

C16

periodico mensile per Commodore 16 e MSX
con listati di giochi e routines

Distribuzione per l'Italia: Messaggerie Periodici S.p.A.
aderente A.D.N., viale Famagosta 75, Milano,
tel. 84.67.545 - Categoria postale gruppo 3

L. 11.000

MSX

Ottobre 1990 - n. 39 anno VI

Registraz. n. 557 del 18/10/86 presso il Trib. di Milano
Gruppo Editoriale International Education srl: direz., redaz., amm.ne:
viale Famagosta 75 - Milano

Direttore responsabile: Graham Johnson

GAMES GAMES GAMES GA



12
giochi
12

MSX

C-16
compatibile col
PLUS 4



n° 39

16/MSX

Cosa contiene la Cassetta?



C = 16:

1. Far West
2. Rebus
3. Xargon
4. Wizworms
5. Starways
6. Fantaball

MSX:

1. Circus
2. Boxing
3. Deathwish
4. Chopper
5. Hockey
6. Paradise
7. Giallo

* sommario *

pagina	2	Cosa contiene la cassetta? Sommarlo Abbonamenti Avvertenze Caricamento
	3	Recensione libri Editoriale
	4	Sfida al Commodore - Videogames
	6	Listate con non C= 16 e Plus 4
	7	News
	8	MSX Challenge - Videogames
	11	Basic
	17	Computer & Programmazione
	19	Telematica ed educazione
	20	Dentro l'MSX
	23	Posta

**C16/MSX E' IN EDICOLA
OGNI VENERDI' DELL'ULTIMA
SETTIMANA DEL MESE**

Desidero abbonarmi alla rivista 16/MSX allo speciale prezzo di £ 100.000 per 10 copie.

COGNOME _____ NOME _____

VIA _____

CAP _____ CITTÀ _____ PROV. _____

Allego assegno ric. versamento sul c/c postale n. 11319209
intestato a Gruppo Editoriale International Education.

Le richieste di abbonamento andranno inoltrate a:

Gruppo Editoriale International Education
Viale Famagosta, 75 - 20142 Milano - tel. 89502256

Da oggi potrai abbonarti a 16/MSX e ricevere la tua rivista comodamente a casa semplicemente sottoscrivendo uno speciale abbonamento per 10 numeri allo specialissimo prezzo di £ 100.000 anziché £ 110.000.

Potrai così assicurarti la tua copia, risparmiare subito 10.000 lire e avere la sicurezza del prezzo bloccato per tutta la durata dell'abbonamento.

AVVERTENZE

Questa cassetta è stata registrata con cura e con i più alti standard di qualità.

Leggete con attenzione le istruzioni per il caricamento. Nel caso in cui, per una ragione qualsiasi, trovaste difficoltà nel caricare i programmi, spedite la cassetta al seguente indirizzo:

Gruppo Editoriale International Education srl
Viale Famagosta, 75
20142 Milano

Testeremo il prodotto e, nel caso, lo sostituiremo con uno nuovo senza costi supplementari.

attenzione! attention! look out! achtung!

Occhio all'azimut

Per la buona lettura della cassetta occorre che la testina del registratore sia pulita ed allineata col nastro. Se così non fosse potrebbe accadere che sul video appaia "error". Pulite allora la testina del registratore con un cottonfioc imbevuto di alcool. Se nonostante questa operazione il computer continua a non caricare bene prendete un cacciavite ed agite direttamente (in senso orario o antiorario) sulla vite apposita di regolazione dell'azimut.

Se avete un Commodore 16 digitate LOAD e RETURN, quindi avviate il registratore.

Per un buon caricamento dei programmi è opportuno tenere il registratore lontano dal monitor e dall'alimentatore.

Se siete i possessori di un MSX per caricare ogni singolo programma digitate RUN "CAS:" e RETURN.

Ogni eventuale variazione apparirà in calce alle recensioni dei giochi alla rubrica MSX Challenge.

Recensione libri

IL MANUALE MSX

di P. Hoffman

Edito da: McGraw-Hill

Pagine: 330

Prezzo: 27.000 lire

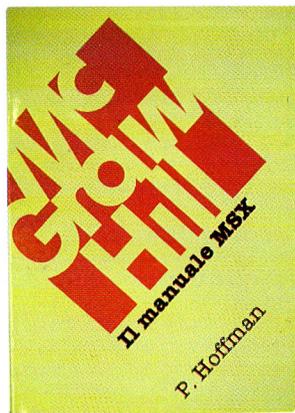
Questo libro è una completa guida al nostro beneamato standard che ci mette a disposizione un ottimo sistema operativo -l'Msx-Dos derivato direttamente dall'Ms-Dos - e da un potente Basic che non ha nulla da invidiare al famoso Basic Microsoft Standard da cui deriva direttamente.

Dopo un'introduzione alle caratteristiche hardware e software dei sistemi MSX, attraverso istruzioni e le funzioni dell'Msx-Basic, questo manuale porta l'utente, anche il più inesperto, a programmare il proprio computer e a sfruttarne le avanzate prestazioni grafiche e sonore.

L'utente più esperto, che si ritrova con un sistema decisamente completo dotato di un ottimo disk drive che serve anche ad usare Msx Disk, troverà una buona presentazione analitica dei comandi dell'Msx-Dos per la gestione dei file su disco e per la creazione di files batch.

Il libro è decisamente di buon livello e completo, e non si dilunga in descrizioni troppo lunghe e complesse che possono annoiare il lettore.

Dal punto di vista tecnico è un supporto valido e può essere



di aiuto anche ai programmatori più esperti.

IL MANUALE MSX è diviso in tre parti fondamentali: la prima - **IL COMPUTER MSX** - introduce al lettore il sistema, introducendo i diversi termini tecnici e le caratteristiche dell'Msx. Parla anche dei diversi programmi esistenti, di Word Processor e Fogli elettronici, e di cosa l'utente deve cercare in un programma. Parla poi dell'espandibilità del sistema e di come farlo, concludendo con la descrizione di come installare il sistema.

La seconda parte - **L'MSX-BASIC** - parte dai più semplici concetti della programmazione fino a portare il lettore principiante a fare un proprio programma. Parla dei diversi comandi Basic dividendoli nei diversi tipi e nelle diverse applicazioni possibili, concludendo con l'introduzione della programmazione in Assembler e dell'interfaccia di questo col Basic.

La terza parte - **L'MSX-DOS** - è interamente dedicata al sistema operativo e inizia con un'introduzione semplice e chiara per concludere il tutto con la descrizione e l'uso delle potenzialità di questo Dos.

In appendice troviamo una guida rapida all'Msx-Basic e una guida rapida all'Msx-Dos, una descrizione dei messaggi d'errore del Basic, una tabella di caratteri ASCII e una guida alla tastiera.

Il libro, come tutti i prodotti dedicati all'MSX, è abbastanza difficile da reperire e forse potete trovarne qualche copia avanzata nelle migliori librerie specializzate. Se non lo trovate l'unica soluzione è quella di rivolgersi all'Msx Club Italia che mette a disposizione dei soci delle fotocopie di diversi testi, compreso questo. I prezzi ovviamente non sono quelli di copertina, ma coprono le spese delle fotocopie. Per maggiori informazioni l'indirizzo è: MSX CLUB ITALIA -C.P. 34 - 20075 LODI CENTRO (MI).

EDITORIALE

Benvenuti ad un nuovo numero di C16/MSX!

Come di consueto introduciamo il contenuto della rivista di questo mese.

Innanzitutto, troverete nuovi giochi e una nuova avventura di testo, la recensione di un libro, una nuova parte del corso di Basic e un articolo che ogni mese varia e che questa volta è dedicato a "Tele-matica e Educazione".

Continuano poi i nostri corsi: il Basic, Computer & Programmazione e la serie di articoli dedicati all'interno della nostra macchina, Dentro l'Msx. Purtroppo anche questo mese niente listato per Msx, ma è stato sacrificato al maggior spazio dedicato agli altri articoli, ma fateci sapere cosa ne pensate. Ritorna la posta e, come lo scorso mese, ritornano le News e con questo e tutto.

Vogliamo ringraziare chi ci segue e chi ci scrive entusiasta.

Scrivete! Scrivete! Scrivete! Vogliamo consigli, aiuti di qualsiasi genere, commenti anche cattivi: tutto ciò che ci possa aiutare a migliorare.

Ma vogliamo anche invitarvi ad entrare in contatto con il nuovo MSX CLUB un'importante risorsa che, anche grazie a noi, sta raccogliendo numerose adesioni. L'indirizzo del club è:

Msx Club - Casella Postale 34 - 20075 LODI CENTRO (MI).

Vi ricordo che inviando 1000 lire in francobolli riceverete il catalogo del software disponibile e tutte le indicazioni relative alle modalità di iscrizione.

Ma ora basta! Inserite la cassetta digitate il comando di caricamento e divertitevi! Arrivederci al prossimo numero...

LA REDAZIONE

sfida alico

1. FAR WEST

Approfittando della tua assenza da casa durante l'ultima esplorazione del Klondyke alla ricerca della mitica miniera d'oro, ecco che la famigerata banda dei McMurdock, pericolosi banditi senza tetto né legge, ha assaltato il paese attaccando alle prime luci dell'alba e cogliendo gli abitanti, ancora immersi nel sonno, completamente di sorpresa e senza che potessero opporre una valida resistenza.

I McMurdock sono da anni il terrore del Far West, ognuno di loro è accusato di decine di rapine e uccisioni in duelli dalla dubbia regolarità, e sul loro capo pende una taglia di centomila dollari, vivi o morti.

Gli abitanti del paese che non hanno fatto in tempo a fuggire sono rimasti barricati in casa, nascosti nelle soffitte o negli armadi, e non osano neanche uscire dai loro nascondigli temendo per la loro vita, mentre i fuorilegge stanno deprestando e distruggendo ogni cosa, dalla banca al

saloon. Chi potrà salvare il paese, costruito faticosamente in mezzo al deserto, e ultimo avamposto della civiltà?

Inutile dire che i tuoi concittadini aspettano fiduciosi il tuo arrivo per porre fine all'occupazione del paese, sconfiggendo una volta per tutte i banditi, protagonisti anche in passato di sanguinose scorribande.

Ma i McMurdock hanno saputo del tuo arrivo e si sono annidati tra le case del paese, pronti a comparire all'improvviso cercando di coglierti alle spalle, di sorpresa. Cerca quindi di muoverti velocemente, tenendo sempre d'occhio porte e finestre, e quando vedi un bandito non esitare neppure un attimo a sparare.

Per muoverti usa il joystick in porta 2 o i tasti: Z sinistra - X destre - ; alza la mira - / abbassa la mira - Spazio fuoco.

Per sparare portati davanti al bersaglio e aggiusta la mira muovendo la leva avanti o indietro (osservando la posizione del braccio con la pistola).

FAR WEST

In alto sono indicati i colpi ancora nel tamburo; quando hai finito le munizioni corri verso destra, fino al cavallo, e ricarica la pistola.

2. REBUS

Eccovi un simpatico programma che, oltre ad esercitare la vostra memoria e la capacità di osservazione, vi permetterà di imparare o ripassare un po' di inglese, lingua ormai universalmente adottata come mezzo di comunicazione internazionale.

