprietà distributiva e quella associativa. Bisogna però notare che anche la proprietà commutativa, almeno in alcuni casi, sembra aver subito un cambiamento netto, ad esempio, Lucia, nel test preliminare scrive "*cambiando l'ordine degli addendi ... il risultato non cambia*", questa definizione di proprietà commutativa è chiaramente collocabile in un contesto di utilizzo della stessa in termini di *calcolo* di un'espressione per ottenere un risultato. Nella consegna CS 7, invece, Lucia scrive "*essa praticamente scambia l'ordine degli addendi o dei moltiplicandi. Es.: $3\bullet4=4\bullet3$ oppure $3+4=4+3$* "; la proprietà non è più espressa facendo riferimenti al risultato, e l'esempio riportato mostra che Lucia pensa in termini di equivalenza di espressioni, piuttosto che di uguaglianza del risultato. Una situazione analoga è riscontrabile nel protocollo di Serena per quanto riguarda la proprietà associativa. Anche lei nella risposta al test preliminare (T 2) dà una versione operativa della suddetta proprietà, in termini di ottimizzazione: "*...è quando associo due numeri per facilitarmi il calcolo*", dopo di che riporta un esempio di calcolo di un'espressione. Per contro, nella consegna CS 7, Serena, riporta due esempi in cui mostra due percorsi di espressioni equivalenti, tali che si possa passare dall'una all'altra tramite la proprietà associativa; in uno dei due percorsi proposti, in particolare, affianca alle espressioni

il risultato di ciascuna, come se volesse ribadire l'equivalenza delle stesse²⁴. In definitiva studenti sembrano essere passati da un atteggiamento prettamente procedurale ad uno tendenzialmente relazionale. A nostro avviso i motivi di questa evoluzione sono da ricercare nella particolarità delle attività svolte e nella mediazione offerta dal software. Il ruolo giocato da L'Algebrista risulta evidente in quanto gli studenti, parlando delle proprietà, utilizzano verbi come "funziona" (vedi Lucia) oppure fanno riferimenti espliciti al programma (vedi Emanuele e Serena), ma un ruolo altrettanto decisivo lo hanno le attività svolte, in particolare l'attività di confronto tra espressioni. Facendo riferimento ai bottoni, gli studenti utilizzano le proprietà, entità matematiche, in discorsi di tipo AR. Una delle nostre ipotesi iniziali era proprio che, grazie all'uso del prototipo, gli studenti potessero oggettualizzare le proprietà delle operazioni. I protocolli che abbiamo analizzato mostrano come questo in effetti sia accaduto: nel test preliminare le proprietà erano state descritte come qualcosa che permette allo studente di agire su un'espressione; nella consegna CS 7 invece, sono le proprietà stesse che agiscono, una proprietà trasforma l'espressione al posto dello studente. Questo cambiamento nel ruolo giocato dalle proprietà è particolarmente evidente nei protocolli di Lucia, Serena (vedi sopra) e di Federico; nel caso di quest'ultimo il cambiamento si può osservare, seguendo l'evoluzione dell'interpretazione data da Federico, prima, una proprietà "consiste nel ...", mentre dopo "semplifica ...", "... mette per esteso" e "..commuta...":

Federico:

T 2:

La proprietà associativa è quella proprietà che consiste nell'associare; proprietà distributiva consiste nel distribuire ai termini di una operazione un altro numero; proprietà commutativa consiste nello scambiare i termini di un'addizione o di una moltiplicazione; le proprietà delle potenze.

CS 7:

Distributiva: la proprietà distributiva, semplifica più moltiplicazioni se queste hanno il numero che moltiplica uguale o viceversa le mette per esteso.

Commutativa: questa proprietà commuta o due numeri legati da un'addizione, o due numeri legati da una moltiplicazione.

²⁴ Ricordiamo che la relazione di equivalenza, tra espressioni, da cui siamo partiti, era basata sull'equivalenza dei risultati.

Il fatto che la proprietà agisca sull'espressione, invece dello studente, può essere giustificato dal fatto che nelle attività che coinvolgono L'Algebrista, in effetti, è il *bottone della proprietà* che trasforma l'espressione, su ordine dell'utente. Un atteggiamento di questo tipo, da parte dello studente, può essere interpretato positivamente poiché sembra essere un primo passo verso l'acquisizione della capacità di distaccarsi dal proprio lavoro in maniera da poterlo giudicare, e far diventare oggetto di riflessione l'attività stessa. In questo modo lo studente si trova ad avere anche uno strumento per il controllo, della correttezza del proprio lavoro, basato sull'uso delle proprietà delle operazioni.

Il selezionatore

Un ruolo fondamentale è stato svolto dal "selezionatore", ovvero quella peculiarità del software per cui ogni volta che vogliamo agire su una sotto espressione dobbiamo selezionarla, ed in particolare è possibile selezionare solo stringhe che siano effettivamente sotto espressioni matematiche. Esso ha avuto conseguenze rilevanti sugli studenti che sin dall'inizio se ne sono appropriati riportandolo anche in ambiente carta e penna; ogni volta che uno studente deve trasformare un'espressione, agendo su una sotto espressione, evidenzia, con qualche accorgimento grafico²⁵, la sotto espressione in questione e scrive accanto quale proprietà, o teorema, applica. Questo a nostro avviso è indice di un approccio strutturale alle espressioni, in quanto quando lo studente "seleziona" una sotto espressione, in realtà evidenzia la struttura della stessa; ed il fatto stesso che la struttura viene evidenziata è indice che lo studente la ritiene in qualche modo importante. Come esempio Riportiamo una parte del protocollo prodotto da Eleonora per la consegna CS 2, in cui è richiesto di dimostrare l'equivalenza tra due espressioni, in ambiente carta e penna:

$$7+(-2)+\boxed{6\cdot(3+4)}+5\cdot6 = (3+4+5)\cdot6+7+(-2)$$

sulla parte evidenziata applico la proprietà distributiva

$$7+(-2)+\boxed{6\cdot3}+(6\cdot4)+5\cdot6 = (3+4+5)\cdot6+7+(-2)$$

applico sulla parte evidenziata la commutativa

²⁵ In genere sono usati evidenziatori, parentesi quadre o sottolineature.

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

$$7+(-2)+(6\cdot 3)+(6\cdot 4)+5\cdot 6 = (3+4+5)\cdot 6+7+(-2)$$

applico la commutativa

[...]

Inoltre la distinzione tra espressione e sotto espressione, su cui si agisce, dirige l'attenzione su aspetti strutturali interrompendo il flusso del calcolo, che vede ogni espressione come prodotto della precedente e come passo intermedio verso il risultato finale.

In conclusione gli studenti si appropriano dell'abitudine, di L'Algebrista, di individuare una trasformazione tramite la proprietà utilizzata e tramite la sotto espressione su cui questa viene applicata. Un'espressione assume lo status di oggetto che ha una sua struttura caratteristica, e che può essere equivalente ad altri oggetti simili; essa non è più, *solo*, un procedimento di calcolo, ma un elemento unitario sul quale è possibile agire in conformità a regole.

La questione linguistica

La mediazione offerta dal software va al di là dell'oggettualizzazione delle proprietà, delle espressioni e delle trasformazioni, e riguarda più in generale, l'evoluzione del linguaggio matematico e le capacità espressive degli studenti. Le scritture rappresentanti un bottone introducono un simbolo per lo stesso oggetto adatto sia al *discorso matematico scritto* (" $a\cdot b$ \downarrow $b\cdot a$ ") ed uno adatto al *discorso matematico orale* ("proprietà commutativa"). Quello relativo al discorso scritto, in particolare, è nuovo e crea spazio²⁶ per tipi di discorso nuovi, come l'uso della simbologia matematica e l'uso delle lettere. La nostra ipotesi è che grazie a questo nuovo spazio creato i ragazzi dovrebbero poter essere in grado di muoversi dentro, ed in particolare ad esempio di trovare scritture simboliche per rappresentare nuovi bottoni. La nostra sperimentazione si è fermata prima di poter analizzare a fondo attività in cui era coinvolta la creazione di nuovi bottoni (teoremi), ma abbiamo comunque potuto notare come gli studenti si siano appropriati di certe simbologie caratteristiche del software. Ad esempio i simboli " \square " e " \square ", che in alcuni bottoni del software rappresentano espressioni generiche, termini qualsiasi, sono stati acquisiti

ed autonomamente utilizzati dai ragazzi, aprendo la strada ad un principio di sostituzione non ancora esplicitato. I seguenti esempi mostrano come gli studenti si siano appropriati dell'uso di tali simboli; nella consegna CS 12, alla richiesta di congetturare una formula $(a+b)^4$, nello stile dei prodotti notevoli, Annalisa scrive " $(\square+\square)^4=(\square+\square)(\square+\square)(\square+\square)(\square+\square)$ "; mentre nella consegna CL 20, alla richiesta di esprimere con una formula il fatto che un numero qualsiasi sia divisibile per 31, Francesco scrive " $31 \mid x$; dove x è il numero inserito".

Un esempio di appropriazione degli strumenti algebrici

Nella prima parte della sperimentazione abbiamo lavorato solo ed esclusivamente con espressioni numeriche, in quanto il nostro obiettivo era che gli studenti si familiarizzassero con le trasformazioni di espressioni mediante le proprietà, in maniera da preparare un solido terreno per affrontare le espressioni letterali. Il passaggio alla seconda fase è avvenuto in maniera brusca ed improvvisa, ma sorprendentemente è stato assorbito dagli studenti in maniera più che positiva. L'attività che segna l'inizio di questa fase è la CL 9: *ai ragazzi sono state presentate tre diverse espressioni letterali come rappresentanti dell'area di una certa figura geometrica, la consegna era individuare quali di queste formule fossero corrette*.

Iniziamo l'analisi dell'attività osservando che i ragazzi si trovavano di fronte a tre diversi tipi di discorsi ai quali riferirsi:

1. Il discorso geometrico, in tal caso provare la correttezza dell'espressione significa mostrare come ognuna delle espressioni possa rappresentare il calcolo dell'area in questione;
2. il discorso relativo al *calcolo* dell'espressione, secondo cui *due espressioni numeriche sono equivalenti se danno lo stesso risultato*. In questo caso la fase *template driven* si può individuare quando lo studente sostituisce "numeriche" con "letterali" ottenendo la frase "*due espressioni letterali sono equivalenti se danno lo stesso risultato*"²⁷. Secondo la terminologia usata in classe, in questo

²⁶ spazio inteso come ambiente d'uso e di comunicazione che per sua natura è aperto e sviluppabile

²⁷ Nel caso di espressioni letterali la parola risultato ha senso solo se si sostituiscono dei numeri alle lettere, a meno che con "risultato" non si intenda l'espressione in forma canonica. La questione è stata

caso si parlerebbe di *verifica*. In questo *template* includiamo anche l'idea di sostituire dei numeri alle lettere e calcolare l'espressione ottenuta.

3. L'*dimostrazione* tramite l'uso delle proprietà delle espressioni, in questo caso la fase *template driven* è individuata dal passaggio da "*due espressioni numeriche sono equivalenti se è possibile trasformare l'una nell'altra tramite le proprietà*" in "*due espressioni letterali sono equivalenti se è possibile trasformare l'una nell'altra tramite le proprietà*".

La consegna non dava ai ragazzi alcuna indicazione sul tipo di approccio a cui rifarsi, c'era da aspettarsi che le strategie degli alunni si diversificassero molto, in effetti la situazione è risultata piuttosto omogenea. Osserviamo che nonostante il problema fosse nuovo, e nonostante fosse la prima volta che lavoravano con le lettere, i ragazzi non hanno avuto alcuna difficoltà iniziale nel capire il problema e farlo proprio; questo può essere riletto in termini di: "ho dei *template* di riferimento, dei vecchi tipi di discorso da riutilizzare, quindi parto riferendomi a loro" (es. formule per l'area). Inoltre osserviamo che a questo punto i ragazzi avevano a loro disposizione anche il *template* (tipo di discorso derivante da precedenti attività con il software) "*due espressioni numeriche possono anche essere equivalenti, non necessariamente sono oggetti totalmente distinti*", dove "*numeriche*" può essere sostituito con "*letterali*".

Alcuni hanno avuto un approccio del tipo 1, cioè hanno cercato di giustificare geometricamente la correttezza di ciascuna espressione interpretandola, e giustificandola, come procedimento di calcolo dell'area.

Altri hanno *calcolato* (2) le espressioni riducendole tutte in forma canonica per confrontarle (osserviamo, però, che non si era mai parlato di questa particolare forma, e, per quanto ne sappiamo, gli studenti non conoscevano il termine "*forma canonica*").

affrontata più volte con i ragazzi durante la sperimentazione, ma i problemi didattici che porta con sé sono troppo complessi per essere affrontati in questa sede.

Altri infine hanno *dimostrato* (3) che le espressioni erano equivalenti trasformando le une nelle altre usando le proprietà; alcuni lo hanno fatto con il computer, altri in ambiente carta e penna.

La maggior parte di loro, dopo la fase di approccio, si sono rifatti anche ad altri tipi di discorsi (sempre a scelta tra 1, 2, 3) per confermare le loro risposte. In particolare, come discorso finale tutti hanno scelto il 3, come per validare la loro tesi *dimostrandola*; inoltre la risposta conclusiva, e definitiva, la hanno data tutti avvalendosi del computer che ha assunto, per loro esplicita affermazione, il ruolo di strumento di controllo che assicura di non fare errori.

In definitiva possiamo notare che i ragazzi sono *spontaneamente* passati attraverso diversi tipi di discorso, riportando ed analizzando l'equivalenza tra le tre espressioni in ognuno di questi ambienti. Questo secondo Sfard (Sfard 1997) dovrebbe essere il motore per la mediazione semantica ed in particolare per la *nascita* di oggetti matematici.

Analizziamo un protocollo in particolare.

Lucia ed Sandra, per prima cosa, hanno prodotto una propria formula, diversa dalle altre proposte dal compito, che rappresentasse l'area in questione e della cui certezza erano sicure. Dopo di che hanno *dimostrato*, con carta e penna, che ognuna delle altre formule era "*giusta*" trasformandole tutte in quella che avevano prodotto loro. Direi che in questo caso la fase *template driven* è stata molto breve, poiché il fatto stesso di aver introdotto spontaneamente un altro simbolo (l'espressione da loro prodotta) che rappresentasse l'oggetto "espressioni equivalenti", è già tipico della fase *object mediated*. Probabilmente sull'istantaneità di questo passaggio influisce il fatto che grazie alle attività precedenti le espressioni avevano già uno status di oggetti dei quali è possibile discutere l'equivalenza. In effetti è possibile osservare, in questo caso, la presenza di un isomorfismo tra discorsi di tipo diverso, quello geometrico e quello algebrico; Lucia e Sandra si muovono con disinvoltura tra i due ambiti. Per concludere, Lucia ed Sandra hanno ridimostrato l'equivalenza delle espressioni avvalendosi di L'Algebrista.

Il software, e gli strumenti algebrici che gli studenti avevano acquisito nei due mesi precedenti, hanno assunto un ruolo fondamentale in questa attività, in quanto i

ragazzi se ne sono serviti, spontaneamente, per risolvere un problema che apparteneva all'ambito geometrico. Questo a parer nostro è un risultato notevole.

Produzione intradiscorsiva di nuovi significati

Riguardo la produzione intradiscorsiva di nuovi significati e/o simboli matematici, citiamo l'esempio della creazione di nuovi bottoni come prodotto dell'esigenza di accorciare i tempi di esecuzione di procedure ormai familiari ai ragazzi. Consideriamo, ad esempio, il bottone per cancellare due termini opposti. La possibilità di cancellare termini opposti era nota agli allievi, ed è stata dimostrata dagli studenti per alcuni casi particolari; in questo modo essi hanno ripetuto più volte una procedura imparando a riprodurla quando vogliono. A questo punto è stato introdotto il bottone " $\square-\square\square\square$ "; con l'introduzione dell'oggetto/simbolo bottone " $\square-\square\square\square$ ", questa procedura assume uno status di oggetto ed entra a far parte del discorso dell'interazione con il micromondo alla pari degli altri bottoni. In sintesi, tramite discorsi interni al nostro micromondo gli studenti sono stati in grado di creare un nuovo oggetto allargando quindi le possibilità discorsive del micromondo stesso. Un aspetto fondamentale per l'introduzione del nuovo oggetto è stata l'idea di "economia di pensiero", legata al riutilizzare di equivalenze già dimostrate. Questo aspetto rappresenta un elemento importante nella introduzione degli allievi al pensiero teorico; infatti, gli allievi hanno sperimentato l'uso ripetuto di una catena deduttiva fino a rendersi conto della generalità della sua applicazione, e solo a questo punto, di fronte alla necessità di risparmiare tempo, è stato introdotto il teorema. Si ha qui l'occasione di fare esperienza con un aspetto tanto importante quanto delicato del dimostrare in algebra; distinguere tra teoremi generalmente interni (equivalenze deducibili dalle proprietà) e teoremi come nuovi schemi di calcolo, significativi nel senso che assumono senso e valore in termini dei problemi che risolvono, in particolare problemi di economia di calcolo.

A questo proposito rimandiamo ad una sperimentazione futura lo studio più approfondito dell'uso del Teorematore.

1 Confronto con altri software

Lo scopo di questo capitolo è il confronto de' L'Algebrista con altri software già esistenti. Il confronto avverrà sulla base di una serie di criteri tratti dalla letteratura esistente in merito e su altri da noi stabiliti; in questo modo cercheremo di evidenziare alcune differenze, tra i software in questione, che riteniamo particolarmente rilevanti da un punto di vista didattico.

Procederemo descrivendo gli obiettivi didattici de' L'Algebrista e le loro prime conseguenze che di per se individuano a grandi linee il modello. Il passo successivo sarà l'introduzione di una serie di criteri che ci permetteranno infine di classificare L'Algebrista e paragonarlo con altri software. Infine definiremo il modello relativo al nostro software e lo confronteremo con gli altri.

Gli obiettivi de' L'Algebrista: le prime conseguenze sul modello

Esaminando un certo numero di software piuttosto popolari, in ambito scolastico, ci siamo subito resi conto che la prima forte distinzione tra L'Algebrista e gli altri ha origini nelle motivazioni che ci hanno spinto alla sua progettazione e realizzazione. Esso, infatti, è stato concepito con l'idea di perseguire due obiettivi didattici ben determinati:

- *introduzione degli studenti alla manipolazione simbolica*, un passaggio che riteniamo fondamentale in quanto fornisce agli studenti alcuni degli strumenti necessari per l'apprendimento dell'algebra;
- *introduzione degli studenti all'idea di teoria e di dimostrazione*, due concetti, questi, che stanno alla base di tutta la matematica e che quindi riteniamo fondamentali.

Questi obiettivi, come vedremo, hanno influito decisamente sulla realizzazione del prototipo distinguendolo in maniera piuttosto netta dagli altri software che abbiamo esaminato.

Riteniamo quindi che il primo criterio di confronto, tra più prodotti, debba basarsi sull'analisi delle motivazioni e degli obiettivi, che stanno alla base della realizzazione di ciascuno di essi.

La nostra intenzione, quindi, era di realizzare un software finalizzato all'apprendimento della manipolazione simbolica, e che offrisse anche un modello dell'algebra delle espressioni algebriche. Per fare questo, appoggiandoci sulle numerose ricerche fatte in proposito, abbiamo scelto di basarci sull'idea di micromondo, con lo scopo di creare un ambiente dove lo studente potesse fare esperienze significative sulla parte di matematica che c'interessava. In particolare ci siamo riferiti ad esperienze d'utilizzo di Cabrì, che condividevano con noi l'obiettivo riguardante l'idea di teoria e di dimostrazione, e che riguardavano l'introduzione degli studenti all'idea di costruzione geometrica (Mariotti 1996, 1998, 1998).