Premete il tasto Return per dare avvio al gioco. Nella parte alta dello schermo compariranno, accompagnate da una simpatica musicchetta, sei caselle, nelle quali verranno posti, uno per uno, sei oggetti, mentre per ognuno di essi potrete leggere, in basso, il nome in inglese:

- a tree (un albero)
- a pig (un maiale)
- a face (una faccia)
- a fish (un pesce)
- a bird (un uccello)
- a house (una casa)

I sei oggetti verranno quindi coperti da una tendina e il programma vi chiederà:

Where is it?

che significa

dove è (esso)

Muovete il quadrato nero con la barra di spazio e premete Return sulla posizione dove ritenete si trovi l'oggetto.

Vediamo le altre frasi visualizzate dal programma:

- Is it there? (è lì?)
- Try again! (prova ancora!)
- Here it is (ecco qual)
- You found it! (l'hai trovato!)
- You found these (hai trovato questi)

REBUS



3. XARGON

"Queste cose succedono solo nei film!" stai pensando mentre la tua astronave si trova improvvisamente in mezzo ad uno scenario apocalittico.

La zona che stai perlustrando si è sempre distinta per la scarsa bellicosità, infatti tu hai inserito il pilota automatico e la tua potente e sofisticata astronave ha mantenuto fin qua una tranquilla andatura... di crociera.

Poi, d'improvviso, l'apocalisse! Non hai avuto nemmeno il tempo di disinnestare il pilota automatico, visto che era molto più urgente mettere mano alle mitragliatrici.

Così, in pochi istanti ti sei reso conto di quali fossero gli oggetti più pericolosi, tra la miriade di strani corpi volanti, di quali fossero da evitare accuratamente e quali invece dovessero essere colpiti con la massima priorità.

Per iniziare scegli il numero di giocatori con il tasto funzione F1, con F2 il livello iniziale di gioco per il giocatore 1; se giochi insieme a un tuo ami-

co, hai anche la scelta del numero di joystick, per combattere, a turni alternati, con lo stesso joystick o con uno a testa, e infine con HELP puoi scegliere il livello di partenza del secondo giocatore.

Se usi un solo joystick collegalo in porta 1; a causa dell'estrema velocità del gioco, non è possibile il comando dell'astronave tramite la tastiera.

Gli attaccanti arrivano dall'alto, secondo un percorso a zig-zag, spesso insieme ad altre astronavi minori, lanciando bombe a frammentazione e dividendosi quando colpiti per rendere più difficile l'abbattimento.

Per di più, gli avversari sono riusciti a piazzare ai limiti dell'area di battaglia due potenti batterie di cannoni laser, che si spostano continuamente battendo con il loro tiro incrociato tutto il campo.

Con la tua astronave puoi spostarti nelle quattro direzioni, ma ricordati che potrai sparare soltanto in avanti, e non perdere mai d'occhio le batterie di laser!

XARGON



modore

4. WIZWORMS

Polso fermo? Riflessi pronti al massimo livello? Nervi d'acciaio?

Eccoti davanti al più impegnativo giardino che si sia mai visto. Lungo lo schermo, con una traiettoria imprevedibile zigzagante scenderà uno strano essere vermiforme con intenzioni tutt'altro che pacifiche.

Dovrai distruggerlo prima che ti arrivi addosso o saranno guai seri!

Attento però: ogni volta che lo colpisci, il verme si dividerà in due segmenti che prenderanno direzioni diverse.

Presta inoltre attenzione alle lumache, ai ragni e agli altri insetti che compariranno qua e là sullo schermo intralciandoti la mira o addirittura investendoti e facendoti perdere una delle 4 vite a disposizione.

Tieni presente poi che, nei punti dove colpirai il serpente o gli altri insetti che compariranno qua e là sullo schermo intralciandoti la mira o

addirittura investendoti e facendoti perdere una delle 4 vite a disposizione.

Tieni presente poi che, nei punti dove colpirai il serpente o gli altri insetti che si agitano sullo schermo, si formeranno dei funghetti che impediranno ai tuoi proiettili di proseguire.

Questi funghi possono essere eliminati con 2 colpi ben assestati, anche se fortunatamente il loro contatto non vi arrecherà alcun danno, anche se rallenteranno comunque i tuoi movimenti.

Forse due occhi non basteranno per guardarti da tutte le insidie che ti aspettano.

Per contrastare i tuoi petulanti avversari potrai muoverti in tutte le direzioni e... sparare, sparare, sparare.

Col tasto F1 seleziona il numero dei giocatori, col tasto F2 il comando tramite tastiera, con i tasti: X sinistra - S destra - ; giù - Spazio fire; con F3 viene abilitato il modo di allenamento, e con HELP viene attivata la pausa gioco (premete un tasto qualsiasi per ripartire).

WIZWORMS



5. STARWAYS

Eccoti uno spaziale velocissimo e che richiede nervi saldi e riflessi pronti, per veri "addetti ai lavori".

Gli alieni invasori stanno tentando un ennesimo attacco al Pianeta. Ma questa volta, visti gli scarsi risultati conseguiti con gli attacchi precedenti, sono riusciti a convogliare sulle difese terrestri uno sciame di meteoriti, tentando di disorientare le nostre difese e di travolgere lo schieramento di astronavi da battaglia posto a salvaguardia della Terra all'altezza delle fasce di Van Allen.

Contro questi meteoriti, alcuni dei quali raggiungono una massa notevole, le normali armi di cui è dotata la tua astronave sono pressoché inservibili; l'unica possibilità di salvezza consiste nello schivarli

abilmente, evitando le collisioni che distruggerebbero inevitabilmente l'astronave.

Solo all'ultimo momento gli scienziati del laboratorio di armamenti spaziali sono riusciti a elaborare uno schermo anti-meteoriti, che ha però un'efficacia limitata e può essere attivato solo 3 volte, premendo il tasto Control, prima che si esaurisca lasciandovi del tutto privi di difesa.

Per schivare i meteoriti usa i tasti:

I sinistra - P destra.

Premendo il tasto I, nella schermata iniziale, verrà visualizzata una simpatica pagina di introduzione e spiegazione del gioco.

Premendo invece Q terminerete il gioco e il computer si resetterà preparandosi per caricare un altro programma.

STARWAYS



6. FANTABALL

Eccoti immerso nel bel mezzo di una partita futuristica nel corso della quale dovrai affrontare durissime quanto velocissime prove di rara abilità e destrezza.

Dovrai dirigere una palla che continua instancabilmente a rimbalzare lungo un percorso irto di difficoltà e di sorprese.

Scopo di questo gioco è infatti far percorrere alla palla il cammino, cercando di non incappare in troppe penalità e restando solo sui punti che permettono di rimbalzare.

Lungo il percorso troverete infatti diversi tipi di superfici, ognuna caratterizzata da particolarità che non tarderete a scoprire. Ad esempio, i riquadri con una freccia ver-

so l'alto faranno lievitare la nostra palla magica per alcuni secondi, permettendo così di superare alcune zone prive di riquadri adatti per rimbalzare.

Per controllare la palla collegate il joystick in porta 2 e premete fire per iniziare.

Potete muoverla lateralmente e, entro certi limiti, accelerare o rallentare il movimento in avanti. Quando avete un ostacolo da superare, raccogliete le forze rimbalzando sulla superficie solida e quindi saltate rapidamente in avanti.

Durante il gioco, premendo Escape e quindi Fire potrete ricominciare la partita dall'inizio.

Su alcuni riquadri vedrete un punto interrogativo: passandoci sopra troverete una sorpresa, anche se non sempre gradita.

FANTABALL



LISTATE CON NOI

STAMPA SPRITE

Il listato di questo mese permette di stampare uno sprite, cioè un'immagine grafica, un simbolo o un marchio, componendolo direttamente sul listato.

Lo sprite è composto di 21 linee in altezza per 17 colonne, e può essere modificato a piacere variando la posizione degli asterischi alle linee 120-320, rimanendo però nel limite dei 21*17 punti a disposizione; la casetta presente nel listato è solo un esempio.

La stampa avviene sfruttando la modalità grafica della stampante, selezionata tramite il carattere di controllo CHR\$(8).

Il listato stampa lo sprite 10 volte; per variare il numero delle stampe, variate il numero di iterazioni del ciclo FOR-NEXT alla linea 520.

Nel listato è stato utilizzato un particolare incolonnamento, detto indentazione, allo scopo di rendere più leggibile la sequenza di esecuzione.



```
100 DIM T$(7),C$(3), T(6)
110 REM 12345678901234567 COLONNE DI STAMPA
120 DATA " " "
130 DATA " " * "
140 DATA " " * "
150 DATA " " * * "
160 DATA " " * * * "
170 DATA " " * * * * "
180 DATA " " * * * * "
190 DATA " " * * * "
200 DATA " " * * "
210 DATA " " * "
220 DATA " * * * * * * * * * * "
230 DATA " * * "
240 DATA " * * * * * * * "
250 DATA " * * * * * * "
260 DATA " * * * * * * "
270 DATA " * * "
280 DATA " * * "
290 DATA " * * * * "
300 DATA " * * * "
310 DATA " * * * "
320 DATA " * * * * * * * * * * "
330 PRINT "COSTO CALCOLANDO I CODICI DELLA CASSETTA"
340 FOR K=1 TO 3
350 :C$(K) = ""
360 :FOR J=1 TO 7
370 ::READ T$(J):PRINT T$(J)
380 :NEXT J
390 :FOR L=1 TO 17
400 ::I=0
410 ::FOR J=1 TO 7
420 :::T(I)=ASC(MID$(T$(J),L,1))
430 :::IF T(I)=42 THEN T(I)=1:ELSE T(I)=0
440 :::I=I+1
450 ::NEXT J
460 ::T=0:FOR J=0 TO 6:T=T+T(J)*2^J:NEXT J:T = T+128
470 ::C$(K) = C$(K)+CHR$(T)
480 :NEXT L
490 NEXT K
500 OPEN 4,4:PRINT#4,CHR$(8) : REM STAMPA SPRITE
510 FOR J=1 TO 3
520 :FOR K=1 TO 10:PRINT#4,C$(J);:NEXT K:PRINT#4
530 NEXT J:PRINT#4,CHR$(15)
540 CMD 4:LIST:PRINT#4:CLOSE 4 : STAMPA LISTATO (OPZIONALE)
```

- Nella rivista di videogames inglese "ACE" abbiamo letto con grande sorpresa che diversi topgames come **Chase HQ**, **Test Driver II**, **The Duel** e **Batman - The Movie** sono disponibili in Spagna anche per lo standard Msx.

Ed è vero!

Ci siamo stati e abbiamo visto molto software, in gran parte a prezzi tra le 20 e le 30 mila lire.

Una soluzione può essere andare in Spagna, ma forse potete ottenere qualcosa rivolgendovi a: LASP - Alfonso I 28 -50003 ZARAGOZA

- La SNK, che produce videogiochi da bar, importerà in Europa il **Neo Geo**, la nuova potente console con la qualità dei giochi da bar.

Il prezzo dovrebbe aggirarsi sulle 800.000 lire mentre i giochi dovrebbero essere venduti a circa 50.000 lire l'uno.

Prezzi incredibili?

No se pensate che praticamente avete in casa un videogioco da bar che permette di salvare il punteggio e la situazione del gioco per poi riprenderla in qualsiasi sala giochi grazie a una scheda di memoria esterna.

Ma è anche possibile l'operazione inversa, cioè continuare a casa una partita iniziata al bar.

- La **Nintendo** avrebbe già lanciato una sua macchina da 16 bit se la Sega non l'avesse battuta sul tempo col suo Genesis.

A partire da questo autunno comincerà in Giappone la commercializzazione di questa nuova macchina destinata a dare battaglia alla Sega.

Tutto all'insegna dell'evoluzione Msx.

- La società giapponese Fujitsu, che più di trent'anni fa aveva costruito il primo computer giapponese, sta per rientrare alla grande nel mercato giapponese, e forse anche europeo, con suo **FM-Tower Machine**, ultimo discendente dell'Msx.

La macchina è basata di un processore 80386 ed è dotata di un CD-Rom.

In più, la macchina ha un display con una risoluzione di 640 per 480 pixels e 256 colori a 64 toni di grigio, e suono stereo.

La macchina è, naturalmente, una console fantastica e anche totalmente Ms-Dos compatibile.

In Inghilterra è già disponibile, anche se in versione Giapponese. Il prezzo è di 2250 sterline inglesi, cioè circa 5.000.000 di lire.

- **MSX 3**...un mistero di Agatha Christie?

Nel paese degli Msx (ndr Giappone) corre sempre la voce di un futuro Msx 3 dopo il recente Msx 2 Plus.

E finalmente, forse, abbiamo una notizia fondata: la ASCII -una società Giapponese forse già conosciuta da alcuni di voi per alcuni giochi - ha annunciato che nella primavera del 1991 inizierà a commercializzare un Msx 3.

Comunque, finché non lo vediamo non ci creiamo!

- Tra i nuovi programmi usciti in Giappone e in alcuni paesi europei c'è **SOLID SNAKE** (= Metal Gear 2).

Quando è uscito, Metal Gear era assolutamente magnifico, ma il suo successore sarà ancora meglio, non ci sono dubbi.

Il 21 luglio 1990 è iniziata la commercializzazione in Giappone ed è andato subito tutto esaurito in un giorno, almeno la prima tiratura.

- Conoscete sicuramente quei giochi con i quali si spara con una pistola ottica allo schermo.

Sì, proprio come quello della Nintendo. Ebbene, adesso c'è qualcosa di simile per il MSX.

Sempre in Giappone è uscito **PLUS-X TERMINATOR LASER**, una pistola ottica fornita con un gioco per MSX 1 e 2 su ROM.

Gli screenshots danno una bella impressione di questo gioco un gioco normale con eccellenti effetti 3D.

MSX

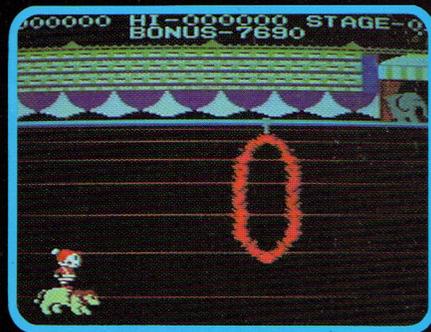
1. CIRCUS

Agli albori dell'era videogame c'erano giochi che, per le loro caratteristiche innovative, attiravano schiere di agguerriti giocatori con conseguente grande incasso. A quei tempi Circus Charlie fu un gioco decisamente innovativo che contribuì all'entrata del mondo del fumetto in quello del fumetto elettronico: un vero e proprio capostipite. Tra i diversi giochi riproposti in versione Msx, uno dei migliori programmi Konami è proprio questo. La storia è abbastanza semplice ma originale: un simpatico clown deve esibirsi in diverse prove corrispondenti ai diversi livelli di gioco e voi, naturalmente, dovrete guidarlo. Nel primo livello sarete sulla groppa di un leone e con esso dovrete saltare attraverso dei cerchi infuocati, ovviamente senza scottarvi. Nel secondo livello dovrete percorrere una corda sospesa in aria senza cadere e saltando le scimmiette dispettose che corrono sulla corda. Il terzo livello vede il nostro simpatico clown in equilibrio su un grosso pallone da cui saltare su un altro, un altro ancora e così via fino alla fine. Nel quarto livello poi sarete in piedi sulla groppa di un potente cavallo bianco che correrà lungo un percorso composto da piattaforme elastiche su cui dovrete saltare per poter-

lo evitare. Del quinto livello non vi diciamo nulla, anche perché è più bello giocare con un po' di mistero su ciò che ci aspetta.

Una volta caricato il gioco vedrete la presentazione iniziale e quindi il menù principale, subito seguito dalla dimostrazione. Quest'ultima può essere interrotta premendo la barra spaziatrice. Il menù principale offre quattro opzioni: la prima e la terza permettono di giocare da soli con il joystick o con la tastiera, mentre la seconda e la quarta opzione consentono di giocare in due, rispettivamente con joystick o con tastiera. Con lo spazio o con il tasto fire del joystick è possibile saltare: premendo poco saltate poco, premendo di più saltate di più! (Ma no?) Questo è particolarmente importante perché è sempre necessario saltare con una certa precisione, evitando gli ostacoli sempre mortali. Il gioco, sempre in scroll orizzontale verso sinistra, ha un tempo limite per ogni livello di gioco: se finisce il tempo perdetevi una vita ma se arrivate alla fine il tempo rimasto si trasforma in punti bonus.

Come di consueto nei giochi Konami, avrete a disposizione tre omni che perderete sia allo scadere del tempo sia colpendo un ostacolo. Quando finite la partita avrete alcuni secondi di tempo per decidere se continuarla oppure no.



COMANDI

Tasti:

[CURSORI] = Scelta opzione nel menù / Movimenti Clown

[SPAZIO] = Inizio gioco / Salto

Joystick in porta 1 per il primo giocatore, in porta 2 per il secondo

2. BOXING

Tra le simulazioni di Boxe esistenti per Msx, questa che vi proponiamo è sicuramente una delle migliori, uno dei pochi programmi che riesce a sfruttare al massimo della sua potenzialità la grafica dell'Msx 1. Una volta completato il caricamento del gioco sullo schermo comparirà la presentazione e quindi il menù principale, subito seguito dalla dimostrazione. Quest'ultima può essere interrotta tramite la barra spaziatrice o un tasto cursore. Il menù principale offre quattro opzioni: le prime due prevedono un giocatore contro il computer con due diversi livelli, le ultime due permettono di giocare in due con due diversi livelli come nelle prime due opzioni. I livelli di difficoltà sono GAME A e GAME B. La selezione dell'opzione può essere fatta muovendo il cursore - che è una "manina" - e poi premendo lo spazio o il tasto di

fuoco quando si è sull'opzione desiderata. Nello schermo di gioco troverete il vostro pugile - che si chiama RYU - nella parte destra dello schermo. Nella parte bassa c'è la campanella, il punteggio e l'energia rimasta a voi e al vostro avversario. In alto è posto il timer mentre sullo sfondo vedrete il pubblico attorno al ring. Usando lo spazio contemporaneamente ai diversi tasti del cursore - quelli con le frecce! - potrete sferrare diversi pugni in punti diversi oppure schivare quelli dell'avversario. Con i soli cursori vi spostate a sinistra o a destra. Potete usare due diversi joystick collegati nelle rispettive porte, la uno e la due.

COMANDI

Tasti:

[CURSORI] = Selezione opzione / Movimenti

[SPAZIO] = Inizio gioco / Dà colpo



[CURSORI] + [SPAZIO] = Dà pugno o Schiva colpo

Joystick in porta 1 per il primo giocatore, in porta 2 per il secondo

3. DEATHWISH

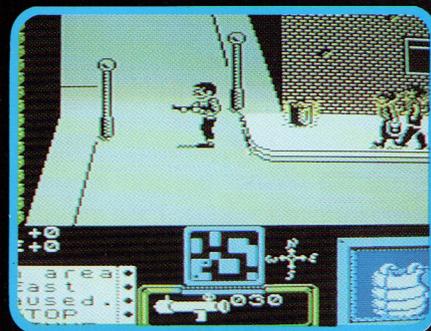
Ricordate la serie di film con Charles Bronson dedicati alla figura del Giustiziere della Notte? Se non li avete visti non importa, ma sappiate che questo gioco si ispira al terzo film della serie: «Il Giustiziere della Notte 3». Nei panni del giustiziere, armati fino ai denti, dovrete seguire il motto "Tieni pulita la tua città" girando per la città alla ricerca di ladri, assassini e compagnia bella per eliminarli fisicamente. Il vostro armamento comprende: un bazooka, un mitra, una pistola automatica e un fucile da caccia tutto con relative munizioni e un giubbino anti-proiettile per difendervi dai colpi dei vari delinquenti. L'ometto del gioco viene controllato abbastanza stranamente: potete farlo muovere a sinistra o a destra mentre per spostarlo nella città dovrete seguire la

bussola. La direzione lampeggiante indica che vi state muovendo verso quel determinato punto cardinale quindi, per esempio, se la bussola lampeggia verso Nord e voi camminate verso sinistra state andando a Nord, ovviamente andate verso Sud camminando a destra. Uccidendo di qua e di là dovrete evitare di colpire i poliziotti, le vecchiette e le donnine allegre che vi mostrano il didietro alzando la gonna: se li uccidete il vostro punteggio viene decrementato. L'indicatore INJURY in basso a sinistra nello schermo di gioco indica i danni subiti: con troppi danni morite e perdetevi l'unica vita a vostra disposizione.

COMANDI

Tasti:

[O] e [K] = Cambiano direzione rispetto alla bussola



[Z] = Sinistra - [X] = Destra

[SPAZIO] = Inizio gioco / Fuoco

[STOP] = Attiva/Disattiva pausa

[SELECT] = Seleziona l'arma da usare

[M] = Mappa

Joystick in porta 1

allende

4. CHOPPER

Tra i grandi classici dei computer games che siamo soliti proporvi non poteva assolutamente mancare Choplifter uno dei primi e più famosi giochi per computer, nato su di un vecchio Apple II. Un impavido pilota deve guidare il proprio elicottero al salvataggio dei compatrioti rimasti coinvolti nella guerra civile scoppiata nel vicino staterello centrafricano. Partendo dalla propria base in territorio neutrale, l'elicottero deve passare la linea di frontiera e raggiungere le zone abitate, atterrare e caricare più persone possibile con un massimo di sedici. Riempito l'elicottero, si deve tornare alla base di partenza scampando a carri armati e aerei nemici. L'elicottero è dotato di una mitragliatrice abbastanza imprecisa e

poco potente, ma è l'unica arma a vostra disposizione. E' proprio una missione suicida, ce la farete? Se usate il joystick è consigliabile usarne uno dedicato allo standard Msx con due diversi tasti di fuoco: uno spara e l'altro fa ruotare l'elicottero. Nella parte superiore dello schermo di gioco appaiono tre sfere colorate seguite da un numero: la sfera rossa indica i civili morti, quella blu i civili che si trovano a bordo dell'elicottero mentre la verde indica i civili portati in salvo.