Un micromondo è caratterizzato da oggetti computazionali in grado di interagire tra loro, e soprattutto da modelli di domini di conoscenza che possono essere investigati tramite l'interazione con il software (Noss & Hoyles²⁸ 1996, p.g. 65). Sulla base di questa definizione abbiamo progettato e realizzato L'Algebrista. A tal fine abbiamo individuato gli assiomi, le definizioni, ed i teoremi elementari che stanno alla base della teoria delle espressioni algebriche, e che d'ora in avanti riuniremo sotto il nome di *assiomi* in quanto li assumeremo tali; quindi abbiamo scelto di realizzare il software fornendolo essenzialmente di comandi di tipo elementare, in modo da avere una corrispondenza biunivoca tra comandi ed assiomi: ogni comando è un oggetto computazionale che rappresenta un assioma. Questa scelta di base distingue L'Algebrista già in origine dai manipolatori simbolici classici in quanto questi sono, in generale, progettati per essere *strumenti* per la manipolazione simbolica, e, in quanto tali, offrono all'utente solamente comandi di tipo complesso, e generalmente senza riferimenti chiari agli assiomi dell'algebra.

²⁸ “ ... at the core of a microworld there is therefore a model of a knowledge domain to be investigated by interaction to promote learning.”

Un'ulteriore implicazione, che segue direttamente dagli obiettivi didattici che abbiamo individuato, è che come micromondo algebrico, L'Algebrista, debba essere anche sviluppabile, in quanto una caratteristica fondamentale della matematica è la dinamicità delle sue teorie le quali crescono continuamente. Esso deve offrire allo studente la possibilità di inserire, in maniera semplice, nuovi comandi, nuove regole di trasformazione delle espressioni, o equivalentemente, lo studente, deve avere la possibilità di cambiare l'insieme dei comandi, in quanto questo corrisponde a cambiare gli assiomi. A questo proposito possiamo riferirci al *pump-mecanism* descritto da Sfard (1997) secondo cui dalle relazioni tra entità intradiscorsive matematiche (ad esempio assiomi, od oggetti particolari che rispecchiano definizioni specifiche), possono nascere nuove entità matematiche che a loro volta diventano parte del discorso matematico contribuendo alla generazione di nuove entità.

Per descrivere in maniera esauriente e comprensibile il modello de' L'Algebrista, di cui ora abbiamo dato un'idea, riteniamo necessario introdurre una serie di definizioni che ci aiutino ad individuarne le caratteristiche fondamentali e a distinguerlo dagli altri software. A tale fine definiremo varie tipologie di ambiente e classificheremo i comandi da esse offerti.

Tipologie di ambiente

L'Algebrista è stato progettato e realizzato come un ambiente particolare in grado di interagire con altri tipi di ambienti in modo da offrire:

- *la possibilità di editare espressioni algebriche e di inserire commenti in ogni punto del foglio di lavoro ed in ogni momento;*
- *la possibilità di trasformare espressioni coerentemente con gli assiomi dell'algebra tramite comandi complessi;*
- *la possibilità di trasformare espressioni coerentemente con gli assiomi dell'algebra tramite comandi elementari;*
- *la possibilità di interagire con altri micromondi matematici.*

Queste caratteristiche potrebbero essere tutte condensate in unico software, ed in effetti, alcune di esse, nel nostro caso lo sono, ma noi riteniamo opportuno distinguerle in quanto individuano quattro tipologie di ambienti che ci permetteranno di delineare il modello di riferimento de' L'Algebrista e di stabilire alcuni criteri di confronto con gli altri software. Al fine di classificare e distinguere i tipi di ambienti individuati dalle proprietà sopra elencate, dobbiamo introdurre alcune definizioni per distinguere le possibilità offerte da ciascuno di essi:

Editare e trasformare: è bene distinguere cosa intendiamo, con i termini editare e trasformare quando parliamo di espressioni algebriche. *Entrambe le parole stanno ad indicare dei cambiamenti che si effettuano su un'espressione, ciò che li distingue è il modo in cui i cambiamenti sono effettuati. Usiamo la terminologia "editare un'espressione" quando ci riferiamo a cambiamenti effettuati "a mano libera", senza utilizzare comandi che contengano parti della teoria in cui stiamo lavorando. Si dice **trasformazione** un cambiamento effettuato rispetto a proprietà prestabilite, ad esempio tramite un bottone che rappresenti un assioma.*

Quando si edita un'espressione è possibile generare degli "errori", ovvero delle espressioni non equivalenti a quella di partenza, quando si effettuano delle trasformazioni invece no. Ad esempio se considero l'espressione $\mathbf{a+b\bullet c}$, tramite un editore di testo (anche con carta e penna) posso editarla e farla diventare $\mathbf{b+a\bullet c}$ semplicemente cancellando alcuni caratteri e digitandone altri, senza fare, quindi alcun riferimento agli assiomi dell'algebra. Invece, ne' L'Algebrista, possiamo selezionare tutta l'espressione, cliccare sul bottone che rappresenta la proprietà commutativa, e in questo modo possiamo trasformare l'espressione iniziale in $\mathbf{a\bullet c+b}$; inoltre non possiamo effettuare edizioni come quella dell'esempio precedente in quanto il software non ci permette selezioni del tipo $\mathbf{a+b\bullet c}$. Osserviamo che anche ne' L'Algebrista è possibile fare particolari edizioni non necessariamente conformi alle proprietà, ad esempio utilizzando il *bottone a rischio* (che toglie le parentesi); ma il nome mette in guardia sul fatto che esso non effettua alcun controllo sulla correttezza dell'operazione (quindi può generare edizioni del tipo: $(\mathbf{a+b})\bullet \mathbf{c} \rightarrow \mathbf{a+b\bullet c}$, non accettabile da un punto di vista matematico).

Infine rileviamo che il risultato di una trasformazione è un'espressione che è comunque ottenibile tramite edizioni, quindi abbiamo la seguente catena di inclusione tra insiemi:

$$\text{effetto di trasformazioni} \sqsubset \text{effetto di edizioni}.$$

Trasformazioni elementari e trasformazioni complesse: una trasformazione si dice **complessa** quando è non ottenibile tramite meno di due trasformazioni elementari. Vice versa una trasformazione si dice **elementare** quando è minimale, ovvero, è possibile effettuarla, o spiegarla, o riprodurla, tramite l'uso di un assioma, o di un teorema, o di una definizione, con un unico passaggio.

Un esempio di trasformazione complessa è il seguente passaggio: $a+3+b \cdot b+2 \rightarrow b^2+a+5$; esso può essere ottenuto tramite quattro trasformazioni elementari come mostrano i seguenti passaggi: $a+3+b \cdot b+2 \rightarrow a+b \cdot b+3+2 \rightarrow a+b^2+3+3 \rightarrow a+b^2+5 \rightarrow b^2+a+5$; nel primo è applicata la proprietà commutativa, alla sotto espressione $3+b$; nel secondo è stata applicata la definizione di potenza a $b \cdot b$; nel terzo passaggio si utilizza la definizione di somma tra numeri interi; infine nell'ultimo passaggio si riutilizza la proprietà commutativa. Ciascuno dei suddetti passaggi corrisponde ad una trasformazione elementare.

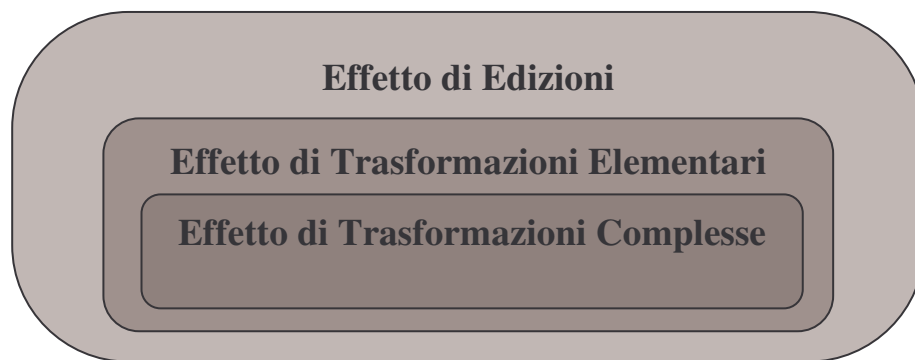
Se chiamiamo **grado di complessità** di una trasformazione, il numero minimo di trasformazioni elementari necessari per effettuarla, allora possiamo dire che una trasformazione si dice **complessa** se il suo grado di complessità è almeno 2.

Osserviamo che la complessità di una trasformazione dipende dall'insieme di assiomi, teoremi e definizioni, che si hanno a disposizione, ad esempio la trasformazione $(a+b)^2 \rightarrow a^2+2ab+b^2$ è sicuramente complessa se si hanno a disposizione i soli assiomi di base della teoria delle espressioni algebriche, ma una volta dimostrato che le due espressioni sono equivalenti, ovvero che è possibile trasformare l'una nell'altra tramite trasformazioni elementari, allora è possibile introdurre il teorema $(a+b)^2 \vdash a^2+2ab+b^2$ grazie al quale possiamo passare da un'espressione all'altra tramite una trasformazione che risulta elementare in quanto consiste in un unico passaggio in cui viene applicato il teorema.

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

Osserviamo infine che esiste una gerarchia anche tra trasformazioni complesse ed elementari, ovvero l'insieme delle trasformazioni ottenibile tramite trasformazioni elementari contiene l'insieme delle trasformazioni ottenibili tramite quelle complesse, quindi abbiamo la seguente catena di inclusioni:

effetto di trasformazioni complesse \sqsupset effetto di trasformazioni elementari \sqsupset effetto di edizioni.



Gerarchia indotta dalle trasformazioni

Le definizioni che abbiamo appena dato ci permettono di individuare varie tipologie di ambienti in funzione delle quali classificheremo i software esaminati:

Editore di Testo: *un editore di testo è un qualsiasi ambiente che permette di scrivere e di editare, in qualsiasi momento, ciò che si è scritto. È caratterizzato sostanzialmente da:*

- I. è un ambiente in cui è possibile scrivere qualsiasi tipo di testo;*
- II. è un ambiente in cui è possibile editare in ogni momento ogni tipo di testo;*
- III. non si appoggia su alcuna teoria matematica;*
- IV. totale libertà da parte dell'utente nell'editare il testo, quindi le espressioni.*

Un editore di testo permette di cambiare aspetto ad un'espressione, ma, in quanto non si appoggia su nessuna teoria, non offre nessun controllo sulla correttezza di un'edizione (trasformazione), proprio perché nell'ambito di un editore di testo ogni cambiamento d'aspetto di un'espressione è lecito.

Manipolatore Simbolico (MS): con il termine **manipolatore simbolico** di solito si indica, e noi faremo altrettanto, un ambiente che mette a disposizione dell'utente una serie di comandi piuttosto potenti, ma complessi, utili per effettuare trasformazioni di tipo standard di espressioni algebriche.

Comandi tipici dei manipolatori simbolici sono quelli del tipo "Fattorizza" o "Espandi", il primo fattorizza un'espressione, mentre il secondo la porta in forma canonica; entrambi condensano in sé procedure molto complesse. In effetti, un manipolatore simbolico è caratterizzato essenzialmente da:

- I. *è un ambiente che consente di manipolare espressioni algebriche tramite un insieme di comandi predefiniti;*
- II. *i comandi sono tendenzialmente di tipo complesso e standard, ovvero condensano procedure complesse per ottenere risultati di tipo standardizzato nella pratica matematica;*
- III. *non fornisce alcuna traccia che illustri le procedure seguite nell'esecuzione di comandi di tipo complesso;*
- IV. *decide autonomamente ed automaticamente come ordinare un'espressione, o in generale in che forma presentarla;*
- V. *impossibilità per l'utente di eseguire trasformazioni elementari, in particolare è impossibile ricostruire, tramite passi elementari, le procedure seguite dai comandi complessi del manipolatore;*
- VI. *la libertà ed il potere dell'utente sono ridotti al minimo in quanto è impossibile eseguire operazioni elementari.*

Ambiente Algebrico Elementare (AAE): un **AAE** è un ambiente che consente di manipolare espressioni usando comandi elementari, esso è caratterizzato da:

- I. *è un ambiente che consente di manipolare espressioni algebriche tramite un insieme di comandi predefiniti;*
- II. *I comandi a disposizione (per lo meno quelli di base) sono di tipo elementare;*
- III. *l'insieme dei comandi deve essere coerente in maniera da costituire l'insieme di assiomi di una teoria;*

- IV. *non esegue automaticamente trasformazioni implicite, ad esempio, a differenza dei manipolatori simbolici, non riordina spontaneamente le espressioni;*
- V. *non prende decisioni (vedi il punto precedente), in particolare non sceglie autonomamente in quale forma rappresentare un'espressione a seconda della situazione;*
- VI. *la libertà ed il potere dell'utente sono massimizzati, nei limiti concessi dagli assiomi dell'algebra, in quanto il software non prende decisioni autonomamente ed in quanto sono possibili trasformazioni elementari.*

Ambiente Globale Multifunzionale (o Micromondo di Micromondi o Macromondo): è un ambiente capace di contenere più ambienti e/o micromondi, in modo che questi possano interagire tra di loro.

Esempi di Ambienti Globali Multifunzionali (AGM) possono essere ARI-LAB (Chiappini & Bottino, 1996), *Mathematica*, *Maple* e la T-92. Un AGM può essere ulteriormente sviluppato tramite l'aggiunta o la modifica di micromondi al suo interno.

Ambiente sviluppabile: un micromondo matematico si dice sviluppabile se possiede strumenti che permettono all'utente di definire nuovi oggetti computazionali matematici utilizzabili per interazioni future, o per sostituire vecchi comandi. In particolare un Manipolatore Simbolico (o un AAE) si dice sviluppabile se possiede strumenti che permettano all'utente di definire nuove regole di trasformazione. Questi strumenti devono essere semplici da utilizzare e non devono presupporre conoscenze di programmazione da parte dell'utente, anzi deve essere possibile definire una nuova regola di trasformazione facendo uso del solo linguaggio matematico (o pseudo matematico). In caso contrario il software non può essere definito sviluppabile.

Un esempio di ambiente sviluppabile è L'Algebrista in quanto il *Teorematore* permette appunto di creare bottoni, che rappresentino nuove regole di trasformazione, utilizzando semplicemente il linguaggio simbolico matematico.

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

Mentre un esempio classico di micromondo sviluppabile è Cabrì in quanto offre all'utente la possibilità di creare delle nuove macro sulla base delle primitive (comandi di base) a disposizione.

Osserviamo che in un ambiente sviluppabile è possibile definire una teoria partendo da zero, ovvero dando gli assiomi che si desiderano uno ad uno.

Considerazioni didattiche sulle tipologie di ambiente

I manipolatori simbolici sono piuttosto popolari in ambito scolastico, in quanto risultano molto utili per eseguire trasformazioni di tipo predefinito; d'altronde essi limitano molto la libertà dell'utente poiché l'insieme delle trasformazioni che è possibile eseguire è piuttosto ristretto, e, soprattutto, non offrono nessun collegamento esplicito con gli assiomi dell'algebra. Riteniamo perciò che ambienti di questo tipo siano adatti soprattutto per utenti esperti, ovvero che già abbiano alcune basi consolidate di manipolazione simbolica.

Un AAE invece non offre necessariamente strumenti di trasformazione potentissimi, perciò può rendere molto laboriose e complicate certe consegne, d'altro canto esso è, paradossalmente, enormemente più potente di un manipolatore simbolico in quanto con esso è possibile, teoricamente, effettuare *qualsiasi* trasformazione. Lo riteniamo quindi più adatto, di un manipolatore simbolico, per introdurre gli studenti alla manipolazione simbolica poiché offre vaste possibilità di esplorazione ed offre la possibilità di costruire strumenti complessi, analoghi a quelli rappresentati dai comandi (complessi) dei manipolatori simbolici, come ad esempio i prodotti notevoli. Una potenzialità del genere viene a nostro avviso amplificata notevolmente quando l'ambiente in questione è anche sviluppabile. In questo caso, proprio come nel caso delle macro di Cabrì, lo studente ha la possibilità di far crescere il micromondo, che rappresenta per lui/lei una parte della matematica, di pari passo con le proprie conoscenze (Mariotti 1998).

Infine, se reinterpretiamo gli ambienti sopra descritti nell'ambito delle teorie sui micromondi, essi possono essere classificati nel seguente modo: un AGM può essere

visto come un Macromondo, al pari della geometria della tartaruga (Noss & Hoyles 1996, p.g. 68), o del LOGO, in quanto capace di contenere più micromondi al suo interno. Un manipolatore simbolico ed un ambiente algebrico elementare sono entrambi dei micromondi, caratterizzati appunto da oggetti computazionali in grado di interagire tra loro, e da modelli di domini di conoscenza che possono essere investigati tramite l'interazione con il software. Un editore di testo invece manca di quest'ultima proprietà, quindi non può essere visto come micromondo, almeno in ambito matematico²⁹.

Le considerazioni didattiche a cui abbiamo accennato riguardano una classificazione piuttosto generale dei micromondi, ma, riteniamo che questi siano caratterizzati fortemente anche dalla tipologia dei comandi che mettono a disposizione dell'utente; nel prossimo paragrafo offriremo appunto dei criteri per individuare varie tipologie di comandi.

Classificazione dei comandi

Nell'ambito dei micromondi matematici è possibile classificare l'insieme di comandi a disposizione in funzione della relazione tra questi e la teoria matematica incorporata nel software. Una scelta di questo tipo è coerente con la caratterizzazione dei micromondi a cui abbiamo già accennato, e ci aiuterà ad esplicitare e confrontare alcune delle peculiarità didattiche de' L'Algebrista, e di altri software, analizzando i rapporti tra i singoli micromondi e le teorie che incorporano.

Comandi chiari e comandi oscuri: *un comando si dice **chiaro** quando il segno che lo rappresenta (sia esso una scritta o un'icona) suggerisce in maniera evidente e non ambigua la funzionalità del comando in questione. Vice versa un comando si dice **oscuro** quando il segno che lo rappresenta non dà indicazioni sulla funzionalità del comando.*

²⁹ Alcuni editori di testo possono essere invece visti come micromondi nell'ambito dell'apprendimento di un linguaggio in quanto offrono strumenti per analizzare la correttezza delle

Un comando chiarissimo è il tasto della lettera "a" sulla tastiera, in quanto, per lo meno nell'ambito della cultura attuale, per l'utente risulta immediato aspettarsi che premendo tale tasto sullo schermo apparirà una "a" da qualche parte. Il comando Solve (presente in vari manipolatori), invece, è oscuro in quanto l'utente inesperto in materia di fattorizzazione di espressioni non può sapere, a priori, quale sarà l'effetto dell'utilizzo del comando. Il tasto F7 infine, sempre sulla tastiera, risulta anch'esso oscuro.

Comandi procedurali e comandi relazionali: un comando si dice **procedurale** quando può essere utilizzato in uno ed un solo verso, si dice **relazionale** quando può essere utilizzato in due versi, "anche per tornare indietro".

Ad esempio generalmente i manipolatori simbolici offrono il comando Expand ed il comando Factor, il primo *svolge*, mentre il secondo *raccoglie a fattori*, questi due comandi sono l'uno l'inverso dell'altro, e sono entrambi *procedurali*. Nel caso de' L'Algebrista invece abbiamo un unico bottone per la proprietà distributiva, esso raccoglie o distribuisce a seconda che la testa dell'espressione su cui deve agire è una somma oppure un prodotto, questo tipo di comando si dice *relazionale* in quanto corrisponde ad una relazione di equivalenza tra espressioni, a tutti gli effetti.

Comandi elementari e comandi complessi: diciamo che un comando è **complesso** quando corrisponde a trasformazioni complesse. Vice versa un comando si dice **elementare** quando corrisponde ad una trasformazione elementare.

Anche in questo caso un esempio evidente di comando complesso è il comando Factor in quanto condensa in sé l'uso ripetuto della proprietà distributiva e della proprietà commutativa. Vice versa come esempi di comandi elementari basta considerare uno qualsiasi dei bottoni delle proprietà de' L'Algebrista.