COMANDI

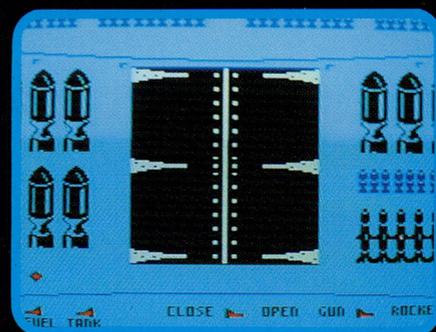
Tasti:

[CURSORI] = Movimenti elicottero

[SPAZIO] = Inizio gioco / Fuoco

[GRAPH] = Ruota elicottero

Joystick in porta 1



5. HOCKEY

Tra i giochi che siamo soliti proporvi ci sono spesso simulazioni di calcio o basket, cioè sport tipici per un videogioco sportivo. Questa volta abbiamo voluto evadere da questo stereotipo proponendovi un gioco di squadra che sta avendo un grosso successo: l'Hockey. La simulazione è fatta abbastanza bene e cerca di riprodurre l'effetto ghiaccio, soprattutto quando si muovono i giocatori in lunghe pattinate. Anche la palla si muove bene e rimbalza qua e là per la pista seguendo l'effetto ghiaccio. Le due squadre che si affrontano, blu e rossi, sono composte da tre giocatori e un portiere ed è possibile giocare in uno o due giocatori che controllano entrambe le squadre tramite due joystick. Tramite i due tasti numerici [1] e [2] è possibile scegliere il numero di giocatori, dopodiché si deve scegliere il livello di difficoltà

tra tre diverse opzioni, selezionabili tramite il corrispondente tasto numerico: [1] - Amatore, [2] - Semi professionista e [3] - Professionista. Se non si fa partire il gioco o non si sceglie il livello il programma attiva la dimostrazione che può essere interrotta premendo lo spazio. Al giocatore numero uno viene assegnata la squadra rossa mentre al computer o al secondo giocatore viene assegnata la squadra blu.

COMANDI

Tasti:

[CURSORI] = Movimenti giocatore 1

[SPAZIO] = Tira disco / Interrompe demo

[STOP] = Attiva/Disattiva pausa

[1] = Gioco con un giocatore / Livello 1

[2] = Gioco con due giocatori / Livello 2

[3] = Livello 3

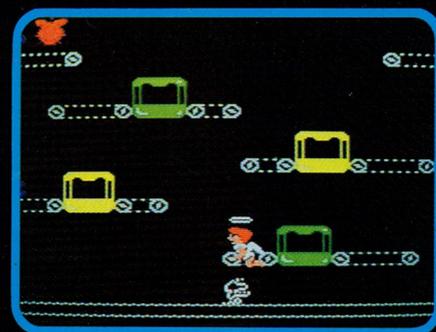
Joystick in porta 1 per il primo giocatore, in porta 2 per il secondo



6. PARADISE

In questo gioco decisamente originale il protagonista è un giovane angelo che deve essere guidato sulle tubature che trasportano le nuvole del paradiso. Il suo compito è di spingere le nuvolette di diverso colore nei bocchettoni da cui fuoriescono. Se le nuvole finiscono nei bocchettoni sbagliati non succede nulla ma se vanno in quelli dello stesso colore si ottiene un bonus di 500 punti. Per muovere le nuvole bisogna spingerle difianco mentre per farle andare nei bocchettoni bisogna spingerle dall'alto. Dai bocchettoni, però, escono delle nuvolette grigie che se si scontrano con quelle colorate le fanno cadere sul tubo. Potete distruggere queste nuvole lanciando l'aureola o spingendole verso il basso. Sul

tubo, ovviamente, non poteva mangiare un gremlin mangia-nuvole: se il mostriciattolo riesce a digerire la nuvola il nostro angelo muore. Per evitare questo dovrete colpire il mostro con l'aureola. L'aureola serve anche per difendersi dai mostri volanti che svolazzano qua e là quando si gioca con il livello di difficoltà maggiore. Una volta caricato il gioco sullo schermo apparirà il menù principale e poco dopo partirà la dimostrazione. Premendo la barra spaziatrice si interrompe la demo e si torna al menù. Tramite questo menù è possibile scegliere se giocare in uno o due giocatori e con quale livello, Normale o Arcade. La selezione avviene muovendo il cursore sull'opzione desiderata e poi premendo lo spazio. Per giocare ci sono a disposizione tre angeli.



COMANDI

Tasti:

[CURSORI] = Movimenti

[SPAZIO] = Inizio gioco / Lancia aureola

Joystick in porta 1

MSX challenge

7. GIALLO

Ecco una nuova classica avventura di testo. Per poter utilizzare l'avventura non dovete usare il solito comando LOAD "cas:", R oppure RUN "cas:", ma dovete usare l'istruzione CLOAD. Quando il caricamento è terminato dovete digitare RUN e premere il tasto di invio.

COME GIOCARE

Giocare una adventure significa guidare il protagonista di una storia verso uno scopo ben preciso, impartendo comandi tipo "PRENDI IL MARTELLO" o "ROMPI IL VASO" formate essenzialmente da un verbo seguito da un sostantivo e dalla pressione del tasto di immissione [RETURN]. Le istruzioni vanno impartite per esteso, in lingua italiana e utilizzando la seconda persona singolare. Potrete spostarvi nelle quattro direzioni cardinali NORD, SUD, EST, OVEST e ordinare SALI e SCENDI o, se la situazione lo richiede, ATTRAVERSA, ENTRA o ESCI. La descrizione del luogo in cui vi trovate e degli oggetti o dei personaggi presenti potrà scomparire a causa dello spostamento verso l'alto (SCROLL verticale) del testo: per rivederla basterà digitare DESCRIVI o GUARDA. Gli oggetti potranno essere manipolati per mezzo dei verbi PRENDI, POSA e INDOSSA o TOGLI (questi ultimi due se si tratta di ornamenti o di vestiti). Per inventariare gli oggetti trasportati dovrete digitare LIST oppure INVENTARIO. L'esame di oggetti, luoghi o personaggi attraverso ESAMINA potrà condurre in molti casi alla scoperta di altri oggetti o di particolari nascosti: se ciò dovesse accadere, lo schermo verrà cancellato e la descrizione del luogo aggiornata. Infine, ricordandovi che la soluzione di una avventura può richiedere anche giorni o settimane di "duro impegno", tenete presente che è possibile registrare su un'altra cassetta disco la situazione corrente di gioco col comando SAVE e ripristinarla in qualsiasi momento col comando LOAD. In seguito a tali istruzioni vi verrà chiesto di inserire la cassetta di destinazione dei dati e premere un tasto, dopodiché lo schermo si cancellerà per il tempo occorrente alla lettura o alla scrittura dei dati. Questo è il vocabolario essenziale di ogni avventura che si rispetti, ma il Dizionario di Gioco (PARSER), contiene molti altri vocaboli, relativi alle situazioni che incontrerete, per cui impartite normalmente le vostre istruzioni, cercando di cambiare i vocaboli nel caso in cui il computer non vi comprendesse. Ora però leggete attentamente il seguente racconto, prassi assolutamente indispensabile per poter affrontare e risolvere l'avventura.

PROLOGO

Il suo nome è Max Plank: non è un agente segreto, né una spia o tantomeno un eroe

nel senso classico della parola. Max Plank è un modesto professore di Fisica, spinto dall'amicizia verso un'avventura che potrebbe portarlo a decidere, con le sue azioni, del destino di migliaia, forse milioni, di persone. La sua auto è ferma da quasi un'ora davanti al portone di un vecchio palazzo, in una via solitaria di una poco tranquilla cittadina, ma non scWnde... uno strano presentimento gli dà la certezza che lì, nel polveroso e disordinato Ufficio di Investigazioni Private, non troverà più il suo vecchio amico Dave Rover ad aspettarlo. Del resto egli è qui proprio per questo, da quando una serie di strani sogni ha cominciato col turbare le sue notti non ha fatto altro che pensare al suo unico, vero amico. Salutò Dave diversi anni prima quando decise di lasciare la cittadina dove erano cresciuti insieme, per trasferirsi in una località del Mid-West ed aprire un'agenzia di Investigazioni Private. Ed ora che ha rintracciato il suo indirizzo quella sensazione di angoscia accenna a lasciarlo, e vorrebbe credere che lo troverà là, nel suo ufficio, a fumare una sigaretta coi piedi sulla scrivania e con il solito sorriso sornione stampato sulla faccia...

Quando salirà le scale di quel vecchio edificio ed entrerà nell'ufficio scoprirà che i suoi presentimenti erano fondati, e inizierà la sua ricerca, tra i mille pericoli della squallida cittadina. L'ultima volta che il professor Plank aveva contattato Dave, pochi giorni prima, l'amico stava indagando sul misterioso furto di un pericolosissimo virus: una brutta storia che da tempo impegnava il Governo impegnato da tempo e di cui, ufficialmente, si conosceva molto poco. Il professore aveva sempre ammirato lo spirito umanitario di Dave e nutriva per lui un'amicizia fraterna: non avrebbe mai permesso che gli succedesse qualcosa, ed ora era pronto a sfidare chiunque pur di ritrovarlo. Ma la città non gli offrirà lo stesso aiuto disinteressato: egli scoprirà che nessuno è fidato e chi lo aiuta non lo fa mai gratis...soprattutto un certo Maskerpy, un informatore o la sua amica che lavora alla biglietteria di uno strano cinema animato da movimenti sospetti: scambio di merci illegali e di informazioni di ogni genere. Ma tali informazioni si possono ottenere solo conoscendo la procedura: la ragazza del cinema, come gli aveva confidato Dave, fa entrare chi vuole vendere qualcosa a Maskerpy, avvisandolo poi al suo arrivo. La polizia è in allarme per il caso del virus, ma da parte loro non c'è nessuno sforzo che vada oltre l'arresto dei sospetti: meglio quindi evitare di affidarsi a loro... Per quanto riguarda la gente comune, nessuno si fida di nessuno e quelli implicati in loschi traffici sono coperti da attività insospettabili.

Per Max Plank i pericoli sono tanti e pur senza alcun aiuto la ricerca deve continuare fino alla fine...

GIALLO

BASIC (Parte 4)

Le variabili, in informatica come in matematica, sono scatole vuote, ciascuna delle quali è contrassegnata da un proprio nome e serve per contenere un valore numerico od una stringa di caratteri. Così facendo il programma risulta pronto per la risoluzione di problemi analoghi. Dobbiamo, però per riutilizzarlo successivamente salvarlo su floppy.

ASSEGNAZIONE

Istruzione LET

L'istruzione sicuramente più importante di un linguaggio di programmazione è quella che permette di attribuire un valore ad una variabile, come anche di modificarlo o di calcolare un'intera espressione per esempio di tipo aritmetico. Abbiamo già incontrato le variabili e ne abbiamo dato una prima definizione. Avrete capito che una variabile è il nome simbolico che diamo a qualcosa in attesa di attribuirgli uno specifico valore. Il concetto di variabile usato in informatica è praticamente lo stesso di quello usato in matematica: un nome letterale per una grandezza numerica. Di variabili nei linguaggi di programmazione ve ne sono: variabili numeriche, variabili stringa e variabili logiche. Come abbiamo già detto all'inizio, preferiamo dapprima fare degli esempi e poi dare delle definizioni rigorose. Cominciamo quindi con il mostrare degli esempi di istruzioni di assegnazione. Nei programmi precedenti, i valori delle variabili A, B, C e D venivano chiesti come dati di ingresso. Se essi fossero invece noti al momento di scrivere il programma, potremmo allora assegnarli direttamente alle rispettive variabili.

Per esempio così:

Programma 1

```
10 LET A = 25
20 LET B = 120
30 LET C = 5.6
40 LET D = 0
50 PRINT A,B,C,D
60 END
```

Se ora diamo il comando RUN, otteniamo:

```
25 120 5.60
```

(Per praticità non abbiamo più riportato il prompt READY in quanto riteniamo che il lettore abbia già capito quando esso appare).

Le istruzioni 10, 20, 30 e 40 sono quattro istruzioni di assegnazione. La parola LET si può tradurre come "poni" e vuol dire proprio "poni A uguale a 25", "poni B uguale a 120", ecc.

Con l'istruzione 50 facciamo invece visualizzare il contenu-

to delle quattro variabili e vediamo proprio i valori che avevamo loro assegnato. E' importante fare subito una precisazione. La parola LET fu coniata per le primissime versioni di BASIC, ma nella maggior parte dei BASIC attuali non è più obbligatorio usarla. Il Programma 1 si può scrivere più semplicemente così:

```
10 A = 25
20 B = 120
30 C = 5.6
40 D = 0
50 PRINT A,B,C,D
60 END
```

Ancora un esempio:

```
10 A = 25 + 10
20 B = 3.14 - 3
30 PRINT A,B
40 END
```

RUN

```
35 0.14
```

Che cosa c'è di diverso nelle ultime istruzioni 10 e 20? A destra del segno di eguale (il segno di assegnazione) invece di un valore costante abbiamo posto una espressione algebrica che il calcolatore ha calcolato ed assegnato alle rispettive variabili. Un'istruzione di assegnazione permette anche di fare dei calcoli, per esempio algebrici, e poi attribuire il risultato ottenuto alla variabile posta a sinistra del segno di eguale. Facciamo qualche altro esempio:

```
10 M = 2*(3 + 6)           M=18
10 M = (180 + 20)/(10 - 6) M=50
```

Con la freccetta indichiamo, in modo simbolico, il risultato di qualche operazione o il valore assunto da una variabile. L'asterisco * e la barra / (in inglese "slash") che ci sono nelle espressioni degli esempi sono rispettivamente i simboli per la moltiplicazione e la divisione usati in quasi tutti i linguaggi di programmazione. Si potrebbero fare altri esempi con espressioni molto più complicate, ma lasciamo al lettore il divertimento di pensarle e, se possibile, anche di provarle su un calcolatore. In questo momento è importante dire due cose. Per primo va precisato, ma lo abbiamo già detto, che il segno di eguale = usato nelle istruzioni di assegnazione non rappresenta una eguaglianza matematica! Non è quindi possibile scrivere cose del genere:

```
10 45 + 5 = A
```

Le assegnazioni in informatica portano un valore da destra

nella variabile di sinistra. Viceversa non significa nulla. E' anche un errore scrivere eguaglianze del tipo:

```
20 Y + 30 = Z
```

Ripetiamo che questa non è un'assegnazione. Chiarito quindi che le assegnazioni permettono di avere a sinistra del segno = solo il nome di una variabile, diciamo qualcosa di più su cosa si può scrivere in un'espressione posta a destra. Un'espressione può contenere delle variabili che, in tal caso, vengono trattate come fossero dei valori già noti. L'unica limitazione all'uso di variabili nella parte destra di un'istruzione di assegnazione è che devono avere un valore al momento in cui tale istruzione viene eseguita. Facciamo degli esempi:

```
50 A = 50
60 B = A + 100      B = 150
30 C = 20 + 15
40 F = 15 / 3
50 Z = C / F      Z = 7
10 K = 1
20 K = K + 1      K = 2
```

Quest'ultimo esempio è molto significativo poiché presenta un caso molto usato: la creazione di un contatore. Se cerchiamo di leggere passo l'istruzione 20 possiamo dire: somma al contenuto attuale di K il valore 1; siccome K vale 1 si ottiene 2. Ma che cosa succede se l'istruzione 20 dovesse essere nuovamente eseguita? K ora vale 2 per cui il nuovo valore sarebbe 3, e così via. Per ripetere varie volte l'istruzione 20 vi anticipiamo quella che sarà una delle istruzioni di salto che vedremo in seguito GOTO ("go to" in italiano "vai a"). Ecco come modificare l'ultimo programma:

```
10 K = 1
20 K = K + 1
30 GOTO 20
```

L'istruzione 30 dice di "saltare" alla 20 (invece che all'istruzione col numero successivo) per cui queste due istruzioni vengono indefinitamente eseguite e il valore di K cresce continuamente: 1, 2, 3, 4, 5, 6, ecc.. Negli esempi di programmi successivi avremo occasione di chiarire molti altri aspetti delle istruzioni di assegnazione.

VARIABILI

Abbiamo visto qualche semplice esempio di programma BASIC. Facciamo ora un passo avanti e proviamo a studiare programmi che siano in grado di svolgere un determinato compito.

Forse qualche punto di questi programmi non risulterà molto chiaro essendo solo all'inizio. Non preoccupatevi troppo, ma cerca di comprendere gli aspetti fondamentali di quanto spiegheremo. Vedere e possibilmente provare molti esempi pratici è di grande aiuto nell'apprendimento, di un linguaggio di programmazione, anche se all'inizio non se ne comprendono tutti i dettagli. Il Programma 2, che possiamo chiamare "programma per la stampa di biglietti da visita", è un'esempio d'impiego delle istruzioni INPUT e PRINT. Ricordiamo che con "stampa" si intende spesso dire "visualizzazione sullo schermo video", come appunto in questo caso. Solo se disponete di una periferica stampante potete effettivamente stampare i risultati di un programma. A parte questo chiarimento sui termini, vediamo cosa fa il Programma 2.

Programma 2

```
10 PRINT "PROGRAMMA PER LA STAMPA DEL"
20 PRINT "      VOSTRO"
30 PRINT "BIGLIETTO DA VISITA."
40 PRINT
50 INPUT "VOSTRO NOME: ";NOME$
60 INPUT "VOSTRO COGNOME: ";COGNOME$
70 INPUT "VIA: ";V$
80 INPUT "NUMERO: ";N
90 INPUT "CITTA': ";C$
100 PRINT
110 PRINT "ECCO IL VOSTRO BIGLIETTO DA VISITA: "
120 PRINT
130 PRINT "-----"
140 PRINT "SIG. ";NOME$;" ";COGNOME$
150 PRINT
160 PRINT "VIA ";V$;" ";N
170 PRINT C$
180 PRINT "-----"
190 GOTO 40
200 END
```

Prova di esecuzione

```
PROGRAMMA PER LA STAMPA DEL
VOSTRO BIGLIETTO DA VISITA
```

```
VOSTRO NOME: ? SERGIO
VOSTRO COGNOME: ? ROSSI
VIA: ? CARPACCI
NUMERO: ? 12
CITTA': ? MILANO
```

Alle linee 10, 20 e 30 vi sono tre istruzioni PRINT per la visualizzazione del titolo del programma. La PRINT della linea 40 serve solo per andare a capo.

BASIC (Parte 4)

Alle linee da 50 a 90 vi sono le istruzioni INPUT per chiedere i vostri dati personali. Ognuna di queste istruzioni contiene un messaggio per indicarvi, di volta in volta, quale dato fornire. Per esempio la prima dice "VOSTRO NOME:" per chiedervi il nome. La parola NOME\$ che compare dopo è una variabile a cui verrà effettivamente associato il vostro nome durante l'esecuzione del programma. Alla linea 100 vi è un'altra PRINT per andare a capo (sullo schermo). Dalla linea 130 alla 180 vi sono le istruzioni di "stampa" il biglietto. Dalla linea 190 c'è un'istruzione di cui parleremo molto in seguito, un'istruzione di salto incondizionato: GOTO. Per ora, diciamo che serve per far ripetere il programma a partire dalla linea 40.

Se invece di GOTO 40 avessimo posto GOTO 120 il programma avrebbe ripetuto le istruzioni di stampa senza richiedere ogni volta i dati, cioè avrebbe ristampato di continuo lo stesso biglietto da visita (provare per credere). Stampare il contenuto di una variabile vuol dire copiarlo sul video: il contenuto originale non va perso. Per seguire il Programma 2 diamo il comando RUN (battendo sulla tastiera R, U, N e poi CR) che, ricordiamolo, ordina al calcolatore di eseguire il programma scritto nella memoria centrale.

Quando appare "VOSTRO NOME:" scriviamo un nome (e poi CR) e così pure rispondiamo a tutte le altre domande (istruzioni INPUT). Le istruzioni PRINT fanno allora apparire i dati in uscita. Certamente, visualizzare dei biglietti da visita sullo schermo non è molto utile, ma per stamparli su carta basta sostituire le istruzioni PRINT ("stampa sul video") con quelle proprie della stampante. In alcuni dialetti BASIC queste istruzioni sono simili alla PRINT e si chiamano LPRINT. La "L" davanti a PRINT ricorda la parola "line printer" che in inglese indica un tipo di stampante. Su altre macchine la cosa è più complessa; vedremo in seguito, in dettaglio, le istruzioni per le stampanti.

Il Programma 3 è invece un esempio di come si possa disegnare con il calcolatore senza disporre di periferiche grafiche o accessori sofisticati. Il Programma 3 disegna una scacchiera. Niente di eccezionale, ma ci sono alcune note interessanti da fare. Alle linee 10 e 20 vengono fatte due assegnazioni. Notate bene che queste assegnazioni non sono di tipo "stringa", o "di caratteri". Nel nostro caso le stringhe sono costituite da una successione di due asterischi e di due spazi.

Quando parleremo delle variabili, vedremo come i dati elaborabili (trattabili) dai calcolatori non sono solo i numeri della matematica, ma anche insieme di caratteri, ad esempio il nome di una persona. Questi si chiamano stringhe. Il Programma 2 ne è un primo esempio.

Programma 3

```
10 L1$ = " ** ** ** ** ** ** ** **"  
20 L2$ = " ** ** ** ** ** ** ** **"
```

```
30 PRINT L1$  
40 PRINT L1$  
50 PRINT L2$  
60 PRINT L2$  
70 PRINT L1$  
80 PRINT L1$  
90 PRINT L2$  
100 PRINT L2$  
110 PRINT L1$  
120 PRINT L1$  
130 PRINT L2$  
140 PRINT L2$  
150 PRINT L1$  
160 PRINT L1$  
170 PRINT L2$  
180 PRINT L2$  
190 PRINT L1$  
200 PRINT L1$  
210 PRINT L2$  
220 PRINT L2$  
230 END
```

Prova di esecuzione

```
** ** ** ** ** ** ** **  
** ** ** ** ** ** ** **  
  ** ** ** ** ** ** ** **  
  ** ** ** ** ** ** ** **  
** ** ** ** ** ** ** **  
** ** **^**^**^**^**^  
  ** **^**^**^**^**^  
  ** **^**^**^**^**^  
** ** **^**^**^**^**^  
** ** **^**^**^**^**^  
  ** **^**^**^**^**^  
  ** **^**^**^**^**^  
** ** **^**^**^**^**^  
** ** **^**^**^**^**^  
  ** **^**^**^**^**^  
  ** **^**^**^**^**^  
** ** **^**^**^**^**^  
** ** **^**^**^**^**^  
  ** **^**^**^**^**^  
  ** **^**^**^**^**^
```

In BASIC le stringhe sono delimitate normalmente da virgolette ("") poste all'inizio ed alla fine di un gruppo di caratteri. Se guardate bene con attenzione alle linee 10 e 20 del Programma 3, a destra del segno di eguale (=), vedrete appunto le virgolette che delimitano due stringhe fatte di asterischi e di spazi. Abbiamo scelto questi asterischi solo per rendere più uniforme la scacchiera, ma potevamo prendere qualunque altro carattere.