Relazione tra comandi ed assiomi: la **relazione tra comandi e gli assiomi** da essi rappresentati si dice **biunivoca** quando l'applicazione che fa corrispondere gli uni negli altri è biunivoca.

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

In un manipolatore simbolico tale relazione non è biunivoca: manca l'iniettività in quanto i comandi non sono relazionali, manca la surgettività in quanto non vi sono comandi elementari.

La questione dell'interfaccia

Parte della discussione a questo proposito è già stata affrontata nei paragrafi precedenti, ma gli obiettivi didattici che hanno ispirato la progettazione de' L'Algebrista hanno ulteriori implicazioni a livello di interfaccia. Analizziamo tali aspetti in questo paragrafo onde stabilire ulteriori criteri per il confronto con altri software:

Interfaccia di tipo strutturale: l'ambito in cui distinguiamo questo tipo di interfaccia è quello degli ambienti che consentono di manipolare espressioni algebriche tramite un insieme di comandi predefiniti, e ci riferiamo specificamente all'interazione tra utente ed espressioni tramite i comandi di un software. Un'espressione algebrica è costituita da termini organizzati secondo una certa struttura ad albero, ed in particolare organizzati in sotto espressioni, che a loro volta sono termini, non che sotto alberi. Due espressioni equivalenti possono avere strutture molto diverse tra loro.

*Diciamo che una **trasformazione** di un'espressione è di tipo **strutturale** quando è effettuata su una sua sotto espressione ed ha come risultato una nuova espressione che differisce dall'originaria solo ed esclusivamente per il sotto albero rappresentato dalla sotto espressione su cui è effettuata la trasformazione.*

*Un'interfaccia si dice di **tipo strutturale** se permette all'utente di trasformare espressioni in maniera strutturale.*

Interfaccia completa: la trasformazione di un'espressione, in una equivalente, è caratterizzata dai seguenti elementi: espressione di origine, sotto espressione su cui si agisce, regola di trasformazione utilizzata, sotto espressione finale, espressione

finale. Nel caso la trasformazione sia effettuata tramite un manipolatore simbolico o un AAE, si parla di comando invece che di regola di trasformazione.

*Diciamo che un'interfaccia è **completa** quando fornisce all'utente informazioni su tutti gli elementi caratterizzanti una trasformazione.*

L'Algebrista un esempio di interfaccia completa in quanto, ogni volta che si effettua una trasformazione, all'espressione finale è associata una label che indica il comando utilizzato e la sotto espressione su cui questo era stato applicato; di conseguenza è possibile, a partire dall'espressione finale, ricostruire tutta la trasformazione.

Le definizioni che abbiamo appena dato, riguardo i tipi di interfaccia, possono essere generalizzate ed utilizzate per analizzare un qualsiasi software matematico in quanto si basano essenzialmente sulla relazione tra oggetti computazionali ed entità matematiche da essi rappresentati. Vedremo più avanti come in effetti si possa analizzare un software di geometria dinamica, come Cabri, sulla base di queste definizioni.

Infine osserviamo che sia la completezza che la strutturalità, insieme alla chiarezza di un comando, possono essere inclusi, sotto le voci *Predictability* e *Synthesizability* due principi caratterizzanti la facilità con cui viene appreso un'interfaccia, secondo la teoria della Human Computer Interaction (Dix et Al., 1998, p.g. 163). Il primo dei due termini sta ad indicare il supporto offerto all'utente per determinare l'effetto di azioni future sulla base della storia delle azioni passate; mentre il secondo termine indica il supporto offerto all'utente per capire l'effetto delle azioni passate sullo stato corrente dell'interfaccia. Ad esempio ne' L'Algebrista, grazie al particolare raggruppamento delle celle, è sempre possibile sapere se una data espressione appartiene all'ambiente editore di testo o se invece appartiene all'ambiente algebrico elementare (Synthesizability). Nel secondo caso il confronto tra la struttura (indicata dal selezionatore) di un'espressione e la simbologia che rappresenta un bottone, suggerisce quale sarà l'effetto del bottone sulla parte di espressione selezionata (Predictability); in particolare, se non fosse

chiaro che l'espressione si trova in ambiente AAE, l'effetto del bottone non sarebbe prevedibile.

Considerazioni didattiche sulla questione dell'interfaccia

Riteniamo che un'interfaccia di tipo strutturale sia fondamentale in un software diretto all'apprendimento della manipolazione simbolica in quanto:

- riduce notevolmente la distanza tra espressione rappresentata sullo schermo e concetto algebrico di espressione;
- riduce la distanza tra edizione di un'espressione e trasformazione algebrica di un'espressione.

A questo proposito appare lecito un parallelo con Cabrì; tra "disegni fatti a mano", "costruzioni di Cabrì" e "costruzioni geometriche" da una parte, e "edizione di un'espressione", "trasformazione di un'espressione in un AAE", "trasformazione algebrica" dall'altra. Se l'interfaccia di un AAE (o di un manipolatore simbolico) è di tipo strutturale, allora l'oggetto computazionale rappresentante l'espressione incorpora la struttura della stessa, analogamente a quanto accade per le costruzioni geometriche eseguite in ambiente Cabrì.

Una delle proprietà più apprezzate dei software di geometria dinamica, quali Cabrì e Geometer Sketch Pad, risulta essere il comando che fornisce la storia delle primitive (comandi di base) utilizzate per effettuare una costruzione geometrica. I risultati positivi dovuti a tale caratteristica sono già stati messi in luce (Mariotti et al., 1998). In effetti la storia delle operazioni eseguite offre allo studente la possibilità di valutare oltre al proprio prodotto, anche il modo in cui lo ha ottenuto; in particolare affinché si formino i concetti di espressione algebrica e di trasformazione algebrica riteniamo necessario supportare lo studente offrendogli la possibilità di investigare le relazioni tra gli elementi caratterizzanti un'espressione algebrica. Un'interfaccia di tipo completo offre appunto questa possibilità e rende possibile il confronto ad ogni istante tra espressione vecchia ed espressione nuova.

Costruzione del modello

A questo punto abbiamo tutti gli strumenti necessari per descrivere e schematizzare il paradigma di software rappresentato da L'Algebrista. Procederemo analizzando il prototipo sulla base dei concetti e della terminologia introdotti nei paragrafi precedenti e fornendo una giustificazione didattica per ciascuna delle nostre scelte. Ogni paragrafo corrisponderà ad un criterio di classificazione.

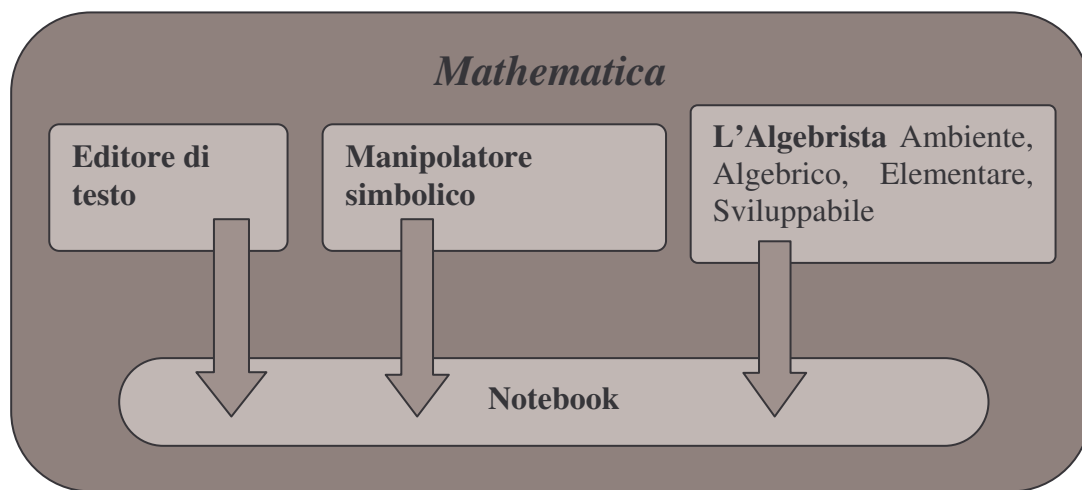
Gli obiettivi didattici

Gli obiettivi didattici alla base del prototipo sono l'introduzione degli studenti alla manipolazione simbolica, e l'introduzione degli stessi all'idea di teoria e di dimostrazione. Nella definizione del modello inseriamo anche questi obiettivi in quanto sono all'origine della definizione dello stesso e lo differenziano rispetto a i modelli relativi ad altri software già presenti in commercio.

L'ambiente

L'Algebrista è un micromondo inserito in un Ambiente Globale Multifunzionale, *Mathematica*, il quale a sua volta può contenere altri sotto ambienti, o comunque sintetizza funzionalità tipiche di ambienti anche molto diversi tra di loro. Questa globalità e multi funzionalità di *Mathematica* rende disponibili strumenti anche di genere molto diverso, ma che non sono in conflitto tra di loro e che possono essere utilizzati contemporaneamente; in questo modo l'utente può godere di una notevole libertà di azione e di espressione (nell'ambito della comunicazione con il computer). In particolare, per quanto ci riguarda, *Mathematica* contiene, anzi è, un ambiente che funziona da manipolatore simbolico (contiene tutti i normali comandi standard³⁰, tutti molto potenti), e contiene anche un editore di testo. All'interno di questo ambiente abbiamo appunto inserito L'Algebrista che è un Ambiente Algebrico Elementare Sviluppabile. I tre sotto ambienti sono tutti simultaneamente disponibili ad ogni istante all'interno del foglio di lavoro, che prende il nome di *notebook*.

³⁰ In particolare include al suo interno tutte le funzionalità di DERIVE.



Gli ambienti relativi a L'Algebrista

La scelta di fondere un editore di testo, un manipolatore simbolico ed un AAE, all'interno di un AGM, permette di avvalersi di alcune caratteristiche di ciascuno di loro, in particolare:

- La possibilità di editare espressioni algebriche e di inserire commenti in ogni punto del foglio di lavoro ed in ogni momento (grazie all'editore di testo);
- La possibilità di trasformare espressioni coerentemente con gli assiomi dell'algebra tramite comandi elementari (grazie al AAE L'Algebrista);
- La sviluppabilità (grazie al Teorematore de' L'Algebrista³¹);
- La possibilità di trasformare espressioni coerentemente con gli assiomi dell'algebra tramite comandi complessi (grazie al manipolatore simbolico);
- La possibilità di interagire con altri micromondi matematici (grazie a Mathematica);

L'editore di testo garantisce allo studente una totale libertà di azione all'interno del foglio di lavoro (notebook) favorendo la rivisitazione, da parte dell'utente, dei percorsi seguiti e, sotto opportuni stimoli, la verbalizzazione dei processi euristici attivati durante lo svolgimento di un'attività. Esso in effetti risulta essere

³¹ In effetti anche *Mathematica* offre la possibilità di creare nuovi strumenti di manipolazione simbolica, ma per farlo è necessario avere una buona conoscenza del suo linguaggio di

un'estensione dell'ambiente *carta e penna* con il quale gli studenti sono soliti interagire spontaneamente.

L'Ambiente Algebrico Elementare permette di muoversi all'interno della teoria data tramite gli assiomi che l'utente ha la sensazione di *usare* veramente grazie agli oggetti computazionali che li rappresentano. Coerentemente con la teoria di Sfard, gli assiomi diventano oggetti, dei riferimenti fisici ben precisi per lo studente, il quale, quindi, può avvalersene più facilmente durante attività di verbalizzazione. Nell'ambiente L'Algebrista (AAE), le trasformazioni possibili sono solo quelle corrette nell'ambito della teoria, ovvero solo quelle ottenibili tramite l'uso degli assiomi, dei comandi. Questo ambiente può intrecciarsi con l'editore di testo in quanto è possibile scrivere tra una cella e l'altra de' L'Algebrista, tra un'espressione e l'altra. Ciò nonostante i due ambienti sono ben distinti e l'utente è sempre in grado di distinguere tra scritture dell'ambito AAE e scritture dell'ambito editore di testo³². Questa netta linea di demarcazione, costruita dal comando *Inserisci Espressione*, costringe lo studente a riflettere di volta in volta sulle operazioni che è lecito fare in ciascuno degli ambienti, contribuendo a forgiare l'idea di teoria come ambiente determinato dalle regole che lo definiscono. In particolare la manipolazione simbolica risulta di conseguenza governata dalle trasformazioni consentite dai comandi a disposizione nell'ambiente AAE, ovvero gli assiomi. In altre parole L'Algebrista è un *ambiente computazionale auto espressivo*, in quanto in esso l'unico modo di manipolare oggetti (espressioni ed assiomi) consiste nell'esplicitare le relazioni tra gli oggetti stessi.

"Autoexpressive computational environments: that is, environments in which the only way to manipulate and reconstruct objects is to express explicitly the relationship between them" (p.g. 207 Noss, Healy, Hoyles, 1997).

Ambienti di questo tipo costituiscono dei mondi dove è possibile manipolare oggetti e riflettere sulle relazioni tra di essi; l'idea è che concentrandosi sul

programmazione, quindi questo software non rientra nella nostra definizione di *svilupabile*. Vice versa L'Algebrista grazie al *Teorematore* risulta essere svilupabile.

³² In questo L'Algebrista si distingue da Cabri nel quale non vi è alcuna distinzione evidente tra disegni e costruzioni geometriche, tra ambiente di edizione di disegni ed ambiente di costruzione (basato sugli assiomi) di figure geometriche

linguaggio (di manipolazione) come meccanismo per controllare gli oggetti, si cerca di rendere concreta l'algebra delle relazioni (Noss, Healy, Hoyles, 1997).

Per quanto riguarda la sviluppabilità, coerentemente con l'obiettivo di introdurre gli studenti all'idea di teoria e di dimostrazione, essa permette allo studente di creare nuovi comandi, ovvero di introdurre nuovi teoremi. Il software non offre alcun controllo automatico sulla correttezza del teorema, in quanto questo andrebbe contro il concetto di sviluppabilità. Per poter essere sviluppabile, un micromondo, non deve basarsi su un insieme di assiomi predeterminato e fisso, poiché in tale caso per l'utente sarebbe impossibile sostituire vecchi comandi con i nuovi o definire un nuovo insieme di assiomi. Di conseguenza nell'ambito di un ambiente sviluppabile, per controllare la correttezza di un teorema bisognerebbe fornire di volta in volta la lista degli assiomi rispetto al quale vogliamo validare il teorema. Un passaggio di questo tipo non è da escludere a priori, ma appare molto complesso, e comunque il controllo dovrebbe necessariamente avvenire su richiesta dell'utente poiché se fosse automatico allora sarebbe impossibile definire una teoria a partire da un insieme vuoto di assiomi. Per questi motivi abbiamo deciso di non fornire alcun strumento di controllo automatico nel prototipo de' L'Algebrista, ma preferiamo scaricare la responsabilità della dimostrazione di un eventuale teorema sull'utente, il quale ad ogni modo può avvalersi dei comandi, ovvero degli assiomi.

Infine osserviamo che l'Ambiente Globale Multifunzionale in cui è inserito rende possibile il confronto tra la teoria locale rappresentata da L'Algebrista ed altre teorie matematiche, in particolare facilita il confronto con gli strumenti complessi di un manipolatore simbolico il quale può andare ad integrarsi con L'Algebrista per offrire un insieme completo di strumenti per la manipolazione simbolica.

Teoria di base

La teoria incorporata da L'Algebrista si basa sulla teoria delle espressioni algebriche. L'insieme degli assiomi non è però fisso ed univocamente determinato, è infatti possibile scegliere a piacere i comandi (quindi assiomi) con cui lavorare. Per questo motivo non elenchiamo in questa sede gli assiomi, ma rimandiamo questo compito al capitolo sulla sperimentazione del prototipo in cui descriveremo un

percorso didattico dove inizialmente l'insieme dei comandi è piuttosto ristretto, ma si evolve di volta in volta con l'aggiunta di assiomi e di teoremi (chiamiamo teoremi i comandi corrispondenti a proposizioni dimostrate dagli alunni ed assiomi tutto ciò che è fornito dall'alto senza essere dimostrato).

Tipologia dei comandi e relazione con gli assiomi

Una delle caratteristiche fondamentali degli assiomi dell'algebra è che sono relazioni di equivalenza, ed ancora, molte definizioni, possono essere espresse sotto forma di relazione di equivalenza; per questo motivo abbiamo deciso di basare L'Algebrista su comandi di tipo relazionale. Nel caso in cui la trasformazione a cui si riferisce un bottone non sia esprimibile tramite una relazione di equivalenza, allora il comando in questione sarà procedurale.

Ancora una volta quindi L'Algebrista si distingue dai manipolatori simbolici classici in quanto questi sono in genere basati su comandi di tipo procedurale³³.

Ritenendo fondamentali gli assiomi (e definizioni), nel quadro della teoria dell'*object mediation* (Sfard, 1997), abbiamo cercato di dare loro un ruolo cruciale nell'ambito del software conferendo loro lo status di oggetti di base attraverso i quali passa l'interazione utente computer durante le attività di manipolazione delle espressioni. Abbiamo perciò deciso di basare l'interazione su bottoni rappresentanti appunto gli assiomi e che permettano di trasformare un'espressione selezionandola e cliccando sul bottone. In questo modo ogni assioma sottostante una trasformazione deve necessariamente essere esplicitato: in effetti affinché la trasformazione abbia luogo bisogna passare attraverso il bottone rappresentante l'assioma stesso. Inoltre il bottone rappresenta un riferimento visivo (nonché manipolabile, almeno indirettamente) per l'utente, facilitando così lo studente nel compito di verbalizzare le sue azioni facendo riferimento agli assiomi tramite i bottoni. In questo senso siamo di fronte ad un esempio tipico di *object mediation* (Sfard).

Dato il ruolo di rappresentanti, ed evocatori di assiomi, che volevamo attribuire ai bottoni, coerentemente con il principio di completezza, abbiamo deciso di

³³ Osserviamo che questa forse più che una caratteristica è una necessità in quanto tali software sono basati su comandi complessi, quindi non necessariamente, e soprattutto non banalmente, invertibili.

rappresentarli tramite due tipi di iconografia, una verbale, ed una basata sulla simbologia matematica. Quest'ultima offre in effetti un'indicazione visiva piuttosto efficace dell'effetto del bottone, in questo modo i comandi risultano essere chiari. Infine osserviamo che in questo modo offriamo allo studente essenzialmente tre "simboli" per ogni assioma, adatti a tre diversi tipi di "discorso":

- il bottone, adatto all'interazione con la macchina;
- il nome dell'assioma (iconografia verbale) adatto al linguaggio verbale;
- la formula espressa in simboli matematici, adatti al linguaggio algebrico.

Tramite questa pluralità di rappresentazioni intendiamo favorire il processo per cui gli assiomi diventano effettivamente parte costituente della verbalizzazione degli studenti.

Infine ricordiamo che, in quanto Ambiente Algebrico Elementare Sviluppabile, L'Algebrista si basa su comandi di tipo elementare (eventualmente condensabili in nuovi comandi più complessi); questa proprietà, unita alla relazionalità dei comandi, rende biunivoca la relazione tra gli stessi e gli assiomi che rappresentano.