Per costruire la scacchiera non facciamo altro che visualizzare due volte la stringa del primo tipo L1\$ e poi quella secondo tipo L2\$. L1\$ e L2\$ sono due variabili stringa, cioè variabili che invece di numeri contengono stringhe di caratteri. Notate che il nome di queste variabili termina con il carattere \$ (dollaro).

Notate anche che lo spazio è un carattere come tutti gli altri. Pensiamo che il Programma 3 sia molto facile da capire e basti osservarlo. Provate a modificarlo o a ideare altri programmi che disegnino, per esempio, una cornice o una quadrettatura sullo schermo. I programmi visti finora sono molto semplici e permettono di effettuare solo delle operazioni di ingresso dei dati (con INPUT) nella memoria centrale, di stampa degli stessi (con PRINT) sullo schermo. Abbiamo accennato a come sia possibile inviare i risultati di un programma ad una stampante. I dati possono anche essere memorizzati (scritti) su memorie esterne (dischetti o cassette magnetiche), da dove poi possono essere nuovamente prelevati e riportati nel computer. La registrazione di dati su una memoria esterna, cioè la scrittura di file, è uno degli aspetti più interessanti ed utili della programmazione, ma prima di affrontarlo è meglio avere un po' di dimestichezza col BASIC in generale.

E' utile invece imparare subito a registrare il programma su una memoria esterna, in modo che possa essere conservato. La memoria centrale di un calcolatore perde infatti ogni contenuto nel momento in cui la macchina viene spenta (si dice che è una "memoria volatile"). E' necessario quindi registrare un programma su una memoria esterna non volatile, ad esempio un dischetto magnetico, per poterlo conservare. Per effettuare questa operazione si usa l'istruzione SAVE descritta in dettaglio nel relativo riquadro. Poiché l'uso di SAVE non è uguale in tutte le macchine, vi consigliamo di consultare il manuale del computer che state usando. Il trasferimento inverso del programma, dalla memoria esterna a quella centrale, si effettua con l'istruzione LOAD (in alcune versioni BASIC si usa l'istruzione OLD che è equivalente a LOAD).

Sul supporto su cui si vuole salvare il programma (cassetta, dischetto o altro) non deve esistere un altro programma con lo stesso nome. In caso contrario, è probabile che il vecchio programma vada perso e venga sostituito con quello nuovo (alcuni sistemi operativi chiedono prima una conferma). Questa operazione viene spesso eseguita per aggiornare la registrazione di un programma, registrando al suo posto una versione più recente dello stesso. E' una pessima pratica, perché in caso di errore fisico di scrittura, o di caduta di rete (mancanza di corrente), si perdono entrambi i programmi. E' molto meglio registrare i programmi con i numeri progressivi, conservando le ultime due o tre versioni i cancellando quelle vecchie. Per poter essere eseguito da un computer, un programma BASIC deve risiedere completamente nella memoria centrale. Questo significa che non

può essere più lungo, in numero di byte, dell'area di memoria disponibile per i programmi. Poiché alcuni computer sono in vendita con diverse dimensioni (configurazioni) di memoria, non è detto che un programma registrato da una macchina sia sempre usabile su un'altra: può non esserci memoria a sufficienza (ricordiamo che le memorie centrali si misurano in Kbyte: 16, 32, 48, 64, 128 Kbyte...).

Per esempio, un programma molto lungo scritto su un Sinclair Spectrum da 48K (48 Kbyte) può non funzionare su uno Spectrum da 16K. In mancanza di informazioni, si può provare a caricare, in ogni caso, il programma e vedere che non venga segnalato alcun errore del tipo "Out of memory" (memoria insufficiente) o simili, e che il programma poi funzioni regolarmente.

Può essere utile, dopo un'operazione di LOAD, fare il listato del programma (LIST) per controllare che sia proprio quello voluto, prima di dare il comando RUN, se si desidera eseguirlo. Il comando LOAD, prima di cominciare a caricare il nuovo programma, cancella completamente ogni altro programma e/o dato presente in memoria (centrale). Prima di dare un comando LOAD, assicurarsi quindi che l'eventuale programma presente in memoria sia stato salvato con SAVE, o possa comunque essere cancellato. In alcune versioni BASIC esistono speciali comandi, come CHAIN o MERGE, che permettono di sovrapporre o unire due programmi senza cancellare completamente il precedente ed i suoi dati (variabili).

Questa possibilità si utilizza quando un programma è molto lungo e non è caricabile interamente in memoria. Lo si fraziona allora in due o più parti, tra cui esista una connessione logica, e queste si caricano e si eseguono una alla volta. La prima viene caricata con LOAD, le successive con CHAIN o MERGE.

Un'altra istruzione che abbiamo già visto è quella di assegnazione, che in BASIC si chiama LET. L'istruzione di assegnazione, che esiste in tutti i linguaggi di programmazione, permette di assegnare (attribuire) un certo valore ad una variabile. Non abbiamo ancora detto cosa siano le variabili in un linguaggio di programmazione, anche se le abbiamo già usate negli esempi. Se ricordate le variabili imparate a scuola durante le lezioni di matematica, possiamo allora dire che in informatica sono pressappoco la stessa cosa.

Una variabile è come una scatola che può contenere qualcosa (in BASIC, un numero od una stringa). Più propriamente, è un nome simbolico per indicare una grandezza a cui può essere assegnato un valore effettivo.

Forse un semplice esempio può chiarire alcuni dubbi. Supponiamo di dover riempire un modulo per una domanda di assunzione in un'azienda.

Quasi tutti questi moduli iniziano richiedendo i dati anagrafici: cognome, nome, anno di nascita, ecc. Vicino ad ognuno si trovano dei puntini per indicare dove dobbiamo scrivere i nostri dati personali:

Cognome..... Rossi
Nome..... Sergio
Anno di nascita..... 1958

“Cognome”, “Nome”, e “Anno di nascita” sono variabili a cui chi scrive attribuisce dei valori effettivi (nel nostro caso “Rossi”, “Sergio” e “1958”). Tutte le domande di assunzione portano indicate le stesse variabili, che ognuno “riempie” con i propri dati.

Nel Programma 2 le variabili erano NOME\$, COGNOME\$, V\$, N e C\$. Si potrebbe notare che le variabili imparate a scuola trattavano “numeri”, mentre in questo caso “Rossi” e “Sergio” non sono numeri. E' vero: le variabili usate in informatica sono un concetto più ampio di quello algebrico. In BASIC infatti esistono due tipi di variabili: quelle numeriche e quelle di stringa. I dati che si possono attribuire alle variabili numeriche sono dei numeri, mentre per quelle dette di stringa sono, appunto, delle stringhe. Che cosa siano con precisione le stringhe, o per meglio dire le stringhe di caratteri, lo vedremo più avanti.

Per ora limitiamoci a dire che i dati “Rossi” e “Sergio” sono due esempi di stringhe, come stringhe sono gli insiemi di spazi ed asterischi delle istruzioni 10 e 20 del Programma 3. Una nota importante riguardo alle variabili: in molti dialetti BASIC, solo le prime due lettere del nome di una variabile sono significative, cioè contano per distinguerla dalle altre. Le lettere successive possono essere impiegate per comodità, ma non hanno importanza. Occorre perciò fare attenzione a non usare nomi con le prime due lettere uguali, per esempio, MINIMO e MINUTI o COGNOME\$ e CODICE\$.

ESPRESSIONI NUMERICHE

Le espressioni matematiche, in BASIC, sono praticamente identiche a quelle che si studiano a scuola, con alcune piccole differenze nella scrittura degli operatori (i simboli delle operazioni).

Prendiamo un'espressione algebrica, scritta come a scuola, e proviamo a riscriverla in BASIC. Per fare questo, dobbiamo usare i simboli delle operazioni algebriche (operatori algebrici) accettati dal BASIC. Come si vede le operazioni di “somma” e di “sottrazione” si indicano come al solito con “+” e “-”. Il prodotto viene invece indicato con l'asterisco “*” e non con “x” o il punto come si usa in matematica. Spesso in matematica il prodotto di due valori si indica semplicemente scrivendoli uno dopo l'altro; così “a b” vuol dire “a per b”.

In informatica, invece, nulla è mai sottinteso, per cui il prodotto deve essere espressamente indicato con l'asterisco: A * B (notare che in BASIC tutto va in lettere maiuscole).

La divisione è indicata con il carattere “/” (barra, in inglese slash). I due punti “:” hanno in BASIC un significato completamente diverso e non costante in tutti i dialetti.

L'elevamento a potenza è un'altra operazione con un simbolo particolare. Non si scrive l'esponente come un numero più piccolo posto in alto a destra della base come “4-”. Si scrivono invece base ed esponente sulla stessa riga, separati da uno dei simboli “**” o “^” o “^”; cioè, per il nostro esempio, 42 o 4**2 o ancora, 4^2.

Per quanto riguarda le parentesi, le uniche che si usano nelle espressioni sono quelle rotonde, che funzionano esattamente come in matematica. Le parentesi quadre e graffe sono usate in questo testo e in alcuni manuali, ma solo per descrivere le istruzioni del linguaggio di programmazione. Non fanno parte del linguaggio BASIC (sono usate invece in altri linguaggi, come PASCAL e C).

Altre funzioni matematiche che si trovano normalmente pronte in BASIC sono l'esponenziale EXP(X) e la sua funzione inversa LOG(X) che è il logaritmo in base “e”, le funzioni trigonometriche seno, coseno e tangente, rispettivamente SIN(X), COS(X) e TAN(X).

Di queste ultime bisogna ricordare che si riferiscono ad angoli in radianti (20 radianti = 360 gradi). Dato un angolo espresso in gradi, per convertirne il valore in radianti basta moltiplicarlo per:

$$X_{\text{rad}} = x * (3.1416/180)$$

ALCUNI PROGRAMMI

Vediamo ora un programma in cui vengono calcolate alcune espressioni algebriche.

La prima serve per il calcolo del volume di un parallelepipedo e la seconda per quello di una sfera. Il Programma 4 chiede le dimensioni del parallelepipedo, l'altezza, lunghezza e larghezza, poi alla linea 80 le moltiplica tra di loro per calcolare il volume e assegna il risultato alla variabile V. Successivamente chiede il raggio di una sfera e poi ne calcola il volume alla linea 140. Notate che viene usato lo stesso nome di variabile (V) sia per il volume del parallelepipedo che per quello della sfera. Questo è possibile purché si usi la variabile V correttamente.

Vediamo di chiarire questo punto. Inizialmente V non contiene niente (contiene zero), poi alla linea 80 le viene assegnato il valore del volume del parallelepipedo. Subito dopo, alla linea 100, viene visualizzato con PRINT V il suo contenuto (volume parallelepipedo).

In un secondo tempo, alla linea 140 viene calcolato il volume della sfera che viene assegnato alla stessa variabile V: da questo momento V contiene solo il volume della sfera e quello del parallelepipedo è stato cancellato. In effetti a noi interessa ora il volume della sfera che viene infatti visualizzato con PRINT V (linea 160).

Ovviamente, se avessimo avuto bisogno di conservare sia il valore del primo che del secondo volume avremmo dovuto usare due variabili diverse, per esempio VP e VS.