Tipologia dell'interfaccia

La prima conseguenza rilevante degli obiettivi didattici sul tipo di interfaccia è che questa sia strutturale in quanto in questo modo, le espressioni rappresentate ne L'Algebrista, diventano oggetti computazionali molto vicini al concetto matematico di espressione algebrica. La distanza tra l'oggetto matematico e la sua rappresentazione sullo schermo si riduce drasticamente, ma non basta: un'espressione algebrica non è caratterizzata solo dalla propria struttura ad albero, ma anche dall'insieme di espressioni ad essa equivalente, nel quale ci si può muovere attraverso trasformazioni algebriche. L'Algebrista è stato realizzato in modo che ogni volta che si trasforma un'espressione, tramite un assioma, la nuova espressione è riportata immediatamente sotto quella originaria, senza alcuna gerarchia tra le due; si può quindi creare tutta una serie di espressioni equivalenti tra di loro e tornare a modificare una qualsiasi di esse in qualsiasi momento. La cronologia delle trasformazioni non è rispettata, in questo modo si elimina ogni sorta di gerarchia all'interno di un insieme di espressioni equivalenti. Ciononostante, come già detto,

riteniamo che sia molto importante fornire all'utente la possibilità di esaminare la *storia* delle trasformazioni effettuate su un'espressione. Per fornire una storia delle trasformazioni, evitando di instaurare gerarchie implicite tra le espressioni, nel caso de' L'Algebrista, sarebbe sufficiente aggiungere una facilitazione per registrare la sequenza delle trasformazioni effettuate e renderla visibile su richiesta dell'utente. Il prototipo usato per la sperimentazione, per motivi tecnici, non ha potuto avvalersi di una tale facilitazione, ma progettiamo di aggiungerla per future sperimentazioni.

Ad ogni modo la versione attualmente usata de' L'Algebrista offre un'interfaccia di tipo completo³⁴, in quanto fornisce traccia, all'utente, dei comandi utilizzati per ogni trasformazione effettuata. L'idea è che lo studente possa in ogni istante confrontare la nuova espressione con quella/e vecchia/e. Per questo motivo abbiamo realizzato L'Algebrista in modo che il risultato di una trasformazione sia una nuova espressione che viene inserita sotto quella originale, che sia accompagnata da una label che indichi il tipo di comando utilizzato e la sotto espressione su cui questo è stato applicato. Chiaramente si possono fare in proposito scelte diverse, ma riteniamo opportuno che alla nuova espressione siano associati, in maniera esplicita e visibile, tutti gli elementi caratterizzanti la trasformazione, ovvero: espressione di origine, sotto espressione su cui si è agito, comando utilizzato, sotto espressione finale, espressione finale. In questo modo si riduce la distanza tra una trasformazione algebrica e la corrispondente trasformazione all'interno del micromondo. Infatti la completezza dell'interfaccia aiuta a non cortocircuitare il processo e a non fissarsi sul prodotto.

Il modello

Riportiamo uno schema che sintetizzi le caratteristiche fondamentali del modello al quale si riferisce L'Algebrista. Esso è da intendersi come un insieme di linee guida per la progettazione di software didattici per l'algebra, ma hanno valenza più generale, in quanto i criteri utilizzati possono essere estesi ad un qualsiasi settore dell'insegnamento della matematica come mostreremo in un prossimo paragrafo in cui confronteremo L'Algebrista con Cabri.

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

Modello de' L'Algebrista	
Obiettivi didattici	<ul style="list-style-type: none"> • Introduzione degli studenti alla manipolazione simbolica • Introduzione degli studenti all'idea di teoria e di dimostrazione
Tipologia	Ambiente Algebrico Elementare Sviluppabile
Ambiente in cui è immerso	Ambiente Globale Multifunzionale
Ambienti con cui interagisce	<ul style="list-style-type: none"> • Manipolatore Simbolico • Editore di testo
Teoria di base	Teoria delle espressioni algebriche
Relazione tra comandi ed assiomi	Biunivoca
Tipologia dei comandi	Elementare Relazionale Chiara
Tipologia dell'interfaccia	Strutturale Completa

L'Algebrista e l'object-mediation

Il nostro micromondo dà *vita propria* ad alcune entità matematiche rendendole accessibili allo studente attraverso l'interazione diretta con il computer. In particolare sono percettivamente accessibili gli assiomi dell'algebra (bottoni delle proprietà) e gli oggetti dell'algebra, le espressioni, ma sono anche accessibili le relazioni tra espressioni ed assiomi. L'assioma diventa a sua volta uno *strumento* (in quanto oggettualizzato nel bottone) per modificare le espressioni, e questa interazione avviene per manipolazione diretta; le entità matematiche in questione quindi, insieme con le loro relazioni, acquisiscono una percettibilità ed una maneggiabilità, che le rendono accessibili all'utente e ne mediano i significati. Osserviamo inoltre che le relazioni tra diverse espressioni e tra espressioni ed assiomi vengono esplicitate solo grazie all'interazione utente-micromondo, esse non sono illustrate esplicitamente; ne segue che l'utente, se le attività sono opportunamente progettate costruisce i

³⁴ Vedi par. "Classificazione dei comandi"

significati in base all'interazione con il micromondo, tramite l'uso dei bottoni e il feedback offerto dalla macchina. Riallacciandoci alla teoria dell'object-mediation, già descritta nei primi capitoli, possiamo descrivere il tipo di interazione e la mediazione offerti da L'Algebrista:

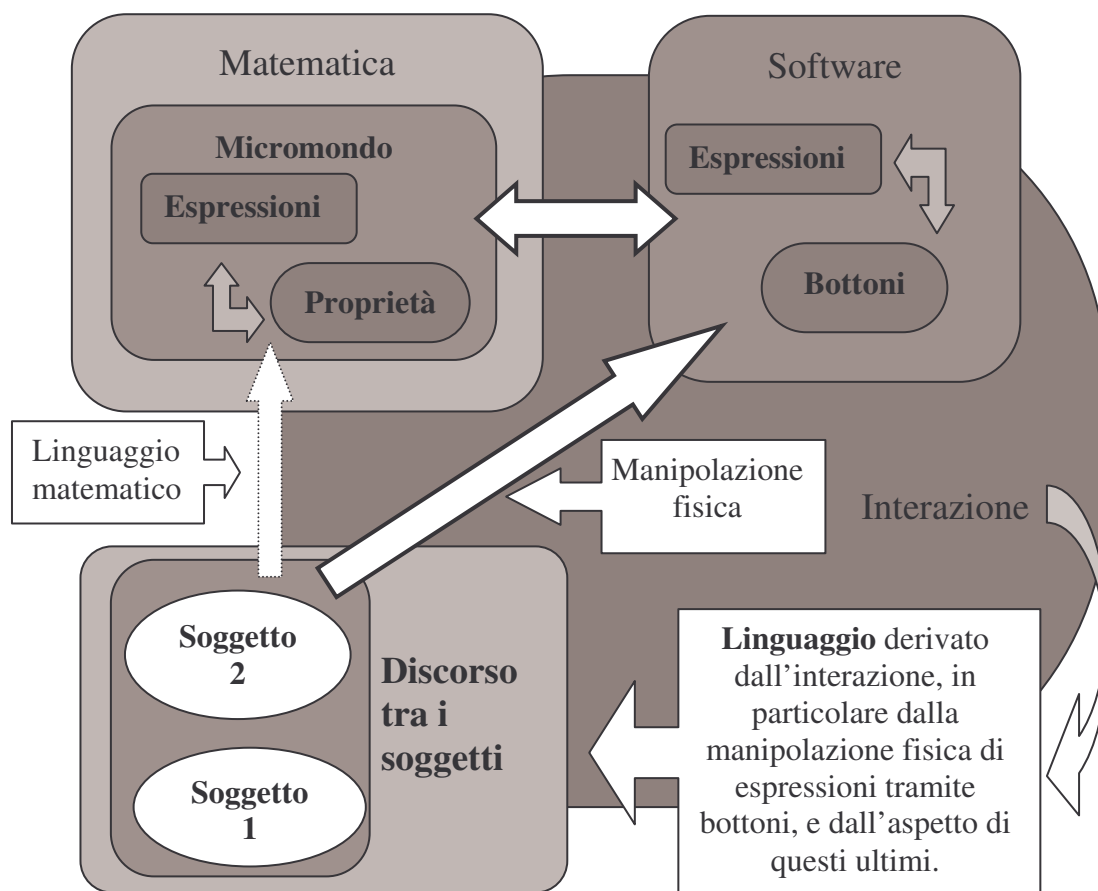
1. I riferimenti percettivi per quanto riguarda gli assiomi e ad altre possibili regole di trasformazione di espressioni forniscono ai ragazzi riferimenti, condivisibili, utili nei loro discorsi di tipo matematico facendo sì che questi siano sempre *in focus* ed *integrated*.
2. Il fatto che il riferimento percettivo di cui sopra sia in realtà un riferimento fisico, sono dei veri e propri bottoni da pigiare (cliccare), fa sì che il linguaggio tramite il quale gli studenti interagiscono con il software sia di tipo AR e favorisca l'oggettualizzazione. In pratica è come se si fosse data agli studenti la possibilità di esprimersi in un linguaggio a loro familiare ma parlando di cose nuove che stanno imparando e per le quali non avrebbero altrimenti avuto un linguaggio.
3. I bottoni in questione non solo hanno una loro fisicità ed una loro funzione, ma hanno anche dei simboli che li rappresentano (verbali e matematici) che richiamano la loro funzione, quindi sono per lo studente un riferimento percettivo utile perché contengono un suggerimento esplicito a riguardo della loro funzione.
4. Un modo per interpretare la mediazione offerta dai bottoni è il seguente.

La funzione del bottone è il corrispondente, all'interno di questo micromondo, della proprietà che esso rappresenta. Il micromondo corrisponde ad un "corpus" di conoscenze matematiche. I simboli scritti sul bottone appartengono al linguaggio della comunità matematica e forniscono modi di rappresentare la proprietà in tale linguaggio.

Quando uno studente agisce su un'espressione tramite un bottone, egli agisce all'interno della matematica (rappresentata dal micromondo), fa sì che un'entità matematica (la proprietà) agisca su un'altra entità matematica secondo un paradigma di interazione proprio della matematica. In altre parole quello che avviene è un discorso matematico, cioè che avviene all'interno della matematica. Questo discorso è costituito dall'interazione tra le entità in questione, proprietà ed espressione. Esso è però specificato in termini *fisici* (si seleziona parte dell'espressione e si clicca sul

bottone), ovvero secondo un discorso di tipo AR, ma in più può essere tradotto in linguaggio verbale o scritto tramite il riferimento al bottone ed ai simboli su di esso rappresentato. In sostanza, il bottone, in questo gioco di traduzioni, funziona da tramite tra discorso matematico e discorso verbale (orale e/o scritto). In questo senso esso media il proprio significato offrendo la possibilità di essere usato e contemporaneamente di essere incluso in discorsi verbali.

5. Quando un nuovo simbolo viene introdotto, si cerca automaticamente di



La mediazione offerta da L'Algebrista

interpretarlo rifacendoci a significati e schemi di utilizzo a noi già familiari, magari propri di altri simboli: in altre parole il nuovo simbolo viene utilizzato in maniera templates driven. Questo comportamento si ritrova quando un'utente lavora con L'Algebrista: egli impara ad usare certi assiomi tramite il "discorso" della manipolazione diretta dei bottoni e delle espressioni. Quando un nuovo assioma (o teorema, o definizione) viene inserito, l'utente ha già a disposizione

un tipo di *frase* in cui usare il nuovo bottone; è sufficiente riempire il buco del discorso "*seleziono l'espressione e clicco _____ per ottenere una nuova espressione equivalente*" per ottenere un modo di utilizzo del nuovo teorema, magari anche senza riflettere sul possibile effetto del bottone in questione o senza fare previsioni. Questo modo di utilizzo corrisponde alla fase *templates-driven* (Sfard, 1997). Questa fase può durare molto poco, in effetti non appena l'utente ha ottenuto un qualche cambiamento nell'espressione, quando deciderà di riutilizzare il bottone in questione, lo farà prevedendo cosa potrà accadere, quindi ne farà un uso non più *templates driven*.

L'Algebrista e Cabrì Géomètre

I due software in questione a prima vista possono sembrare profondamente diversi in quanto il primo è un micromondo disegnato per l'insegnamento dell'algebra, mentre il secondo ha come obiettivo l'insegnamento della geometria. Ciononostante bisogna rilevare che, in effetti, L'Algebrista è stato progettato e realizzato facendo spesso riferimento a Cabrì e ad esperienze didattiche basate su di esso (Mariotti 1996,1998); in effetti i principi didattici su cui si basano sono molto simili. Entrambi i software si possono inquadrare nell'ambito della teoria dell'*object-mediation* e possono essere utilizzati con un comune obiettivo didattico: introdurre gli studenti all'idea di teoria e di dimostrazione; per perseguire tale obiettivo si passa, nel caso de' L'Algebrista, attraverso l'introduzione alla manipolazione simbolica, mentre nel caso di Cabrì si passa attraverso l'idea di costruzione geometrica. Il punto chiave che accomuna i due software è che entrambi offrono rappresentazioni sullo schermo di entità matematiche (espressioni in un caso, figure geometriche nell'altro) sulle quali l'utente può agire applicando assiomi o teoremi, anch'essi rappresentati sullo schermo tramite comandi. Il tipo di interazione è molto simile e si basa in entrambe i casi sul principio di selezione di oggetti matematici tramite il mouse (anche la tastiera ne' L'Algebrista) e l'applicazione di assiomi o teoremi cliccando sul comando apposito nei relativi menu. In Cabrì i comandi sono distinti in *primitive* e

macro, le prime sono da considerarsi come gli assiomi della teoria incorporata dal micromondo in questione, mentre le seconde sono da considerarsi come teoremi in quanto possono essere create dall'utente tramite l'uso delle primitive. In altre parole, secondo la terminologia introdotta nei paragrafi precedenti, Cabrè è un ambiente sviluppabile basato su comandi di tipo elementare, proprio come L'Algebrista.

Per quanto riguarda gli ambienti in relazione con i due software, osserviamo che come L'Algebrista è immerso in *Mathematica*, anche Cabrè, specialmente nelle sue ultime versioni (Cabrè II e nella T-92), ingloba funzionalità tipiche di micromondi di vario tipo, ad esempio offre facilitazioni per la creazione di grafici di funzioni; di conseguenza attualmente Cabrè risulta essere un ambiente globale multifunzionale che, in particolare, ingloba un ambiente per le costruzioni geometriche, ovvero il Cabrè originale. In questa sede non ci occupiamo della potenza dei micromondi che ruotano intorno al cuore dei due software, riteniamo invece più interessante distinguere il modo in cui essi sono in relazione con ambienti di tipo *editore*. Ricordiamo che una distinzione fondamentale tra un editore di testo ed un manipolatore simbolico (equivalentemente un AAE) è che il primo non è regolato da alcuna teoria, mentre il secondo si appoggia e si comporta conformemente con la teoria matematica che incorpora. In L'Algebrista i due tipi di ambienti convivono all'interno di uno stesso notebook (foglio di lavoro), e possono intrecciarsi, ma data una qualsiasi porzione di foglio di lavoro è sempre possibile stabilire se appartiene all'uno o all'altro ambiente; inoltre i comandi da utilizzare in un ambiente sono distinti nettamente da quelli validi per l'altro ambiente. In questo modo l'utente sa sempre cosa è lecito fare ed in quale ambiente è lecito farlo, di conseguenza la distinzione tra trasformazioni algebriche e edizioni di espressioni risulta esplicita e netta. Infine il passaggio di un'espressione dall'ambiente editore di testo all'ambiente AAE è sancito irrevocabilmente dall'uso del comando *inserisci espressione*.

Per quanto riguarda Cabrè, possiamo distinguere tra un ambiente edizione costituito dal foglio di lavoro con i comandi di *disegno*, ovvero quelli che non sono basati sugli assiomi della geometria, ed un ambiente geometrico costituito dal foglio di lavoro e dalle primitive (e *macro*) che incorporano gli assiomi della geometria. I due ambienti sanciscono anche in questo caso la differenza tra operazioni all'interno

di una teoria ed operazioni che non si appoggiano su alcun assioma matematico, ma dobbiamo rilevare che in Cabrì non sono altrettanto distinti quanto lo sono ne' L'Algebrista. In effetti l'ambiente di disegno e l'ambiente geometrico condividono lo stesso foglio di lavoro e non vi è alcun accorgimento di interfaccia che indichi quali tipi di comandi sono stati utilizzati per produrre una figura. In altre parole, data una figura sullo schermo, è impossibile stabilire se sono stati utilizzati comandi di disegno o geometrici, a meno di avvalersi della funzione di trascinamento. Di conseguenza la distinzione tra i due tipi di ambienti non è immediata come lo è ne' L'Algebrista anche perché Cabrì non impedisce di costruire una figura utilizzando entrambi i tipi di comando, mentre ne' L'Algebrista, in ambiente matematico (AAE) non si possono usare comandi dell'ambiente edizione, e vice versa.

In conclusione, entrambi i software obbligano l'utente a distinguere i due tipi di ambiente, solo che mentre L'Algebrista lo fa ad ogni momento dell'interazione, Cabrì lo fa solo al momento della validazione della figura tramite il dragging, quindi non supporta l'utente durante tutta l'interazione, ma solo in una fase particolare e su richiesta dello stesso. Osserviamo che L'Algebrista non necessita di alcun strumento per la validazione, in quanto, quando lo studente si muove all'interno dell'ambiente AAE, sono consentite solo operazioni *corrette*; la sperimentazione ha in effetti rivelato che L'Algebrista può assumere anche il ruolo di validatore di trasformazioni effettuate dagli studenti in altri ambienti (es. carta e penna).

Per concludere analizziamo la questione della completezza. Ricordiamo che L'Algebrista è completo in quanto, per ogni espressione ottenuta da un'altra, fornisce informazioni su tutti gli elementi caratterizzanti la trasformazione. Analogamente Cabrì può essere classificato completo in quanto, tramite il comando *history*, fornisce all'utente tutte le informazioni che caratterizzano una costruzione geometrica. Anche in questo caso, però, osserviamo che L'Algebrista e Cabrì si differenziano in quanto il primo supporta l'utente in ogni istante ed automaticamente, mentre Cabrì lo fa solo quando l'utente lo richiede. Osserviamo invece che i due software risultano essere equivalenti sul piano del supporto fornito all'utente riguardo le strutture degli oggetti matematici in questione: il selezionatore de' L'Algebrista indica la struttura delle

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

espressioni, mentre in Cabrè la funzione di trascinamento, e la selezione (per applicare primitive), suggeriscono la struttura di una figura geometrica.

L'Algebrista ed i linguaggi di programmazione

Uno dei punti di forza delle teorie didattiche basate sull'uso di linguaggi di programmazione nell'insegnamento della matematica è la considerazione che per scrivere o interpretare un programma, lo studente, deve necessariamente passare attraverso un formalismo *in qualche modo legato* alla matematica ed al suo linguaggio. Chiaramente, se gli ambienti di programmazione, e le attività proposte in essi, non sono studiati appositamente, si rischia che lo studente alla fine abbia appreso poca matematica ed abbia, più che altro, acquisito un certo gusto (o odio a seconda dei risultati) per il formalismo informatico. Esistono però in letteratura moltissimi esempi interessanti di attività o micromondi, basati sull'idea di programmazione, che danno risultati molto positivi. La principale corrente di pensiero in proposito si basa sull'uso di LOGO; un esempio piuttosto interessante è quello di *Mathsticks*. Si tratta di un micromondo realizzato e sperimentato da Noss, Healy e Hoyles, (1997) in cui l'utente può agire sia per manipolazione diretta che programmando in LOGO; nel primo caso il computer produce automaticamente il listato del programma che soggiace alle operazioni effettuate dallo studente. Osserviamo però che in questo caso, come per ogni attività basata su linguaggi imperativi³⁵, ci si basa essenzialmente su operazioni procedurali: si danno degli ordini al computer ed il computer gli esegue. I linguaggi procedurali difficilmente possono introdurre al pensiero relazionale, quindi, all'idea di teoria e di dimostrazione. Per questo scopo appaiono più appropriati linguaggi di tipo logico³⁶ dove un programma non corrisponde ad un insieme di ordini, ma ad un insieme di regole di comportamento alle quali il computer si deve attenere (ovvero degli assiomi su cui basarsi); una volta date le regole il computer interpreta un input basandosi su

³⁵ I linguaggi imperativi sono quelli in cui il *programma* corrisponde ad una serie di ordini che il computer deve eseguire; esempi piuttosto noti sono: LOGO, Pascal, C, Fortran, Basic.

di esse e l'output non sarà altro che l'input trasformato tramite le regole/assiomi di comportamento. Inoltre è l'utente stesso che può definire le regole, di conseguenza la programmazione in un linguaggio logico³⁷ risulta essere molto più vicina alla pratica matematica, di quanto non lo sia la programmazione in un linguaggio imperativo.