BASIC (Parte 4)

Questo esempio conferma che le variabili in informatica si comportano come scatole in cui si può mettere, di volta in volta, ciò che si vuole.

L'ultimo oggetto (numero) introdotto "caccia via" quello che c'era prima.

Programma 4

```
10 PRINT "PROGRAMMA PER IL CALCOLO DI VOLUMI"
20 REM -----
30 PRINT
40 PRINT "CALCOLO DEL VOLUME DI UN PARALLELE-
PIPEDO"
50 INPUT "ALTEZZA ";H
60 INPUT "LUNGHEZZA ";LG
70 INPUT "LARGHEZZA ";LA
80 V=LA*LG*H
90 PRINT "IL VOLUME E' : ";
100 PRINT V
110 PRINT
120 PRINT "CALCOLO DEL VOLUME DI UNA SFERA "
130 INPUT "RAGGIO DELLA SFERA ";R
140 V=3. 1416*(4/2)*R^2
150 PRINT "IL VOLUME DELLA SFERA E' : ";
160 PRINT V
170 END
```

Prova di esecuzione

```
PROGRAMMA PER IL CALCOLO DI VOLUMI

CALCOLO DEL VOLUME DI UN PARALLELEPIPEDO
ALTEZZA ? 5
LUNGHEZZA ? 6
LARGHEZZA ? 8
IL VOLUME E' : 240

CALCOLO DEL VOLUME DI UNA SFERA
RAGGIO DELLA SFERA ? 5
IL VOLUME DELLA SFERA E' : 523. 6
```

Il Programma 5 è un esempio di come si possa tenere una piccola contabilità e calcolare dei valori percentuali.

Supponiamo di aver fatto un viaggio e voler sapere come si sono ripartite le spese, in percentuale, tra automobile, hotel e ristoranti.

Dopo avere ricevuto i dati in ingresso, alla linea 90 viene calcolato il totale delle spese, che viene conservato nella variabile T.

Alle linee 100, 110 e 120 vengono calcolate le tre percentuali e assegnate rispettivamente alle variabili APERC, HPERC e RPERC.

Programma 5

```
10 REM PROGRAMMA PER IL CALCOLO
20 REM DEI VALORI PERCENTUALI DI
30 REM SPESA RISPETTO AL TOTALE
40 REM IN OCCASIONE DI UN VIAGGIO
50 REM -----
60 INPUT "SPESE AUTO ";A
70 INPUT "SPESE HOTEL ";H
80 INPUT "SPESE RISTOR. ";R
90 T=A+H+R
100 APERC=A*(100/T)
110 HPERC=H*(100/T)
120 RPERC=R*(100/T)
130 REM
140 PRINT "SPESE AUTO IN PERC. ",APERC
150 PRINT "SPESE HOTEL IN PERC. ",HPERC
160 PRINT "SPESE RISTOR. IN PERC. ",RPERC
170 END
```

Prova di esecuzione

```
SPESE AUTO ? 20000
SPESE HOTEL ? 120000
SPESE RISTOR. ? 35000
SPESE AUTO IN PERC.          11. 4286
SPESE HOTEL IN PERC.        68. 5714
SPESE RISTOR. IN PERC.      20
```

Quest'ultimo esempio sembra banale e ci si può chiedere perché usare un calcolatore, o scrivere un programma, per calcoli più facilmente eseguibili con una calcolatrice tascabile.

Ma naturalmente questi sono soltanto esempi, che devono essere il più possibile semplici.

Quello che conta, per ora, è il principio. Un programma effettivo potrebbe essere impostato più o meno nello stesso modo e lavorare non su tre voci, ma su centinaia, e ricevere i dati non a mano dalla tastiera, ma da un archivio su disco.

E da ultimo potrebbe anche tracciare un "grafico" della ripartizione delle spese o estrapolare (predire sulla base dei dati precedenti) previsioni per gli anni successivi.

Programmi di questo genere, fatti per immagazzinare e successivamente rivedere, elaborare, presentare grandi quantità di dati, si chiamano "programmi di data base", o semplicemente "data base".

Fine Quarta Parte

DISPOSITIVO A DUE STADI ON & OFF

Per meglio capire cosa si intenda quando si parla di un dispositivo a due stadi (on-off) sarà meglio ricorrere ad un esempio.

Immaginiamo una lampadina in un certo istante: essa può essere accesa oppure spenta.

Ogni calcolatore è basato su questo principio, ogni suo componente, in un dato istante può essere o non essere attraversato da una carica elettrica.

I due stadi, sono corrisposti alivello fisico da due tensioni differenti come potrebbero essere 0 Volt e 5 Volt oppure una tensione negativa ed una positiva ecc., ecc. I due stadi sono invece corrisposti a livello logico dalle cifre 0 e 1 appartenenti al sistema binario, sistema di numerazione che si serve esclusivamente delle cifre 0 e 1 per rappresentare tutti i numeri del sistema decimale.

Il binario dà la possibilità di rappresentare in modo molto comodo la situazione interna dei componenti elettronici del calcolatore, quindi per meglio assimilare il suddetto concetto, potremmo dire che al numero binario 1 corrisponde un passaggio di corrente mentre al numero binario 0 non corrisponde un passaggio di corrente.

Questi singoli numeri si chiamano BIT, dall'inglese "Binary digIT".

COME RAGIONA UN CALCOLATORE E PERCHÉ ?

Se non vi siete mai posti questa domanda, questo è il momento per farlo.

Infatti il modo migliore per capire come è architettato un calcolatore è quello di vedere come lavora ossia come pensa.

Desiderando risolvere un problema molto banale, come quello di riconoscere il nome Claudio in un elenco di altri dieci nomi, si consideri dapprima di assegnare questo problema ad un essere umano.

Per risolverlo, è necessario fornire all'uomo due informazioni : un'istruzione (trova il nome Claudio) e l'elenco dei nomi.

Usando in modo accorto esclusivamente occhi e cervello, l'uomo individuerà rapidamente il nome desiderato.

Anche il calcolatore, per risolvere il problema necessita di due elementi:

- 1) l'insieme delle istruzioni (che ora chiameremo programma)
- 2) la lista dei nomi (dati).

La differenza tra l'istruzione per il ragazzo e quelle per il calcolatore è che quest'ultimo necessita di più precise e numerose istruzioni.

Il programma consiste infatti in una serie di istruzioni che operino su dei dati specifici.

L'uomo per ricordare dei dati fa ricorso alla propria memoria, e analogamente deve fare il computer.

Esso possiede infatti una memoria dove poter immagazzinare i dati ma non per poter svolgere calcoli.

Per svolgere i calcoli infatti vi è, diciamo, un reparto chiamato unità aritmetica.

Siccome in questo reparto possono essere svolte anche operazioni logiche (ad esempio confrontare due numeri) il suo nome completo è "ALU" unità aritmetico-logica.

Per spostarsi dall'ALU alla memoria, i dati fanno uso del BUS DATI e del BUS INDIRIZZI assimilabili per semplicità e dei veri e propri autobus adibiti al trasporto dei dati e dei loro indirizzi di memoria.

Tutto ciò è controllato dalla cosiddetta "UC", unità centrale, che è responsabile del movimento dei dati verso l'ALU, delle decisioni relative a quali operazioni aritmetiche o di confronto eseguire e del ritorno dei risultati in memoria centrale.

Tutti questi sono gli unici componenti presenti all'interno di ogni computer che nell'insieme vengono raccolti nella "CPU" ossia Unità Centrale di Processo.

I DISPOSITIVI DI INPUT/OUTPUT

Un dispositivo di ingresso di un calcolatore ha lo sco-

po di permettere l'introduzione delle informazioni e dei dati nella memoria centrale.

Per fare ciò, il dispositivo di ingresso converte i caratteri alfanumerici (lettere, numeri, simboli e punteggiatura) in formato binario in modo tale che possano essere contenuti nella memoria centrale.

Un dispositivo di uscita ha praticamente lo scopo inverso, ovvero traduce le informazioni immagazzinate in memoria in formato binario, in un formato leggibile dall'uomo. Vi sono attualmente numerosi tipi di dispositivi di ingresso/uscita: Lettori di schede perforate, tastiere, stampanti, monitors, mouses, disk drives, etc. Questi dispositivi vengono tutti controllati dall'unità centrale ma siccome non fanno parte della CPU (Unità Centrale di Processo) vengono anche chiamati periferiche del calcolatore.

LA PROGRAMMAZIONE: CONCETTI BASILARI

Ora che a grandi linee siamo a conoscenza di come funziona un calcolatore, possiamo parlare di programmazione o meglio dei concetti elementari e basilari indispensabili per essa.

Per prima cosa si sa che lo scopo di un programma è quello di risolvere un problema.

Risolvere un problema significa ricercare un procedimento risolutivo (d'ora in poi chiamato anche algoritmo) che, eseguito fornisca delle informazioni finali (risultati, dati di uscita) dipendenti da alcune informazioni iniziali (dati di ingresso).

Queste informazioni iniziali e finali devono essere sempre codificabili mediante sequenze finite di simboli dette stringhe (caratteri) e di numeri naturali detti dati.

E' bene chiarire bene la differenza sostanziale fra dati e informazioni: i primi non costituiscono di per sé informazione, ma la rappresentano quando riescono a portare qualche nuova conoscenza a chi riceve questi dati (in questo caso l'esecutore che attende i dati iniziali da una parte e l'utente che aspetta di vedere i risultati dall'altra).

Chiarita quindi questa differenza, parleremo genericamente di «dati» distinguendo, se necessario, fra dati di ingresso, dati intermedi e dati di uscita.

I dati intermedi sono quelli che possono essere creati

o trasformati dall'esecutore della procedura risolutiva durante il calcolo, cioè durante l'esecuzione della procedura risolutiva o algoritmo.

USO INTERATTIVO E NON INTERATTIVO DI UN SISTEMA DI CALCOLO

Come abbiamo già detto, esiste una figura chiamata utente che introduce i dati nel sistema di calcolo e attende i risultati.

Il suddetto utente può lavorare in due modi:

- A) modo non interattivo:
l'utente inserisce i dati di ingresso tutti in una volta e preleva i risultati tutti in una volta.
- B) modo interattivo:
l'utente introduce una parte dei dati, attende un primo risultato, in base al quale decide quale nuovo ingresso fornire e così via fino al termine del colloquio: esiste quindi una interazione fra l'utente e la macchina.

Quest'ultima soluzione è ormai permessa da tutti i nuovi sistemi di calcolo in commercio ed è, per fare un semplice esempio quella utilizzata in tutti i videogames. I sistemi interattivi sono sicuramente preferibili per la loro notevole immediatezza nelle risposte e sono gli unici possibili in quelle applicazioni in cui il tempo ha un ruolo determinante (applicazioni in tempo reale).

Per meglio capire cosa si intenda per applicazione in tempo reale sarà opportuno fare un esempio: Immaginiamo un sistema di calcolo che con appropriate apparecchiature elettriche debba sondare le radiazioni presenti nei pressi di una centrale nucleare.

E' ovvio che la risposta del calcolatore debba essere immediata e ripetuta ogni frazione di secondo per evitare un eventuale emergenza nucleare.

Dovendo giocare a scacchi con un sistema interattivo è sufficiente che l'utente comunichi di volta in volta la propria mossa perché il sistema conosce già lo stato della scacchiera e le informazioni riguardanti la propria strategia.