Un problema comune ad ogni tipo di linguaggio è costituito dalla distanza linguistica che lo divide dal linguaggio della matematica, di conseguenza uno dei nostri scopi, nella realizzazione de' L'Algebrista, era creare un interfaccia che riducesse tale distanza. In effetti L'Algebrista fornisce un'interfaccia che permette all'utente di utilizzare la simbologia matematica per interagire con un ambiente di programmazione logica basato sulla teoria delle espressioni algebriche. In particolare il *Teorematore* permette di creare nuovi assiomi, nuove regole di comportamento, ovvero di *programmare* nel senso della programmazione logica.

Infine, osserviamo che il micromondo L'Algebrista può essere interpretato come un ambiente per la programmazione ad oggetti, comprendente quindi: un ambiente, degli oggetti, un linguaggio di interazione tra gli oggetti, ed un linguaggio di interazione tra l'utente ed il micromondo. Descriviamo le varie componenti:

- **L'ambiente:** negli OOPL³⁸ c'è sempre un ambiente, un mondo dove gli oggetti in questione *prendono vita* nel senso che possono essere creati ed interagire tra di loro (tramite l'apporto umano oppure no), ed in particolare più oggetti uno uguale all'altro (figli magari dello stesso file sorgente) possono essere creati, come se fossero più persone della stessa razza. Nel nostro caso abbiamo:
 - ◊ Un ambiente globale che è costituito dal software *Mathematica* (anche se più in generale potremmo individuare come ambiente globale il computer stesso) e che contiene ogni altro sottoambiente del prototipo in questione ed ogni suo oggetto
 - ◊ La/le finestra/e di lavoro (o notebook), dove gli studenti hanno libertà assoluta nei termini di quello che *Mathematica* permette loro di fare

³⁶ Un esempio è il Prolog.

³⁷ Abbiamo parlato di linguaggi logici, ma intendevamo riferirci, con motivazioni analoghe, anche a linguaggi di tipo funzionale.

³⁸ Object Oriented Programming Language

- ◇ La cella di lavoro (ambiente AAE) che viene creata all'interno di una finestra quando un'espressione viene inserita tramite il bottone *inserisci espressione*. Essa è l'ambiente di lavoro specifico del prototipo da noi creato, è solo qui dentro che può avvenire l'interazione tra l'utente e la teoria rappresentata dal nostro prototipo. Di celle come questa ne possono essere create tante a piacere.
 - ◇ Il Teorematore: un ambiente in cui è possibile, usando la simbologia matematica, creare nuovi bottoni (i quali a loro volta sono oggetti), ovvero nuovi assiomi, nuove regole di comportamento che il computer dovrà utilizzare su richiesta dell'utente.
- Gli **Oggetti** del nostro micromondo sono:
 - ◇ Le celle di cui sopra, come in ogni ambiente di programmazione ad oggetti che si rispetti, ogni cosa può essere un oggetto. Le celle contengono espressioni e vi si può agire tramite i bottoni.
 - ◇ Le espressioni sono contenute nelle finestre e/o nelle celle
 - ◇ I bottoni sono contenuti in una o più apposite finestre chiamate palette. Essi agiscono sulle finestre, sulle celle e soprattutto sulle espressioni. Osserviamo che possono essere create quante copie vogliamo delle palette dei bottoni. I bottoni non interagiscono tra di loro, l'unico caso in cui questo può avvenire è in ambiente *Teorematore* dove si usano una serie di bottoni apposti per creare nuovi bottoni.
 - Il **linguaggio di interazione tra gli oggetti**: sostanzialmente questo tipo di interazione è rilevante solo nel caso dell'interazione bottone-espressione. Esso è unidirezionale, il bottone agisce sull'espressione trasformandola. Il *nome* del bottone condensa in sé il tipo di azione che è definita nel bottone.
 - Il **linguaggio di interazione tra l'utente ed il micromondo**: a parte la normale interazione utente-*Mathematica*, che c'è quando l'utente agisce su una normale finestra, l'interazione avviene per manipolazione diretta una volta che si è in una delle celle di lavoro di cui sopra. L'utente può selezionare le parti di espressione

su cui agire e *premere* (cliccare) un bottone affinché questo trasformi la parte di espressione in questione. Questo linguaggio di interazione (considerate tutte le componenti sopra elencate) è un vero e proprio OOPL poiché permette di creare oggetti (es. inserendo un'espressione o anche agendo su un'espressione, questa viene sempre duplicata), agire su di essi direttamente, e fa interagire l'uno con l'altro.

Per concludere, torniamo all'esempio di *Mathsticks*. In *Mathsticks* l'utente ha la possibilità di usare due diversi linguaggi per agire sul micromondo, la programmazione verbale e la manipolazione diretta; inoltre lo studente può agire in un modo ed interpretare i risultati dell'interazione nell'altro, passando dalla finestra grafica a quella testuale e vice versa. In *L'Algebrista* c'è un modo solo di agire sugli oggetti del micromondo: la manipolazione diretta; ma, tramite la mediazione dei nomi dei bottoni, la manipolazione condensa in se una sorta di programmazione verbale. Tale programmazione può essere resa esplicita chiedendo ai ragazzi di verbalizzare in qualche modo (scritto o orale) le loro azioni o più semplicemente può essere resa esplicita dalla *traccia dei bottoni usati*. In particolare, osserviamo che in questo caso il linguaggio di programmazione verbale in questione non è altro che un sotto insieme del linguaggio dell'algebra, quindi, a nostro avviso, molto più utile del LOGO e dei più popolari linguaggi di programmazione, tutti linguisticamente molto distanti dal linguaggio matematico.

Il percorso didattico

Riportiamo tutta la sequenza, in ordine cronologico, delle consegne sottoposte agli studenti. La sigla CL sta per "attività svolta in classe", mentre CS indica le attività svolte a casa dai ragazzi. Le consegne sono state raggruppate a seconda della data del giorno in cui sono state presentate ai ragazzi. In quasi tutte le attività svolte in classe gli studenti sono stati liberi di scegliere il tipo di approccio e gli strumenti che preferivano (ad esempio L'Algebrista o "carta e penna"). Sono esclusi alcuni casi in cui abbiamo indicato nella consegna il tipo di approccio e/o il tipo di strumento a cui riferirsi. Per quanto riguarda le attività svolte a casa, in nessuna di esse era richiesto l'utilizzo de' L'Algebrista in quanto gli alunni non disponevano del software in questione.

La prima consegna che riportiamo riguarda il test preliminare, gli esercizi che lo riguardano sono contrassegnati dalla sigla T.

Il test

T 1. Osserva le seguenti scritture, per ciascuna di esse spiega perché pensi sia corretta o perché pensi sia sbagliata:

- $-(6 - 1) + 3 = -6 + 1 + 3$
- $17 + (6 + 9) = (17 + 6) + 9$
- $6 + 2 \cdot (4 \cdot 5) = 6 + (2 \cdot 4) \cdot 5$
- $3 \cdot 11 + 6 - 6 = 3 \cdot 11$
- $15 + 6 \cdot 4 + 19 \cdot 4 + 11 = 15 + (6 + 19) \cdot 4 + 11$
- $17 = 10 + 7$
- $5 \cdot 2 + 7 = 7 + 7$
- $8 + 9 \cdot (3 + 2) - 17 = 8 + 27 + 18 - 17$
- $10 + 7 = 5 \cdot 2 + 7$
- $3 + 6 \cdot 73 + 6 \cdot 8 + 13 = 3 + 6 \cdot (73 + 8) + 13$

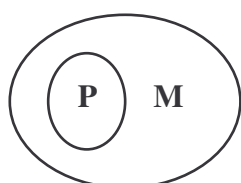
T 2. Scrivi cosa sai a proposito di ciascuno dei seguenti modi di dire, ad esempio puoi scrivere una frase che le contenga oppure puoi spiegare il loro significato. Puoi anche aiutarti con qualche esempio.

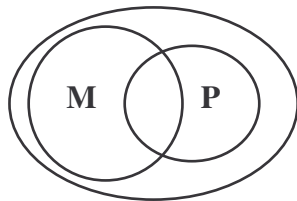
- | | | |
|---------------------------|---------------------------|---------------------------|
| 1. Svolgere | 9. Equazione | 17. Regola |
| 2. Risultato | 10. Espressione letterale | 18. Calcolo del valore di |
| 3. Espressione | 11. Prodotto notevole | un'espressione |
| 4. Potenza | 12. Operazione | 19. Formula |
| 5. Proprietà associativa | 13. Somma algebrica di | 20. Disuguaglianza |
| 6. Raccogliere | monomi | 21. Incognita |
| 7. Proprietà distributiva | 14. Monomio | 22. Uguale |
| 8. Proprietà commutativa | 15. Soluzione | 23. Polinomio |
| | 16. Proprietà | |

Quali dei precedenti modi di dire pensi che siano collegati tra loro? Perché?

T 3. Quali delle seguenti scritture possono corrispondere alla frase "*Nel cesto ci sono più mele che pere*" :

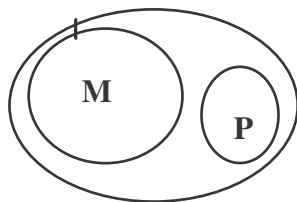
- $m + p$
- $m > p$
- mele + pere
- $p > m$
- mele > pere
- $p < m$
-





-
-

T 4. | Quali delle seguenti frasi possono corrispondere alla scrittura " $2m = p$ ":



- Per ogni pera ci sono due mele
- Per ogni mela ci sono due pere
- Le pere sono il doppio delle mele
- Le mele sono il doppio delle pere

25/11/98 e 26/11/98

Ricordiamo che in principio i ragazzi hanno a disposizione solo la paletta dei

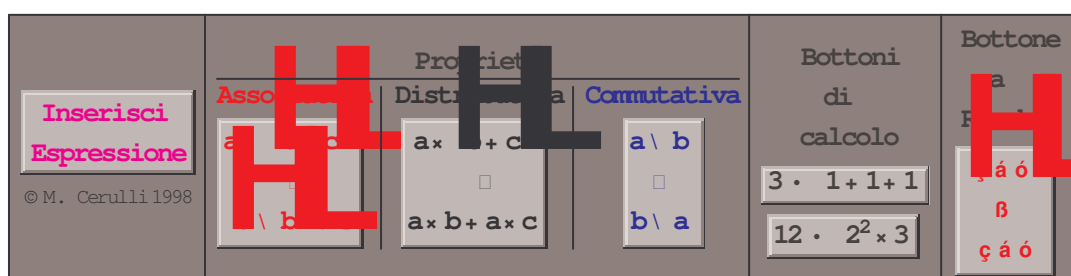


Fig. 1: la paletta iniziale

comandi rappresentati in figura 1.

Le consegne sono riguardanti la fase delle espressioni numeriche hanno lo scopo di introdurre le proprietà una ad una, e di consolidare il ruolo che esse giocano nella trasformazione di espressioni. Esse si dividono essenzialmente in due tipi:

- consegne in cui lo studente deve confrontare due o più espressioni e trasformare l'una nell'altra;
- consegne in cui lo studente deve analizzare un percorso dato di espressioni, ed individuare tramite quali trasformazioni o comandi de' L'Algebrista si può ricostruire il percorso. In genere è presente anche qualche passaggio scorretto generato con il bottone di calcolo. Con questo tipo di consegna, in particolare, cerchiamo di fornire allo studente un mezzo di controllo per giudicare il lavoro, abituandolo ad analizzare percorsi di espressioni non prodotti da lui.

Qualora un'attività abbia un'ulteriore scopo particolare, lo indicheremo sotto la descrizione dell'attività stessa.

CL 1. Inserire la seguente espressione³⁹: $(3+5)*7-3*(2-8)$

Cosa è successo alle differenze?

Calcolare il risultato tramite i bottoni di calcolo.

Scopo: introdurre il software agli studenti e farli riflettere sull'addizione e la sottrazione.

CL 2. Eseguire con i bottoni di calcolo: **7-5**.

Eseguire con i bottoni di calcolo: **5-7**.

Scopo: introdurre la proprietà commutativa e far osservare che non vale per la sottrazione, motivare quindi la scelta di usare solo addizioni.

CL 3. Alla lavagna si propone la scrittura $3*(3+5)$. Si chiede di calcolare e scrivere a parole il percorso di calcolo. Poi si chiede di pensare, per la stessa espressione, un percorso di calcolo diverso dal precedente, scriverlo a parole e verificare che il risultato sia lo stesso (esempio di due percorsi diversi:

$(3+5)*3 \vee 3*3+5*3 \vee 9+15 \vee 24$ e $(3+5)*3 \vee (8)*3 \vee 24$).

Discutere l'equivalenza dei due percorsi, scrivere le due espressioni corrispondenti, mettere un doppio uguale tra le due.

Scopo: introduzione di due relazioni di equivalenza tra espressioni: in base al risultato ed in base alla possibilità di trasformare un'espressione nell'altra tramite i bottoni delle proprietà. Introduzione dei termini *dimostrare* e *verificare* per indicare rispettivamente l'utilizzo delle proprietà o del calcolo per mostrare l'equivalenza tra due espressioni.

CL 4. Dimostrare la seguente equivalenza lavorando prima solo su un'espressione e poi solo sull'altra: **$3*4+3*7 == 3*(4+7)$**

Scopo: consolidare l'idea di equivalenza in termini di trasformazioni sia a destra che a sinistra. Introdurre la proprietà distributiva ed, in particolare, stabilirne la validità sia nel verso di raccogliere che di svolgere.

CS 1. Abbiamo trasformato due espressioni usando i bottoni che abbiamo a disposizione, sapresti indicare per ogni passaggio quali bottoni sono stati utilizzati⁴⁰ e su quale pezzo di espressione sono stati applicati? I passaggi sono tutti corretti?

³⁹ Consegna presentata a voce

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

$$2*3+2*5+2*(4+7)-(8+5)*2$$

$$\boxed{2*3+2*5+2*(4+7)+(-1)*(8+5)*2}$$

$$2*3+2*5+2*(4+7)+(-1)*(8+5)*2$$

$$2*3+2*5+2*(4+7)+(2*(8+5))*(-1)$$

$$2*3+2*(5+(4+7))+(2*(8+5))*(-1)$$

$$2*3+2*(5+(4+7))+(2*8+5)*(-1)$$

$$2*3+2*(5+(4+7))+((2*8)*(-1)+5*(-1))$$

$$2*3+2*(5+(4+7))+(2*8)*(-1)+5*(-1)$$

$$2*3+2*(5+4+7)+(2*8)*(-1)+5*(-1)$$

$$3*(2*5)+(2*3)*5-60$$

$$\boxed{3*(2*5)+(2*3)*5+(-60)}$$

$$3*(2*5)+(2*3)*5+(-60)$$

$$(3*2)*5+(2*3)*5+(-60)$$

$$(3*2+2*3)*5+(-60)$$

$$(6+6)*5+(-60)$$

$$(12)*5+(-60)$$

$$60+(-60)$$

$$0$$

Scopo: consolidare il ruolo giocato dalle proprietà; distinguere le situazioni di uso corretto o meno del bottone a rischio;

CS 2. Considera le due seguenti espressioni:

$$7-2+6(3+4)+5 \cdot 6$$

$$(3+4+5) \cdot 6+7-2$$

1) Come diventerebbero se tu le inserissi nel programma che usiamo in laboratorio?

2) Verifica che sono equivalenti calcolando il risultato di entrambe

3) Dimostra che sono equivalenti:

a) Trasformando la prima nella seconda spiegando ogni passaggio facendo riferimento alle proprietà delle operazioni oppure ai bottoni del programma.

b) Trasformando la seconda nella prima spiegando ogni passaggio facendo riferimento alle proprietà delle operazioni oppure ai bottoni del programma.

3/12/98

CL 5. Inserisci e confronta le due seguenti espressioni, se pensi che siano uguali dimostralò, altrimenti verifica che sono diverse:

$$4*(7+11)+8*4+(3+17)*4+4*(23+2) == (7+11+17+3+8+23+2)*4$$

⁴⁰ Nella consegna era presente anche un'immagine raffigurante i bottoni del programma, come quella di fig. 1.

Quindi rispondi alle seguenti domande:

- 1) Che procedimento hai seguito? In particolare quali bottoni hai utilizzato?
- 2) Sapresti spiegare il funzionamento, a parole tue, di ciascun bottone utilizzato?
- 3) A quali proprietà si riferiscono?
- 4) Quale bottone secondo te ha giocato il ruolo più importante?

nota: Se non ti crea difficoltà scrivi le tue risposte sul computer sotto ciascuna domanda, altrimenti scrivile su un foglio

CL 6. Inserisci e confronta le due seguenti espressioni, se pensi che siano uguali dimostrarlo, altrimenti verifica che sono diverse:

$$3(5*8)+8*11 == (3*5+11)*8$$

Quindi rispondi alle seguenti domande:

- 1) Che percorso hai seguito? In particolare quali bottoni hai utilizzato?
- 2) Se hai utilizzato bottoni nuovi, sapresti spiegarne il funzionamento?
- 3) A quali proprietà si riferiscono?
- 4) Quale bottone secondo te ha giocato il ruolo più importante?

nota: Se non ti crea difficoltà scrivi le tue risposte sul computer sotto ciascuna domanda, altrimenti scrivile su un foglio

Scopo: introduzione della proprietà associativa; consolidare il concetto di relazione di equivalenza; consolidare il ruolo giocato dalle proprietà commutativa e distributiva;

CS 3. Considera le tre seguenti espressioni:

$$(3*(-8) + (3*7)*6 - 7*6) + (7+3)*6$$

$$7+3*6+3*(-8) + 3*7*6 - 7*6$$

$$(-8+7*6)*3 + (7*6+6*3-7*6)$$

- 1) Come diventerebbero se tu le inserissi nel programma che usiamo in laboratorio?
- 2) Confronta le tre espressioni, sono tutte equivalenti? Se si dimostrarlo spiegando ogni passaggio facendo riferimento alle proprietà delle operazioni oppure ai bottoni del programma.
- 3) Che tipo di difficoltà hai trovato nello svolgere questo esercizio?

4) Sei sicuro di tutte le risposte che hai dato o c'è qualcosa che preferiresti riaffrontare in classe per chiarirla meglio? Se sì, a quali cose in particolare ti riferisci?

CS 4. Abbiamo trasformato due espressioni usando i bottoni che abbiamo a disposizione:

1) Indica, per ogni passaggio quali bottoni⁴¹ sono stati utilizzati e su quale pezzo di espressione sono stati applicati.

2) I passaggi sono tutti corretti? Motiva la tua risposta.

3) Che tipo di difficoltà hai trovato nello svolgere questo esercizio?

4) Sei sicuro di tutte le risposte che hai dato o c'è qualcosa che preferiresti riaffrontare in classe per chiarirla meglio? Se sì, a quali cose in particolare ti riferisci?

$$(4+2)-5+3\cdot(7\cdot2)+4\cdot5+2\cdot5$$

$$\boxed{(4+2)+(-5)+3\cdot(7\cdot2)+4\cdot5+2\cdot5}$$

$$(4+2)+(-5)+3\cdot(7\cdot2)+4\cdot5+2\cdot5$$

$$(4+2)+(-5)+(3\cdot7)\cdot2+4\cdot5+2\cdot5$$

$$(4+2)+(-5)+(3\cdot7)\cdot2+(4+2)\cdot5$$

$$(4+2)+(-5)+2\cdot(3\cdot7)+(4+2)\cdot5$$

$$(4+2)+(-5)+2\cdot(3\cdot7)+4+2\cdot5$$

$$4+(2+(-5))+2\cdot(3\cdot7)+4+2\cdot5$$

$$4+(2+(-5))+4+2\cdot(3\cdot7)+2\cdot5$$

$$4+2+((-5)+4)+2\cdot(3\cdot7)+2\cdot5$$

$$4+2+((-5)+4)+2\cdot(3\cdot7+5)$$

$$4+2+(-5)+4+2\cdot(3\cdot7+5)$$

21/12/98

Compito in classe

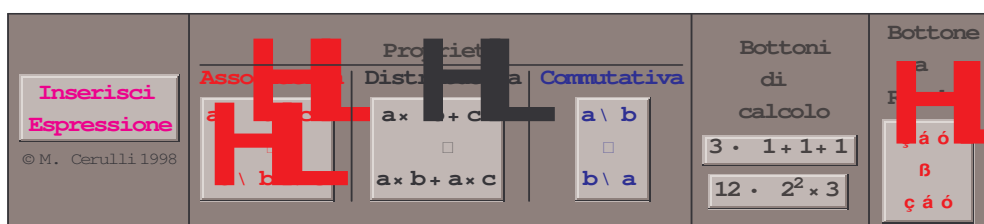
CL 7. Considera le tre seguenti espressioni:

$$8 \cdot (6+3) + 5 \cdot (3 \cdot 2) + 8 \cdot 8 - 6 \cdot 2$$

$$2 \cdot (5 \cdot 3 - 6) + 6 \cdot 8 + (8+3) \cdot 8$$

- 1) Come diventerebbero se tu le inserissi nel programma che usiamo in laboratorio?
- 2) Confronta le tre espressioni, sono tutte equivalenti? Se si dimostrarlo spiegando ogni passaggio facendo riferimento alle proprietà delle operazioni oppure ai bottoni del programma.

- 1) Indica, per ogni passaggio quali bottoni sono stati utilizzati e su quale pezzo di espressione sono stati applicati.
- 2) I passaggi sono tutti corretti? Motiva la tua risposta.



$$\boxed{(3+4)+5+3\cdot(7\cdot4)+6+(3+5)\cdot4}$$

101

13/1/99

Compiti per le vacanze di Natale

CS 5.

a) Dimostra le seguenti equivalenze avvalendoti della proprietà associativa della somma e la proprietà associativa del prodotto, specificando in quale caso usi l'una ed in quale caso usi l'altra.

$$(7 \cdot 2) \cdot 9 \cdot 15 == 7 \cdot 2 \cdot (9 \cdot 15)$$

$$(3+5) + (6+13) == 3 + ((6+5) + 13)$$

$$6 + (2 \cdot (4 \cdot 7) + 8) == (6 + (2 \cdot 4) \cdot 7) + 8$$

b) Dimostra le seguenti equivalenze avvalendoti della proprietà distributiva.

$$(4+7) \cdot 3 + 5 \cdot 3 == (4+7+5) \cdot 3 == 4 \cdot 3 + 7 \cdot 3 + 5 \cdot 3$$

$$3 \cdot 4 + 3 \cdot 5 + 3 \cdot 6 == 3 \cdot 4 + 3 \cdot (5+6) == 3 \cdot (4+5+6)$$

c) Dimostra la seguente equivalenza: $(3 \cdot 8) \cdot 5 + 6 \cdot 8 == 8 \cdot (3 \cdot 5 + 6)$

d) Considera le seguenti espressioni:

$$(4+3) \cdot (4+3) + 2 \cdot (4+3) \cdot 5$$

$$2 \cdot 3 \cdot 4 + 5 \cdot 3 + 4 \cdot (4+2 \cdot 5) + 3 \cdot 3$$

$$a+a+2 \cdot (4 \cdot 5) + 5 \cdot 2 \cdot 3$$

$$(4+3) \cdot (2 \cdot 5 + 4 + 3)$$

$$2 \cdot (3 \cdot 4 + 5 \cdot 3) + 4 \cdot (4+2 \cdot 5) + 3 \cdot 3$$

$$4 \cdot (2 \cdot 5 + 4 + 3) + 3 \cdot 2 \cdot 5 + 4 + 3$$

d.1) Come diventerebbero se tu le inserissi nel programma che usiamo in laboratorio?

d.2) Confronta le sei espressioni, sono tutte equivalenti? Se si dimostraloro spiegando ogni passaggio facendo riferimento alle proprietà delle operazioni oppure ai bottoni del programma.

d.3) Che tipo di difficoltà hai trovato nello svolgere questo esercizio?

d.4) Sei sicuro di tutte le risposte che hai dato o c'è qualcosa che preferiresti riaffrontare in classe per chiarirla meglio? Se sì, a quali cose in particolare ti riferisci?

Scopo: consolidare tutto il lavoro svolto fino a questo momento, in particolare, distinguere tra proprietà associativa e distributiva, e consolidare l'idea che la proprietà associativa valga anche per il prodotto.

CS 6.

Abbiamo trasformato un'espressione usando i bottoni che abbiamo a disposizione:

- 1) Indica, per ogni passaggio quali bottoni sono stati utilizzati e su quale pezzo di espressione sono stati applicati.
- 2) I passaggi sono tutti corretti? Motiva la tua risposta.
- 5) Che tipo di difficoltà hai trovato nello svolgere questo esercizio?
- 6) Sei sicuro di tutte le risposte che hai dato o c'è qualcosa che preferiresti riaffrontare in classe per chiarirla meglio? Se sì, a quali cose in particolare ti riferisci?

$$(5*6)*3+(4*2+7*3)+(6+5)*4$$

$$\boxed{(5 \cdot 6) \cdot 3 + (4 \cdot 2 + 7 \cdot 3) + (6 + 5) \cdot 4}$$

$$(5 \cdot 6) \cdot 3 + (4 \cdot 2 + 7 \cdot 3) + (6 + 5) \cdot 4$$

$$(5 \cdot 6) \cdot 3 + (7 \cdot 3 + 4 \cdot 2) + (6 + 5) \cdot 4$$

$$(5 \cdot (6 \cdot 3) + 7 \cdot 3) + 4 \cdot 2 + (6 + 5) \cdot 4$$

$$(5 \cdot (6 \cdot 3) + 7 \cdot 3) + 4 \cdot 2 + (6 \cdot 4 + 5 \cdot 4)$$

$$((5 \cdot 6) \cdot 3 + 7 \cdot 3) + 4 \cdot 2 + (6 \cdot 4 + 5 \cdot 4)$$

$$((5 \cdot 6 + 7) \cdot 3) + 4 \cdot 2 + (6 \cdot 4 + 5 \cdot 4)$$

$$(5 \cdot 6 + 7) \cdot 3 + 4 \cdot 2 + 6 \cdot 4 + 5 \cdot 4$$

$$(5 \cdot 6 + 7) \cdot 3 + 2 \cdot 4 + 6 \cdot 4 + 5 \cdot 4$$

$$(5 \cdot 6 + 7) \cdot 3 + (2 + (6 + 5)) \cdot 4$$

$$(5 \cdot 6 + 7) \cdot 3 + ((2 + 6) + 5) \cdot 4$$

$$(5 \cdot 6 + 7) \cdot 3 + (2 + 6 + 5) \cdot 4$$

$$((5 \cdot 6) \cdot 3 + 7 \cdot 3) + (2 + 6 + 5) \cdot 4$$

$$(5 \cdot 6 \cdot 3 + 7 \cdot 3) + (2 + 6 + 5) \cdot 4$$

$$(5 \cdot 6 \cdot 3 + 7 \cdot 3) + 2 + 6 + 5 \cdot 4$$

$$5 \cdot 6 \cdot 3 + 7 \cdot 3 + 2 + 6 + 5 \cdot 4$$

CS 7.

a) Quando un'espressione viene inserita nel programma che usiamo in laboratorio abbiamo visto che le sottrazioni vengono trasformate, come? Sapresti dire perché il programmatore ha scelto di trasformare le sottrazioni?

b) Perché in alcuni bottoni del programma le operazioni sono rappresentate con il simbolo "□" invece che con i normali simboli "+" per la somma e "." per il prodotto?

c) Cosa intendiamo quando diciamo che due espressioni sono equivalenti?

d) In classe abbiamo distinto due parole, *dimostrare* e *verificare*, sapresti spiegare il significato di ciascuna delle due?

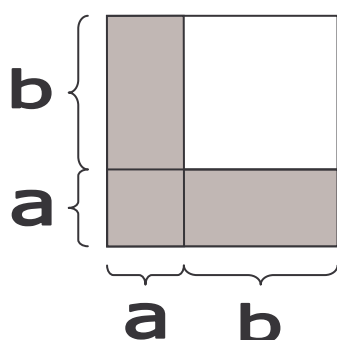
e) Fai uno schema che riassume le proprietà delle operazioni che abbiamo incontrato in classe scrivendo ciò che abbiamo imparato a proposito di ciascuna di loro.

f) Pensi che i bottoni del programma siano tutti quelli necessari o tu ne aggiungerei qualche altro? Se sì spiega quale e come dovrebbe funzionare.

Scopo: riflessione conclusiva sul lavoro svolto onde consolidare i concetti introdotti e preparare il campo per affrontare le espressioni letterali; abituare gli studenti a verbalizzare ciò che pensano.

23/1/99

CL 9.



Discutendo sulla figura qui a lato tre amici hanno trovato tre modi differenti di calcolare l'area della parte colorata di grigio:

Roberto propone di fare:

$$(a + b) * a + a * b$$

Valentina invece:

$$(a + b) * b - b * b + a * (a + b)$$

Marco infine propone:

$$b * b + (b + a) * a + (a - b) * b$$

- 1) Secondo te chi dei tre ha ragione?
- 2) Perché?

Scopo: introduzione delle espressioni letterali; introduzione della relazione di equivalenza sulle espressioni letterali in base all'area che rappresentano ed in base alla trasformabilità dell'una nell'altra; introduzione della relazione di equivalenza basata sulla sostituzione di numeri.

CS 8. Considera le seguenti espressioni:

$$a*a - b*b$$

$$(a - b)*(a + b) + 2*b*(a + b)$$

$$(a - b)*(a - b) + 2*(a - b)*b$$

- 1) Quali di esse pensi che siano equivalenti?
- 2) Perché? Sapresti dimostrarlo?
- 3) Analizza la dimostrazione che hai fatto nel punto precedente ed indica, per ogni passaggio fatto, se hai utilizzato un teorema o un assioma (proprietà).
- 4) Sapresti trovare una figura geometrica la cui area si possa calcolare con le espressioni sopra elencate?

Scopo: studio dei diversi tipi di relazione di equivalenza tra espressioni; introduzione delle parole *assioma* e *teorema* onde distinguere tra gli enunciati dati e gli enunciati dimostrati in classe dagli studenti; analisi delle trasformazioni eseguite onde scoprire passaggi dati per scontati ma non basati sulla teoria.

4/2/99

CL 10. Considera le seguenti espressioni:

$$m + n + (2 + m)*n + (3 + m)*m + 3*(m + n) - m*(m + n)$$

$$7*m + 6*n$$

$$6*(m + n) + m*(n + 1) - n$$

- 1) Quali di esse pensi che siano equivalenti?
- 2) Perché?

Scopo: consolidare l'idea di relazione di equivalenza tra espressioni.

6/2/99

CS 9. Considera le seguenti espressioni:

$$(m - n)^2$$

$$2 \cdot (m + n) \cdot n + m^2 - 2 \cdot n \cdot n$$

$$n \cdot m + (n^2 + m \cdot m) - 3 \cdot m \cdot n$$

$$m^2 - 2 \cdot m \cdot n + n^2$$

- 5) Quali di esse pensi che siano equivalenti?
- 6) Perché? Sapresti dimostrarlo?
- 7) Analizza la dimostrazione che hai fatto nel punto precedente ed indica, per ogni passaggio fatto, se hai utilizzato un teorema o un'assioma (proprietà).
- 8) Sapresti trovare dei numeri da sostituire ad **m** ed **n** nelle quattro espressioni in modo che diano tutte uno stesso risultato?
Quanti ne hai trovati?

Pensi che se ne possano trovare altri?

Scopo: approccio funzionale alle espressioni; introduzione dell'idea di teoria; consolidamento relazione di equivalenza; distinzione tra dimostrare e verificare.

13/2/99

CL 11. Dimostra che $13 \cdot m + m \cdot 17 == 30 \cdot m$

Scopo: ufficializzare la "dimostrazione" della somma di termini simili ed introdurre un bottone che esegua tale somma.

CL 12. Considera le seguenti espressioni:

$$(h - k) \cdot (k + h) + (c - d) \cdot (2 \cdot d - 2 \cdot c) - 8 \cdot c \cdot d$$

$$8 \cdot c + (h - k) \cdot (k + h) - 5 \cdot d$$

$$h \cdot h - (c + d) \cdot (2 \cdot d + 2 \cdot c) - k \cdot k$$

$$h \cdot h - k \cdot k$$

9) Quali di esse pensi che siano equivalenti?

10) Perché?

11) Se sostituisce dei numeri alle lettere **c** e **d**, ottieni delle nuove espressioni che contengono solo le lettere **h** e **k**. Sapresti trovare dei numeri da sostituire a **c** e **d** in maniera da far diventare uguali due delle espressioni che prima erano diverse?

Scopo: utilizzo dei primi *teoremi* (bottoni) dimostrati (cancellazione di opposti e somma di termini simili); approccio funzionale alle espressioni; differenziare la verifica numerica dell'equivalenza o meno di espressioni dalla dimostrazione tramite l'uso della teoria.

CS 10.

1. Dimostra che:

a. $a - a == 0$

b. $b \cdot 5 - 6 \cdot b + b == 0$

c. $7 \cdot m + 2$ non è equivalente a $3 + m \cdot 2$

d. $x^2 - y^2 == (x - y) \cdot (x + y)$

e. $(3 \cdot a + m \cdot 2)$ non è equivalente a $2^2 \cdot (a^2 + m^2 + 3 \cdot m \cdot a)$

2. Analizza ciascuna delle dimostrazioni che hai fatto indicando, per ogni passaggio, se hai utilizzato un teorema o un assioma (proprietà).

3. Considera le due espressioni indicate nel punto 1.c.: sapresti trovare dei numeri da sostituire ad **m** affinché le due espressioni diventino equivalenti?

4. Considera le due espressioni indicate nel punto 1.e. sapresti trovare dei numeri da sostituire ad **a** affinché le due espressioni diventino equivalenti?

5. Inventa due espressioni letterarie che non siano equivalenti ma che possano diventarlo sostituendo dei numeri particolari alle lettere.

Scopo: approccio funzionale alle espressioni; costruzione idea di teoria; consolidamento relazione di equivalenza; distinzione tra dimostrare e verificare; ricerca di passaggi scontati.

20/2/99

CL 13. Dimostra che $(a - b) \cdot (a + b) = a^2 - b^2$

Scopo: introdurre un nuovo teorema ed il relativo bottone tramite il Teorematore

CL 14. Considera le seguenti equivalenze, dimostrale se pensi che siano vere, altrimenti dimostra che sono false.

- I. $a \cdot a = a^2$
- II. $(-b) \cdot a = -(a \cdot b)$
- III. $(-b) \cdot b = (-1) \cdot b^2$
- IV. $-(b \cdot b) = (b \cdot b) \cdot (-1)$
- V. $(-b) \cdot (-b) = b^2$
- VI. $a \cdot (-b) + b \cdot a = a \cdot 0 \cdot a$
- VII. $a \cdot (-b) + b \cdot c = a \cdot 0 \cdot c$

Scopo: riflessione su alcuni errori fatti nei compiti per casa.

CL 15. Qualcuno di voi ha scritto che per dimostrare che due espressioni erano equivalenti (o non equivalenti), le ha "*calcolate algebricamente*". Secondo te cosa intendeva dire il compagno, o la compagna, in questione?

Scopo: verificare cosa i ragazzi intendono con "calcolare algebricamente", una frase che noi non abbiamo mai introdotto ma che loro usano spesso.

CS 11. Considera il seguente dialogo:

Valentina: "*Per dimostrare che due espressioni letterarie sono uguali (o diverse), basta sostituire dei numeri alle lettere.*"

Marco: "*Io invece credo che sia sufficiente trasformarle l'una nell'altra utilizzando gli assiomi ed i teoremi che si hanno a disposizione.*"

Luigi: "*Secondo me nessuno di voi due ha ragione, infatti se io non riesco a trasformare un'espressione nell'altra, come faccio a sapere che non ci riesco perché sono veramente diverse o perché invece io non ne sono capace? Inoltre, se sostituendo dei numeri alle lettere le due espressioni*

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

danno lo stesso risultato, come faccio ad essere sicuro che non esistono numeri per cui il risultato viene diverso?".

Tu invece come la pensi? Aiutati, eventualmente, portando degli esempi.

Scopo: riflessione finale sulla distinzione tra verificare e dimostrare l'equivalenza o meno di due espressioni.

27/2/99

CL 16. Congettura una formula per l'espressione $(a+b)^3$ e crea un bottone che la rappresenti. Sei sicuro/a che sia corretta? Perché?

Scopo: stimolare la creazione di congetture e nuovi teoremi da utilizzare in seguito.

CL 17.

I. Dimostra che $2 \cdot a + 2$ è divisibile per 2.

II. Dimostra che $3 \cdot a + 7 - 4$ è divisibile per $(a+1)$

III. Dimostra che $a^2 - 4$ è divisibile per $(a+2)$. Per cos'altro è divisibile?

IV. Dimostra che $n \cdot (m + 2) \cdot 3 \cdot (4 + n)$ e $4 \cdot n + n^2$ hanno dei fattori in comune.

Scopo: introduzione del concetto di divisibilità e di fattori di un'espressione;

CL 18. Dimostra l'equivalenza $(a + b)^3 == (a + b) \cdot (a^2 + b^2 + 2 \cdot a \cdot b)$. Puoi farlo riducendo il numero di passaggi?

Scopo: applicazione di teoremi dimostrati dagli studenti, in particolare, utilizzo dei bottoni che li rappresentano; motivazione dello studio dei prodotti notevoli.

CL 19. Considera l'espressione $(a - b)^2$, congettura una formula analoga a quella trovata per $(a + b)^2$ e confronta i due bottoni creati. Pensi che siano necessarie entrambe le formule o pensi che una possa bastare a svolgere anche i compiti dell'altra? E per quanto riguarda i bottoni?

Scopo: distinguere tra prodotti notevoli "necessari" e non.

CS 12. Congettura una formula per l'espressione $(a + b)^4$. Sei sicuro/a che sia corretta? Perché?

Scopo: stimolare la creazione di congetture e nuovi teoremi da utilizzare in seguito; indirizzare verso la generalizzazione di teoremi particolari già noti.

CS 13.

I. **Dimostra che** $b^2 + 4 \cdot c \cdot b + 2 \cdot (2 \cdot c^2)$ **è divisibile per** $(b + 2 \cdot c)$

II. Dimostra che $a^2 + b^2 + 2 \cdot a \cdot b$ e $a^2 - b^2$ hanno dei fattori in comune

Scopo: riflessione sul concetto di divisibilità; costruzione del concetto di scomposizione in fattori.

CS 14. Dimostra l'equivalenza $(a + b)^2 \cdot (a - b) == (a^2 - b^2) \cdot (a + b)$. Puoi farlo riducendo il numero di passaggi?

Scopo: utilizzo dei teoremi dimostrati dai ragazzi per motivare l'esistenza degli stessi, in particolare, dei prodotti notevoli.

CS 15.

I. Costruisci un'espressione letteraria, usando la lettera **a** fatta in modo che se sostituisci il numero **2** alla lettera **a**, il risultato dell'espressione numerica ottenuta è **0**. (*se vuoi sei libero/a di usare anche altre lettere*)

II. Prova a sostituire altri numeri alla lettera **a**, ne puoi trovare altri in modo che il risultato sia **0**?

III. Prova ora a costruire un'espressione, sempre usando la lettera **a**, in modo che se sostituisci **4** oppure **5** ad **a**, allora il risultato viene **0**.

Scopo: approccio funzionale alle espressioni; consolidamento della relazione di equivalenza tra espressioni; motivare la trasformazione di espressioni in forme diverse a seconda dell'obiettivo.

CS 16.

I. Dimostra che $a^2 \cdot a^3 == a^5$

II. Dimostra che $a^b \cdot a^c == a^{(b+c)}$

III. Dimostra che $a^{(3 \cdot 2)} == (a^3)^2$

IV. Dimostra che $a^{(b \cdot c)} == (a^b)^c$

Scopo: stimolare la generalizzazione di una dimostrazione.

34/3/99

CL 20. Considera il numero $2 \cdot 3 \cdot 4 \cdot 6 \cdot 7 \cdot 9 \cdot 10 \cdot 13 \cdot 17 \cdot 27 \cdot 31$

- Sapresti dire, senza calcolarlo, se è divisibile per **2**? E per **19**?
- Sapresti dire, senza calcolarlo, se è multiplo di **3**? E di **8**? E di **13**•**7**?
- Per quali altri numeri è divisibile?
- Di quali altri numeri è multiplo?
- Sapresti esprimere con una formula il fatto che un numero qualsiasi è divisibile per **31**?
- Sapresti esprimere con una formula il fatto che un numero qualsiasi è multiplo di **31**?
- Sapresti costruire un numero che sia divisibile per 23 ma non per 3?
- Sapresti costruire un numero che sia multiplo di 36 ma non di 4?
- Costruisci un'espressione letterale tale che, qualunque numero tu sostituisca alle lettere il risultato sia dispari.

Scopo: introdurre gli allievi al processo di nominalizzazione attraverso problemi di divisibilità

CL 21. Considera l'espressione $2 \cdot (a+3) \cdot 6 \cdot (b-1)$

- Per quali numeri è divisibile? Spiega bene il tuo ragionamento.

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

- b. Di quali espressioni è multipla? Spiega bene il tuo ragionamento.
- c. Costruisci un'espressione che sia divisibile per **2**, ma non per **3**
- d. Costruisci un'espressione che sia multipla di **a+1**
- e. Costruisci due espressioni che abbiano **3** fattori in comune ma che siano diverse,
- f. Costruisci un'espressione letterale che sia divisibile per **3** e per **(a - b)**; quante se ne possono costruire?

Scopo: introduzione alla scomposizione in fattori, nel caso di espressioni letterali, tenuto conto che il concetto di scomposizione in fattori è tutt'altro che banale, sia per i numeri che, a maggior ragione, per le espressioni.

CS 17. Il gioco della verità: Luigi, Marco, Roberto e Valentina leggono su un libro l'espressione: $(2 \bullet a + 4 \bullet b) \bullet (a - 2 \bullet b + a - 2 \bullet b)$.

il professore chiede loro cosa possono sapere di quell'espressione; i ragazzi rispondono:

Marco: secondo me è uguale all'espressione $2 \bullet (a + 2 \bullet b) \bullet (2 \bullet a - 4 \bullet b)$

Valentina: secondo me ti sbagli, infatti non è divisibile per **2** ma è divisibile per **4**

Roberto: io penso che sia uguale a $4 \bullet a^2 + 4 \bullet b^2$

Luigi: io invece credo che sia uguale a $(2 \bullet a)^2 - (4 \bullet b)^2$

Tu cosa diresti?

Scopo: motivare l'attività di trasformazioni in ambiti diversi dal "gioco con il computer"; favorire il passaggio tra diverse strategie di soluzione.

6/3/99

CL 22. Durante l'esercizio fatto in classe giovedì scorso, alcuni di voi hanno scritto le seguenti espressioni che diano come risultato un numero dispari, qualsiasi numero si sostituisca alle lettere:

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

a. $8 \cdot a + 1$

b. $y^0 \cdot x^0$

c. x/x

Quali numeri dispari è possibile ottenere?

È possibile ottenere il numero **5** con una di queste espressioni?

Scrivi un'espressione con la quale è possibile ottenere tutti i numeri dispari.

Scrivi un'espressione analoga per i numeri pari.

Scopo: riflessione sul lavoro svolto dai ragazzi nella lezione precedente; stimolare la produzione autonoma di formule.

CL 23. Per ciascuna delle seguenti affermazioni spiega perché pensi che sia vera oppure che sia falsa.

a. Le due espressioni $9 \cdot 3 \cdot 5 + 7^2$ e $9 \cdot 3 \cdot 5 - 5^3$ sono diverse, ma hanno 3 fattori in comune

b. $(a - b) \cdot 3 \cdot x + 7$ è multipla sia di $(a - b)$ che di **3**

c. Le due espressioni seguenti sono diverse ma hanno 3 fattori in comune

$$(3 \cdot a - 4) \cdot b^2 \cdot 5 \cdot c \cdot (a^2 - a) \cdot 4 \cdot a \cdot 3 \cdot b \cdot 2 \cdot c$$

$$(4 \cdot a - 3) \cdot b^2 \cdot 5 \cdot c \cdot (a^2 - a) \cdot 2 \cdot b \cdot 4 \cdot c \cdot 5 \cdot a$$

d. Le due espressioni seguenti sono diverse ma hanno 3 fattori in comune

$$a \cdot b \cdot c^2 + b^2$$

$$a^2 \cdot b + c + 3 \cdot a \cdot b$$

e. La seguente espressione è divisibile sia per **3** che per $(a - b)$

$$3 \cdot (x + y + z) + a \cdot b \cdot (a - b)$$

Scopo: preparare il terreno per introdurre la scomposizione in fattori di un'espressione.

CL 24. Fattorizza le due seguenti espressioni (sugg. cerca quali tutti i divisori delle due espressioni):

$$a^2 - b^2 + a - b$$

$$b^2 + a + 2 \cdot a \cdot b + b + a^2$$

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

Scopo: la discussione delle soluzioni dovrebbe portare alla definizione di scomposizione in fattori di un'espressione.

CS 18. considera l'espressione $a + b + (3 \cdot a - 3 \cdot b) \cdot 2 + 3 \cdot b$, secondo te in quale espressione equivalente è più conveniente trasformarla nel caso che:

- Devi confrontarla con un'altra espressione.
- Devi dimostrare che sia divisibile per 2 e non per 3.
- Devi dimostrare che sia divisibile per $a + b$.
- Devi sostituire dei numeri ad a e b per calcolarla e vuoi che il calcolo, per te, sia più facile possibile.

Scopo: motivazione della trasformazione di espressioni e della scomposizione in fattori.

CS 19. Confronta le tre espressioni seguenti

$$(m + 2) \cdot (m - 3) \cdot m \cdot (m - 4) \cdot (m - 5) (2 - m - 2 \cdot m^2 + m^3)$$

$$(m - 1) \cdot (m + 1) \cdot (m - 2) \cdot (34 \cdot m^2 - 120 \cdot m + 23 \cdot m^3 - 10 \cdot m^4 + m^5) (m - 6)$$

$$(m + 2) \cdot (m - 3) \cdot m \cdot (m - 4) \cdot (m - 5) \cdot (m - 1) \cdot (m + 1) \cdot (m - 2)$$

- Sono equivalenti o diverse? Perché?
- Fattorizza le tre espressioni
- Quanti fattori hanno in comune?
- Quali numeri puoi sostituire alla lettera m affinché il risultato sia 0.

Scopo: riflessione su tutto il lavoro svolto sulle espressioni letterali.

Altre proposte di attività

In questa sezione riportiamo alcune proposte di attività da svolgere con L'Algebrista, o comunque collocabili in un percorso didattico basato su principi simili a quelli su cui si è fondata la nostra sperimentazione.

E.1.Data un'espressione qualsiasi, gioca con i bottoni dei comandi, tenendo traccia di ciò che fai e cercando di capire la funzione di ciascun comando. Descrivi la funzione di ciascun comando. Confronta le tue conclusioni con quelle dei tuoi compagni.

E.2.Una volta che si ha confidenza con i comandi del prototipo si possono introdurre quelli standard di *Mathematica* e chiedere agli allievi di scoprire cosa fa ciascuno di questi comandi, descrivendone il funzionamento in termini dei comandi del prototipo e/o delle proprietà fondamentali delle operazioni.

E.3.*Il Bottone Misterioso*: si presenta agli alunni un bottone senza alcuna indicazione della sua funzione. Agli alunni é quindi richiesto di *scoprire* cosa fa il bottone e di trovare un'espressione adeguata da scrivere sul bottone e che ne rappresenti la funzione. Ci possono essere due modalità per questo tipo di esercizio:

- supponiamo di avere introdotto agli alunni solo alcuni dei bottoni previsti, allora gli presentiamo uno degli altri come bottone misterioso;
- supponiamo invece di avere introdotto tutti i bottoni previsti, allora possiamo introdurre alcuni bottoni supplementari, inventati e costruiti appositamente, ad esempio:
 - $a + b \text{ } \downarrow \text{ } b + a$
 - $a \cdot b \text{ } \downarrow \text{ } b \cdot a$
 - $a \cdot (b + c) \text{ } \square \text{ } a \cdot b + a \cdot c$
 - $a \cdot b + a \cdot c \text{ } \square \text{ } a \cdot (b + c)$
 - $(a + b)^2 \text{ } \downarrow \text{ } a^2 + 2 \cdot a \cdot b + b^2$
 - Altri bottoni che siano versioni ridotte (cioè che funzionino solo in un senso o che valgano per una sola un'operazione) di bottoni preesistenti.
 - Altri bottoni che rappresentino prodotti notevoli o altri eventuali teoremi.

E.4.*Il bottone sbagliato*: agli alunni viene presentato un bottone che produca delle scritture errate, e viene chiesto di scoprire dove sta l'errore e di discutere su come correggerlo. Notiamo che possiamo scegliere di scrivere sul bottone la sua vera funzione, oppure una simile ma corretta giusta, oppure niente; insomma possiamo creare delle discrepanze tra bottone e scrittura che lo rappresenta. Possibili esempi di bottoni sbagliati:

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

- $a \cdot (b+c) \cdot d \not\equiv a \cdot b + a \cdot c \cdot d$
- $a \cdot b + c \not\equiv a \cdot c + b$
- Un bottone qualsiasi funzionante bene ma rappresentato da una generalizzazione errata (in questo caso si può chiedere agli alunni di prevedere il comportamento del bottone e poi di verificarlo in pratica).
- Un bottone mal funzionante ma rappresentato da una scrittura teoricamente corretta.

E.5.Ci sono alcuni bottoni che non sono invertibili direttamente, poiché non sono “iniettivi”, chiedere agli alunni di trovare il modo di invertire tali bottoni usando anche tutti gli altri bottoni. In particolare, qualora non fosse possibile invertire il bottone in questione con i comandi disponibile, si incoraggiano gli alunni a proporre la creazione di uno o più bottoni da utilizzare per tale scopo.

E.6.Dimostrare che $a+(b+c)=(c+a)+b$ (usando solo le proprietà commutativa ed associativa).

E.7.Dimostrare che $(a+b)+(c+d)=(a+c)+(b+d)$ (come sopra)

E.8.Dimostrare che $-(a+b)=(-a)+(-b)$ (come sopra)

E.9.Fattorizzare $b \cdot c(b+c)+c \cdot a(c-a)-a \cdot b(a+b)$ (sugg. sviluppa e ricomponi)

E.10.Fattorizzare $(x \cdot u - y \cdot v)^2 + (y \cdot u + x \cdot v)^2$

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

Confronto de' L'Algebrista con Milo, Theorist e Derive

Riportiamo delle tabelle che riassumono il confronto tra i quattro softwares. Le prime tabelle che presentiamo sono ricavate da un confronto di Milo, Theorist e Derive, redatto da Auricchio – Dettori - Lemut, 1996.

Caratteristiche Di utilizzo	DERIVE	MILO	THEORIST	L'ALGEBRISTA
Modo di utilizzo del video	Spazio di lavoro e spazio di rappresentazione separati.	Si lavora direttamente nello spazio di rappresentazione.	Si lavora direttamente nello spazio di rappresentazione.	Schermo diviso in palette, solo cliccabili, e notebook, editabili. Il lavoro si articola tra questi due ambienti. All'interno dei notebook si crea un'ulteriore spazio di lavoro e/o rappresentazione.
Sequenza di rappresentazioni e sul video	Le espressioni che compaiono sullo schermo sono numerate in sequenza temporale (quindi può venire persa la sequenza logica). Globalmente non è possibile distinguere le espressioni inserite dall'utente da quelle ottenute per manipolazione (informazioni fornite di volta in volta solo per la riga selezionata.	Manca ogni indicazione di sequenza temporale o logica; una nuova espressione è inserita dove l'utente ha posizionato il cursore. Le manipolazioni sostituiscono le espressioni anziché riscriverle; l'espressione originale occorre duplicarla prima di manipolarla.	Espressioni ordinate in sequenza logica, non numerate. Un quadrato indica espressione introdotta dall'utente, un triangolo espressione ottenuta manipolando; l'indentazione indica a quale espressione è stata applicata la manipolazione.	L'utente può inserire un'espressione in un punto qualsiasi del notebook (che è anche un'aria editabile). Per poter agire usando i comandi a disposizione l'espressione deve essere inserita in un sotto ambiente del notebook all'interno del quale, se l'utente agisce sempre sull'ultima espressione ottenuta, abbiamo una sequenza temporale e logica delle espressioni ogni nuova espressione viene scritta immediatamente sotto a quella da cui proviene e si porta con se informazioni su come è stata ottenuta. L'espressione iniziale è ben distinta dalle altre. È possibile tornare ad agire su espressioni intermedie, questo però causa la perdita della successione temporale, però rimane traccia della successione logica. È auspicabile l'introduzione di un controllo temporale
Inserimento di un'espressione	È necessario introdurre parentesi nella scrittura; si scrive su una sola riga nello spazio di lavoro; viene visualizzato su più righe nello spazio di rappresentazione.	Si scrive direttamente sul foglio di lavoro. Il cursore rimane accanto all'ultimo carattere scritto e va spostato come richiesto con mouse o tastiera.	Si scrive direttamente sul foglio di lavoro; due opzioni per la posizione del cursore; si scrive da tastiera, o selezionando nella barra superiore.	Si scrive direttamente sul notebook, un foglio di lavoro che condensa in se le proprietà standard di un qualsiasi editore e di un ambiente di programmazione. L'espressione può essere scritta nella forma più consona all'utente; deve quindi essere selezionata ed inserita in un sottoambiente matematico cliccando un apposito bottone. L'espressione scritta dall'utente rimane nel notebook, ma ne viene creata una nuova copia che appare scritta secondo una notazione standard decisa dal software, e basata su precise scelte didattiche. Su questa nuova copia l'utente può agire con i bottoni a disposizione.
Selezione di un'espressione	Con le frecce, tra più termini allo stesso livello, si può selezionarne uno solo o tutti; per selezionare sottoespressioni occorre racchiuderle in parentesi riscrivendo tutta l'espressione.	Con la barra spaziatrice o col mouse; selezione di più termini allo stesso livello solo col mouse; da tastiera, stesso problema di DERIVE.	Solo col mouse; nessun problema per selezionare sottoespressioni.	Godendo il notebook delle caratteristiche standard di un normale editore di testo è possibile selezionare le espressioni sia tramite la tastiera che tramite il mouse, usando procedure standard. Quando l'espressione in questione si trova nel sotto ambiente matematico, c'è una fondamentale restrizione sulle possibilità di selezione: è possibile selezionare solo pezzi di espressione che corrispondano a termini dal punto di vista matematico, ad esempio non è possibile una selezione del tipo a+b*c .

Caratteristiche di utilizzo	DERIVE	MILO	THEORIST	L'ALGEBRISTA	
Modifica di un'espressione:	Occorre riscrivere	Puntare il cursore nel punto di inserimento e digitare.	Puntare il cursore nel punto di inserimento e digitare.	Si fa come per un editore di testo standard.	All'interno del notebook qualsiasi modifica dell'espressione ha senso ed è possibile sostanzialmente tramite procedure tipiche di editori standard e tramite procedure tipiche dei manipolatori simbolici, messe a disposizione da <i>Mathematica</i> il software all'interno del quale si muove L'Algebrista.
<ul style="list-style-type: none"> • Aggiungere o togliere parti • Aggiungere parentesi 	Occorre riscrivere	Evidenziare la parte da racchiudere e battere una parentesi aperta; la seconda viene aggiunta automaticamente.	Evidenziare la parte da racchiudere e battere una parentesi aperta; la seconda viene aggiunta automaticamente.	Se si è nell'ambiente <i>matematico</i> basta, evidenziare la parte da racchiudere e cliccare il bottone apposito. Altrimenti si usa la prassi standard degli editori.	Nelle area del notebook generata da L'Algebrista, alla quale ci riferiamo chiamandola <i>ambiente matematico</i> , non è possibile <i>modificare</i> un'espressione, ma <i>trasformarla</i> coerentemente con gli assiomi dell'algebra. Questo si può fare tramite comandi base, e tramite nuovi comandi aggiunti dall'utente. La distinzione tra <i>modificare</i> e <i>trasformare</i> in L'Algebrista è ben rappresentata dai distinti ambienti di lavoro (per altro contenuti in un unico notebook, un unico foglio) di tipo <i>matematico</i> e di tipo <i>editore di testo</i> .
<ul style="list-style-type: none"> • Commutare termini in un'espressione 	Occorre riscrivere	Move Term Left o Move Term Right (muovono il termine o gruppo di termini selezionato di un posto rispettivamente a sinistra o a destra).	Commute (inverte l'ordine di tutto un gruppo di termini o fattori evidenziato).	In ambiente <i>matematico</i> si seleziona la coppia di termini in questione e si clicca sul bottone relativo alla proprietà commutativa. Se la testa dell'espressione selezionata è l'addizione commuta rispetto a questa operazione, se la testa è il prodotto commuta rispetto a questa. Se si selezionano più di due termini verrà prodotta una delle tante permutazioni possibili.	L'Algebrista, alla quale ci riferiamo chiamandola <i>ambiente matematico</i> , non è possibile <i>modificare</i> un'espressione, ma <i>trasformarla</i> coerentemente con gli assiomi dell'algebra. Questo si può fare tramite comandi base, e tramite nuovi comandi aggiunti dall'utente. La distinzione tra <i>modificare</i> e <i>trasformare</i> in L'Algebrista è ben rappresentata dai distinti ambienti di lavoro (per altro contenuti in un unico notebook, un unico foglio) di tipo <i>matematico</i> e di tipo <i>editore di testo</i> .
<ul style="list-style-type: none"> • Ordinare un polinomio secondo potenze crescenti o decrescenti 	Simplify (!) (in ordine crescente)	Order Sum (in ordine decrescente). Shift + Order Sum (in ordine crescente)	Simplify (!) (In ordine crescente)	Non è previsto nei comandi di base, ma si può fare utilizzando i comandi messi a disposizione da <i>Mathematica</i> .	L'Algebrista, alla quale ci riferiamo chiamandola <i>ambiente matematico</i> , non è possibile <i>modificare</i> un'espressione, ma <i>trasformarla</i> coerentemente con gli assiomi dell'algebra. Questo si può fare tramite comandi base, e tramite nuovi comandi aggiunti dall'utente. La distinzione tra <i>modificare</i> e <i>trasformare</i> in L'Algebrista è ben rappresentata dai distinti ambienti di lavoro (per altro contenuti in un unico notebook, un unico foglio) di tipo <i>matematico</i> e di tipo <i>editore di testo</i> .
Possibilità di mescolare testo e formule in un foglio di lavoro	Per righe singole	Si	Si		L'ambiente <i>editore di testo</i> e l'ambiente <i>matematico</i> sono immersi nello stesso notebook, anzi, direi che l'ambiente <i>matematico</i> è immerso in quello editore di testo. Questo implica una libertà quasi assoluta di mescolare espressioni algebriche e righe di testo. Con la sola condizione che all'interno dell'ambiente <i>matematico</i> il testo sia inserito sopra o sotto un'espressione, non accanto.

Possibilità di definire nuove regole	No	Si	Si	Questo è reso possibile dal <i>Teorematore</i> un notebook che consente di creare dei bottoni che rappresentino teoremi e che, se cliccati, applicano tali teoremi alle espressioni selezionate. In questo L'Algebrista di distingue dagli altri tre softwares in quanto permette di definire delle vere e proprie relazioni di equivalenza, ovvero bottoni che funzionino in due versi.
--------------------------------------	----	----	----	--

Operazioni Algebriche	DERIVE	MILO	THEORIST	L'ALGEBRISTA
Semplificazione di somme prodotti e frazioni	Si evidenzia l'espressione voluta e si attiva Simplify .	Si evidenzia l'espressione voluta e si attiva Simplify .	Si evidenzia l'espressione voluta e si attiva Simplify .	Queste sono in effetti operazioni che nell'ambito de' L'Algebrista non hanno senso in quanto troppo complesse ed oscure ed in quanto unidirezionali. Più precisamente in L'Algebrista è possibile fare delle <i>trasformazioni</i> invece che <i>operazioni</i> , ovvero trasformare espressioni (inclusi i numeri) in espressioni (numeri) equivalenti, ovvero dato un oggetto si cambia il simbolo che lo rappresenta.
Somma di frazioni	Factor(!) (tutte le opzioni) (Factor include un Simplify , quindi si perdono i passaggi intermedi) Simplify .	Add Fractions .	Collect (con fattorizzazione arbitraria del risultato); Factor (non fattorizza il risultato!)	
Moltiplicazione tra polinomi	Expand (include un Simplify , quindi salta i passaggi intermedi).	Multiply Out . Distribute (applica la proprietà distributiva solo a uno dei fattori).	Expand . MiniExpand (esegue solo il primo livello di calcoli).	
Sostituzione di una lettera con un'espressione o un valore numerico	Manage Substitute .	Substitution + Eval Substitutions , applicati in sequenza.	Substitute (complicata da usare)	Basta usare il normale comando di sostituzione che è conforme agli standard degli editori di testo.
Risoluzione di equazioni • Applicazione principi di equivalenza	Esegui in sequenza: 1. Build (racchiude l'equazione tra parentesi e scrive a fianco l'operazione richiesta). 2. Simplify .	Esegui in sequenza: 1. Battere l'operazione richiesta; 2. Distribute Over Equality ; 3. Simplify .	Esegui in sequenza: 1. Apply (rimangono selezionati i due membri, ma non l'=); 2. Battere l'operazione richiesta 3. Simplify	Non è attualmente implementato niente in proposito. Ma prevediamo di implementarlo come al solito basandoci su operazioni elementari

Operazioni Algebriche	DERIVE	MILO	THEORIST	L'ALGEBRISTA
Fattorizzazione: • Raccoglimento totale	Factor Trivial. Simplify (raccoglie solo coefficienti numerici)	Factor (raccoglie la variabile richiesta su tutti i termini selezionati, + <i>Option</i> solo nei termini che la contengono)..	Factor (raccoglie la variabile richiesta su tutti i termini selezionati).	Anche in questo caso L'Algebrista non ha nessuno strumento, di base, specifico per quanto riguarda la fattorizzazione e/o il viceversa. A disposizione dell'utente ci sono sostanzialmente le proprietà fondamentali delle operazioni, le quali del resto permettono di fare ciascuna delle operazioni algebriche in questione, in quanto assiomi su cui si basa l'algebra delle espressioni. Osserviamo però che qualora l'utente voglia compattare i passaggi necessari per effettuare una certa trasformazione, può avvalersi del <i>Teorematore</i> . Esso permette, ad esempio, di aggregare dei bottoni che rappresentino i prodotti notevoli e che permettano di avvalersi di tali uguaglianze in entrambe i versi.
• Raccoglimento parziale	Factor Trivial (occorre riscrivere l'espressione con opportune parentesi)	Factor (basta selezionare col mouse la sottoespressione che interessa; v. sopra per uso di <i>Option</i>).	Factor. Collect.	Osserviamo infine che la responsabilità del riconoscimento dei prodotti notevoli viene scaricata sull'utente il quale deve farsi carico della decisione di applicare un certo bottone su un certo pezzo di espressione. Qualora nell'espressione selezionata sia presente il prodotto notevole in questione, allora il bottone lo trasforma, altrimenti no, in questo modo l'utente ha un feed back sulla correttezza o meno della sua congettura.
• Riconoscimento quadrati	Factor Squarefree (se parziale, occorre riscrivere l'equazione).	Non li riconosce automaticamente; occorre definire regole ad hoc. Si possono però definire solo regole unidirezionali, e solo una alla volta, se ne definiamo una nuova, la vecchia va perduta.	Factor	
• Riconoscimento prodotti notevoli	Factor Rational (solo coefficienti razionali). Factor Radical (anche coefficienti reali). Factor Complex (anche coefficienti immaginari).	Non li riconosce automaticamente; occorre definire regole ad hoc. Si possono però definire solo regole unidirezionali, e solo una alla volta, se ne definiamo una nuova, la vecchia va perduta.	Non li riconosce automaticamente; occorre definire regole ad hoc. Si possono però definire solo regole unidirezionali, e solo una alla volta, se ne definiamo una nuova, la vecchia va perduta.	
• Con ricerca delle radici	Factor Rational (solo coefficienti razionali). Factor Radical (anche coefficienti reali). Factor Complex (anche coefficienti immaginari).	Non determina le radici.	Factor	

Le tabelle che seguiranno sono invece basate su criteri di confronto da noi stabiliti e definiti nel capitolo riguardante il confronto de' L'Algebrista con altri softwares.

Criterio	DERIVE	MILO	THEORIST	L'ALGEBRISTA
Ambiguità tra i comandi: <ul style="list-style-type: none">Isofunzionale⁴²Multifunzionale⁴³ (Conflittuale)⁴⁴	Isofunzionale, es.: <i>i vari Factor e Simplify</i> . Multifunzionale es.: Factor e Simplify (Conflittuale)	Isofunzionale (?): Multiply Out e Distribute (? verificare)	Isofunzionale: Collect e Factor Multifunzionale: Collect e Factor servono sia per fattorizzare che per sommare le frazioni (conflittuale)	Non c'è alcuna ambiguità nei comandi di base, chiaramente possono nascere ambiguità di vario tipo quando l'utente aggiunge nuovi comandi da lui/lei creati.
Molteplicità di rappresentazione ⁴⁵	No			Si sono presenti addirittura tre diverse rappresentazioni per ciascun comando.
Comandi <i>oscuri</i> o Comandi <i>chiari</i>	<i>I comandi di base sono in generale oscuri, anche perché scritti in inglese.</i>	I comandi di base sono in generale oscuri, anche perché scritti in inglese	I comandi di base sono in generale oscuri, anche perché scritti in inglese	Grazie alle molteplici rappresentazioni, ed alla particolare iconografia utilizzata, risultano essere chiari .

⁴² Diciamo che due comandi diversi sono *isofunzionali* rispetto ad una funzionalità se la condividono.

⁴³ Comandi che hanno più di una funzionalità.

⁴⁴ Comandi che hanno più funzionalità, ma che siano molto distanti tra di loro, ad esempio in THEORIST il comando **Factor** serve sia per fattorizzare che per eseguire la somma tra frazioni.

⁴⁵ Ci riferiamo a situazioni in cui un unico comando è rappresentato in più modi, ad esempio un'icona ed una parola scritta. Questa caratteristica ha rilevanza in quanto fornisce simboli diversi per rappresentare uno stesso oggetto, ma che magari sono tipici di ambiti diversi, quindi forniscono il segno giusto da usare a seconda della situazione. Ad esempio in L'Algebrista ogni proprietà è rappresentata da un bottone, che cliccato agisce sulle espressioni, ma allo stesso tempo sul bottone c'è una scrittura in simboli matematici che lo rappresenta e c'è anche scritto per esteso il nome della proprietà. In questo modo si forniscono all'utente un riferimento da usare durante l'interazione con il computer (il bottone), un riferimento adatto al linguaggio verbale (il nome per esteso), ed un riferimento adatto al linguaggio matematico (la scrittura in simboli).

Comandi <i>ricorsivi</i> ⁴⁶ o Comandi <i>statici</i> ⁴⁷	<i>Ricorsivi</i>	<i>Ricorsivi</i>	<i>Ricorsivi</i>	<i>Statici</i>
È possibile trasformare un'espressione agendo solo su una sua sottoespressione?	<i>No: se si prova a farlo esso automaticamente tutta l'espressione</i>	<i>Si</i>	<i>Si</i>	<i>Si</i>

Analizziamo ora più dettagliatamente il rapporto tra i softwares in questione e la matematica, ed in particolare la teoria delle espressioni algebriche

Criterio e/o assioma	DERIVE	MILO	THEORIST	L'ALGEBRISTA
Esiste un legame limpido tra software ed assiomi?	No: i comandi rappresentano procedure piuttosto complesse	No: i comandi rappresentano procedure piuttosto complesse	No: i comandi rappresentano procedure piuttosto complesse	Si: ciascun bottone di base rappresenta un assioma
Proprietà commutativa	No: esiste solo un comando, Simplify, che riordina una espressione in modo crescente	No: c'è solo la possibilità di riordinare, in modo crescente o decrescente, le somme.	No: solo Simplify in ordine crescente	Si: il bottone <code>a\bjb\</code> ^a applica all'espressione la proprietà commutativa della somma o del prodotto a seconda della testa. Se applicato ad un'espressione composta da più di due termini fornisce una delle tante permutazioni possibili. Se i termini sono due vengono semplicemente permutati.

⁴⁶ Un comando si dice *ricorsivo* quando, automaticamente, si autoapplica ricorsivamente all'espressione finchè questa non cambia più. Un esempio è il comando Expand che effettua ricorsivamente, ed implicitamente, le operazioni "distribuisce", "somma", "riordina"; queste operazioni elementari vengono ripetute ricorsivamente fino a che l'espressione non cambia più.

⁴⁷ Diciamo che un comando è *statico* quando effettua una singola trasformazione senza cercare di mutare, automaticamente, ulteriormente, l'espressione, ovvero si ferma comunque al primo passo, al primo livello..

Proprietà associativa	No è solo possibile aggiungere o togliere parentesi a mano.	No	No	Si: $a \setminus (b \setminus c) \neq (a \setminus b) \setminus c.$
Proprioietà distributiva	Non esiste alcun comando che condensi in se questa proprietà, di fatto è spezzata nei due comandi Expand e Factor	Non esiste alcun condensi in se questa proprietà, di fatto è spezzata nei due comandi Multiplu Out e Factor	Non esiste alcun comando che condensi in se questa proprietà, di fatto è spezzata nei due comandi Expand e Factor	Si: $a \bullet (b+c) \neq a \bullet b + a \bullet c$

Confronto tra il modello de' L'Algebrista e gli altri softwares

Criterio e/o assioma	DERIVE	MILO	THEORIST	L'ALGEBRISTA
Obiettivi didattici	Nessuno: è stato realizzato per essere uno strumento per matematici esperti			<ul style="list-style-type: none"> Introduzione degli studenti alla manipolazione simbolica Introduzione degli studenti all'idea di teoria e di dimostrazione
Tipologia	Manipolatore Simbolico	Manipolatore Simbolico	Manipolatore Simbolico	Ambiente Algebrico Elementare
Sviluppabilità	No	No	No	Si
Ambiente in cui è immerso	Nessuno	Nessuno	Nessuno	Ambiente Globale Multifunzionale
Ambiente con cui interagisce	Nessuno	Editore di testo	Editore di testo	<ul style="list-style-type: none"> Manipolatore Simbolico Editore di testo
Teoria di base	Espressioni algebriche	Espressioni algebriche	Espressioni algebriche	Espressioni algebriche
Relazione tra comandi ed assiomi	Confusa	Confusa	Confusa	Biunivoca
Comandi	Complessi Procedurali Oscuri	Complessi Procedurali Oscuri	Complessi Procedurali Oscuri	Elementari Relazionali Chiari
Tipologia dell'interfaccia	Non Strutturale Non Completa	Non completa	Strutturale Non Completa	Strutturale Completa

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

BIBLIOGRAFIA

- [1] AURICCHIO V., DETTORI G., LEMUT E.: *Uso di software per la manipolazione algebrica nella scuola superiore*. Rapporto tecnico n° 10/96, Istituto per la Matematica Applicata – CNR, Genova, 1996.
- [2] ARZARELLO F., BAZZINI L., CHIAPPINI P.: *L'Algebra come strumento di pensiero*. Progetto strategico del C.N.R. "Tecnologie e innovazioni didattiche (innovazioni didattiche per la Matematica)", Quaderno n. 6, 1994.
- [3] BATTELLI M., PUVIANI G., MORETTI U.: *Matematica sperimentale*. Centro Programmazione Editoriale, S. Prospero, 1986.
- [4] BOARI F.: *Algebra Elementare*. SATET, Torino, 1947.
- [5] BROUSSEAU G.: *The Crucial Role of the Didactical Contract in the Analysis and Construction of Situations in Teaching and Learning Mathematics*. In H. Steiner (Ed.) "Theory of mathematics Education" (pp. 110-19). Bielefeld: Institut für Didaktik der Mathematik (IDM), Universität Bielefeld, 1984.
- [6] CERVELLATI P. L.: *La città bella (il recupero dell'ambiente urbano)*. Società editrice il Mulino, Bologna 1991.
- [7] CHEVALLARD Y., JOHSUA M., MERCIER A. : *Problemes d'ingenierie didactique dans l'enseignement des debuts de l'algebre*. "Troisieme ecole d'ete de didactique des matematiques", 1984.
- [8] CHIAPPINI G., BOTTINO R. M.: *La natura della mediazione offerta dai sistemi basati su micromondi all'apprendimento della matematica: il sistema ARI-LAB*. Unpublished manuscript. Seminario nazionale di Ricerca in Didattica della Matematica "Micromondi, ipertesti e sistemi per la comunicazione nella didattica matematica". Pisa, 12-14 Dicembre 1996.
- [9] COCKROFT W. M.: *Mathematics counts*. HMSO, London, 1982.
- [10] D'AMORE B., DE FLORA A.: *Algebra elementare (per gli istituti magistrali)*. Nicola Zanichelli S.P.A., Bologna, 1974.

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

- [11]DISESSA A., HOYLES C., NOSS R.: *Computers for Exploratory Learning*. Springer-Verlag, Berlino, 1995.
- [12]DIX A., FINLAY J., ABOWD G., BEALE R.: *Human-Computer Interaction*, (seconda edizione). Prentice Hall Europe, 1998.
- [13]GRAY W. T., GLYNN J.: *Exploring Mathematics with Mathematica*. Addison-Wesley Publishing Co., Redwood City, 1991.
- [14]HOYLES C.: *Learning Mathematics and Logo*. MIT Press, Cambridge, 1992.
- [15]HOYLES C., SUTHERLAND R.: *Logo Mathematics in the Classroom*. Routledge, London, 1989.
- [16]KAPUT J.: *Information Technology and Mathematics: Opening New Representational Windows*. "Journal of Mathematical Behaviour" 5, 187-207. 1986.
- [17]KYERAN C.: *Concepts associated with the equality symbol*. "Educational Studies in Mathematics", 12. D. Reidel Publishing Co., Dordrecht, 1981.
- [18]MAEDER R.: *Programming in Mathematica*, (2nd ed.). Addison-Wesley Publishing Co., Redwood City, 1991.
- [19]MARIOTTI M.A.: *Reasoning geometrically through the drawing activity*. In "Proceedings of the 20th PME Conference", Valencia, 1996.
- [20]MARIOTTI M.A.: *Justifying and proving: figural and conceptual aspects*. Hejny & Novotna, (eds) "Proceedings of the European Research Conference on Mathematical Education", Podebrady, 1997.
- [21]MARIOTTI M.A., FISCHBEIN E.: *Defining in classroom activities*. "Educational Studies in Mathematics", 34 (pp. 219-248), 1997.
- [22]MARIOTTI M.A., BARTOLINI BUSSI M.G.: *From drawing to construction: teacher's mediation within the Cabri environment*. Proceedings of 22nd PME Conference, Stellenbosh, South Africa, 1998.
- [23]MARIOTTI M.A BARTOLINI BUSSI M.G.: *Which is the Shape of an Ellipse? A Cognitive Analysis of an Historical Debate*. Proceedings of 22nd PME Conference, Stellenbosh, South Africa, 1998.

Cerulli M.: 1999, *Uso di manipolatori simbolici nell'introduzione all'algebra: realizzazione e sperimentazione di un prototipo*. Unpublished manuscript, Tesi di Laurea in Matematica, "Università degli Studi di Pisa", Pisa.

- [24]MARIOTTI M.A.: *Introduzione alla dimostrazione all'inizio della scuola secondaria superiore*. "L'insegnamento della Matematica e delle Scienze Integrate", **21B**, n.3 (pp. 209 – 52), 1998.
- [25]MARASCHINI W., PALMA M.: *Matematica di base 1*. Paravia, Torino, 1987.
- [26]NOSS R., HEALY L., HOYLES C.: *The construction of mathematical meanings: connecting the visual with the symbolic*. "Educational Studies in Mathematics", Kluwer Academic Publishers, 1997.
- [27]NOSS R., HOYLES C.: *Windows on mathematical meanings (learning cultures and computers)*. Kluwer Academic Publishers, Dordrecht, 1997.
- [28]PAPERT S.: *Mindstorms. Children Computers, and Powerful Ideas*. Basic Books, New York, 1980.
- [29]PRODI G.: *Matematica come scoperta, 1*, (Prima Edizione). Casa editrice G. D'Anna, Messina-Firenze, 1975.
- [30]SFARD A.: *Symbolizing mathematical reality into being – or how mathematical discourse and mathematical objects create each other*. Unpublished manuscript, 1997.
- [31]SIVIGLIA P.: *Elementi di Algebra*. Istituto Geografico De Agostini, Novara, 1987.
- [32]VYGOTSKY L.S.: *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press, Cambridge, Massachussets, 1978.
- [33]VILLANI V.: *Matematica (idee e metodi)*. La Nuova Italia Editrice, Firenze, 1979.
- [34]WOLFRAM S.: *The Mathematica book* (3rd ed.). Wolfram Media / Cambridge University Press, 1996.
- [35]ZINGARELLI N.: *Lo Zingarelli 1994*. Zanichelli editore S.p.A., Bologna, 1993.

INDICE

Introduzione.....	2
Le nostre motivazioni.....	2
Il problema didattico.....	4
Quadro di riferimento.....	5
Presentazione del progetto.....	6
La teoria dell'object mediation.....	8
Cosa significa oggettualizzare.....	8
La costruzione dei significati matematici.....	12
Verso l'oggettualizzazione dei simboli.....	14
Presentazione del software.....	17
Idee di base.....	17
Descrizione del software.....	20
Perché non usare i normali comandi di <i>Mathematica</i>	27
La sperimentazione.....	29
Il percorso pianificato e le nostre aspettative.....	29
Il percorso effettivo.....	32
Osservazioni sulla sperimentazione.....	35
Confusione a livello puramente verbale per quanto riguarda le proprietà fondamentali delle operazioni.....	35
Confusione a livello concettuale per quanto riguarda le proprietà delle operazioni.....	37
Evoluzione del concetto di proprietà: alcuni protocolli esemplari.....	39
Interpretazione della sperimentazione in termini di <i>object mediation</i>	50
Confronto con altri software.....	62
Gli obiettivi de' L'Algebrista: le prime conseguenze sul modello.....	62
Tipologie di ambiente.....	64
Considerazioni didattiche sulle tipologie di ambiente.....	70
Classificazione dei comandi.....	71
La questione dell'interfaccia.....	73
Considerazioni didattiche sulla questione dell'interfaccia.....	75
Costruzione del modello.....	76
L'Algebrista e l'object mediation.....	83
L'Algebrista e Cabri Géomètre.....	86
L'Algebrista ed i linguaggi di programmazione.....	89
Il percorso didattico.....	93
Confronto de' L'Algebrista con Milo, Theorist e Derive.....	117
Bibliografia	126