FINE SECONDA PARTE

TELEMATICA ED EDUCAZIONE

Le risorse tecnologiche dei mezzi di comunicazione telematica investono direttamente i processi educativi, proprio perché innovano le modalità della comunicazione modificando i processi di apprendimento e conseguentemente, le procedure di insegnamento.

La comunicazione mediata da questi strumenti diventa, marcatamente, da individuale, collettiva e viene percepita in contemporanea da un numero di persone che moltiplicano N volte l'unità classe.

L'unità classe tradizionale si frantuma e si riaggrega in nuove composizioni in base ai livelli di comprensione raggiunti. Il rapporto comunicativo si decanta da ogni interferenza di origine emotiva che lo rende indubbiamente meno nitido e chiaro, ma certamente più partecipato.

L'allievo può dare segnali di risposta certi e chiari che consentono la verifica dell'esatta comprensione del messaggio e questo fatto permette successivi interventi di correzione della comunicazione tali da renderla chiara ed efficace per il destinatario.

Risultano altresì possibili ulteriori arricchimenti di quanto è oggetto della comunicazione con una grafualità controllabile e funzionale all'apprendimento.

Tuttavia non tutti i messaggi in uscita e in entrata, di cui si compone il processo educativo, sono efficacemente comunicabili attraverso i canali telematici, poiché tale processo non è riducibile ad una pura comunicazione di contenuti, ma esso nella complessità degli interventi è costituito anche da segno che trasmettono valutazioni comportamentali, anche di ordine etico, di funzioni estetiche, di moduli partecipativi emozionali, soggettivi e interpersonali che superano l'ambito della comunicazione oggettiva.

Un primo problema è quindi di individuare gli ambiti in cui sia opportuno utilizzare i canali telematici; in tutti quei casi in cui la comunicazione richiede chiarezza e oggettività e sia di alto valore scientifico, il ricorso a tali canali è utile, opportuno e potrà divenire anche indispensabile.

Un secondo problema è quello di prefigurare una comunicazione in contemporanea di informazioni uguali per tutti gli studenti e di alto valore scientifico, e verificarne l'esatta

comprensione; sembra di dover ipotizzare presenze di mediazione da parte di docenti che, dopo aver percepito il messaggio con gli allievi, guidino un comune lavoro di rielaborazione al fine di favorire il processo di interiorizzazione critica e di osservare e valutare le procedure di apprendimento e verificare l'efficacia dell'insegnamento.

Un'operazione di tal genere presuppone docenti preparati all'uso della risorsa tecnica, capaci di cogliere le incidenze psicologiche del mezzo di comunicazione e di correlarle tra loro per coordinare e finalizzare gli interventi nel processo educativo.

Questo cenno ai docenti ed alla loro funzione ci riconduce ad un discorso che è prioritario e preliminare in qualsivoglia processo educazionale, quello dei fini e degli obiettivi di tale processo, senza di che gli interventi, rimanendo del tutto occasionali e casuali, non hanno alcun senso educativo.

L'impiego in tutti i campi del lavoro di risorse tecnologiche che, consentendo la comunicazione in tempi reali, accelerano il cambiamento, impone dall'esterno alcune finalizzazioni dell'attività curricolare scolastica.

Il "sapere" è sempre meno riducibile ad una costruzione statica edificata attraverso una sovrapposizione di elementi definiti e bene incastrati tra loro.

Ma è sempre più chiaramente un processo in cui gli elementi base acquisiti consentono l'inserimento di elementi nuovi correttamente compresi e criticamente valutati.

Né può essere definito sapere la pura acquisizione di conoscenze scientifiche e di abilità tecnologiche accettate come se fossero dati determinati, certi ed immutabili.

Atteggiamenti culturali del genere condannerebbero l'allievo divenuto lavoratore, ma prima ancora il sistema educativo stesso nel suo complesso, ad una rapida obsolescenza.

In realtà è quanto sta avvenendo nei diversi settori del lavoro, ma nella crisi sono pienamente coinvolte tutte le agenzie educative, fra cui anche la scuola che viene richiamata ad una rigorosa revisione dei suoi interventi nel processo educazionale.

DENTRO L' MSX (Parte Seconda)

IL PROCESSORE VIDEO

Nell' Msx 1 il VDP (ViDeo Processor), è un Texas Instrument TMS 9129A. Si tratta di un integrato a 40 pin standard "dual in line" (DILIP) che usa 16Kb di RAM dinamica.

Solo il VDP può accedere alla RAM dedicata al video usando un bus dati unidirezionale a 8 bit.

Il VDP viene controllato usando tre linee:

RAS, CAS e R/W

che derivano dal bus indirizzi e dalle linee di controllo del processore.

Il VDP è inoltre connesso al bus dati di sistema.

Per effettuare da BASIC un'operazione di lettura o di scrittura sulla RAM del video è necessario usare le istruzioni VPOKE e VPEEK oppure le appropriate procedure in ROM a livello di linguaggio Assembler.

LA STRUTTURA VIDEO DEL VDP

In tutti i modi di funzionamento il VDP costruisce l'immagine da visualizzare principalmente a partire da due tabelle nella RAM video: la tabella generatrice dei modelli e la tabella dei nomi e dei modelli.

Nella tabella generatrice dei modelli sono contenute le definizioni di ciascun carattere visualizzabile (cioè la descrizione, in termini di posizione dei pixel dei vari caratteri).

L'indirizzo di partenza della tabella generatrice dei modelli in ognuna della modalità di visualizzazioni è la seguente:

Modalità 0 - TEXT 40 - Indirizzo: 2048

Modalità 1 - TEXT 32 - Indirizzo: 0

Modalità 2 - HRG - Indirizzo: 0

Modalità 3 - MULTICOLORE - Indirizzo: 0

In tutte le modalità, tranne quella multicolore, ciascun carattere è definito nella tabella generatrice dei modelli per mezzo di otto byte con cui viene specificata la geometria dei pixel - tutti i caratteri standard sono inoltre giustificati a sinistra con le due colonne più a destra e la riga di base lasciate vuote.

Nella modalità testo a 32 colonne possono essere usate

fino a 256 definizioni di carattere, nel qual caso la tabella risulta lunga 2048 byte.

La modalità HRG consente la definizione di 768 caratteri, producendo in tal modo una tabella lunga 6144 byte. La modalità testo a 40 colonne consente la definizione di 256 caratteri, ognuno di 8 byte, ma non vengono visualizzate le due colonne più a destra di ciascun carattere. In questo modo è possibile ottenere una schermata di 40 colonne con delle matrici di carattere di 8 x 6 pixel e non di 8 x 8 come nella modalità HRG o testo a 32 colonne.

Nella modalità multicolore la tabella generatrice dei modelli contiene 192 entrate, ciascuna di otto byte, formate da quattro coppie di due byte ciascuna. Ogni coppia non definisce il carattere bensì il colore delle celle 4 x 4 nella matrice complessiva di 8 x 8.

ESEMPIO

Nella modalità testo a 32 colonne la tabella generatrice inizia all'indirizzo 0 della VRAM, quindi il carattere con codice 65 - una "A" - viene definito dalle locazioni 65 x 8 = 520 fino alla 527.

La tabella dei nomi dei modelli determina quale carattere compare in ogni posizione sullo schermo (nella modalità multicolore vengono determinati i colori).

Nella modalità testo a 32 colonne ci sono 24 righe di 32 colonne e quindi 32 x 24 = 768 posizioni. Ciascuna di queste può contenere uno dei 256 possibili caratteri e di conseguenza la tabella dei nomi è lunga 768 byte - ognuno dei quali specifica quale carattere verrà messo in corrispondenza con una posizione.

Nella modalità testo a 40 colonne la tavola dei nomi per lo schermo ha 40 x 24 = 960 byte. Analogamente, ciascuno di questi descrive il carattere visualizzato in una determinata posizione dello schermo. Così come nella modalità testo a 32 colonne, la scelta è ristretta ai 256 modelli contenuti nella tabella generatrice dei modelli.

GRAFICA AD ALTA RISOLUZIONE

Nella modalità HRG ci sono ancora una volta 768 posizioni nella tabella dei nomi, ma poiché in questa modalità sono disponibili 768 definizioni di carattere, come può un byte stabilire la selezione? A tale scopo lo schermo è concettualmente diviso in tre zone e le prime 256 entrate nella tabella dei nomi selezionano i caratteri da 0 a 255, le seconde 256 selezionano i caratteri da 256 a 511 e le ultime 256 i caratteri da 512 a 768.

La tabella dei nomi nella modalità multicolore è lunga 768

bytes. Il valore contenuto in ciascun byte è un riferimento a un blocco di otto byte nella tabella generatrice dei modelli per il multicolore.

Due bytes di questo blocco indicano i colori dei blocchi 4 x 4 che formano la matrice del carattere. La colonna sullo schermo ove si trova la matrice carattere determina a quale dei quattro blocchi si fa riferimento. Se si trova sulla riga più alta vengono referenziati i primi due bytes, se si trova sulla seconda riga sono interessati i bytes 3 e 4 e così via, con la coppia successiva usata per la riga seguente.

Quindi le righe 0, 4, 8, 12, 16, 20 faranno tutte riferimento ai primi due bytes.

COLORE

Benché il TMS 9129 VDP sia in grado di usare 15 colori più il trasparente (compresi il bianco e il nero), nella modalità testo a 40 colonne possono essere usati solo due colori, e il bordo assume il colore dello sfondo. In ambiente BASIC questi sono impostati (come per le modalità) per mezzo dell'istruzione COLOR, mentre in linguaggio macchina il registro a sola scrittura numero 7 del VDP determina entrambi i suddetti colori.

Per modificare quest'ultimo è preferibile usare la procedura appropriata contenuta nella ROM.

Nella modalità testo a 32 colonne, la tabella generatrice dei modelli, con dimensione 2k, è divisa in più insiemi di otto definizioni di carattere.

Per ciascun insieme, il colore del primo piano e quello dello sfondo; sono determinati da un solo byte in una tabella generatrice di colori.

Quindi la tabella dei colori è lunga 32 byte.

Il colore del primo piano è contenuto nel semi byte più significativo e viceversa per il colore dello sfondo.

Ne consegue che per visualizzare una lettera due volte con differenti colori, la definizione del carattere deve essere duplicata in un'altra posizione della tabella generatrice dei modelli: a quel punto ad essa può essere assegnato un diverso colore.

Nella modalità HRG la combinazione sfondo / primo piano può essere specificata per ciascuna delle otto "linee" di ciascuna matrice carattere.

A questo scopo sono necessari $768 \times 8 = 6144$ byte che vengono organizzati in un formato analogo a quello della tabella generatrice dei modelli.

Questo metodo di allocazione dei colori rende problematica un'eventuale procedura per lo scroll del video.

SPRITE

Uno sprite è un blocco costituito da un quadrato il cui lato è di 1, 2 o 4 celle carattere.

E' indipendente dal piano di visualizzazione e viene mosso facendo variare le coordinate X e Y che sono in relazione ad esso.

Può essere acceso o spento come si desidera; inoltre ogni collisione tra due sprite viene rilevata automaticamente.

Ogni sprite è dotato di una priorità e se una sezione non trasparente di uno sprite a più alta priorità passa davanti ad un'altro, quest'ultimo viene parzialmente nascosto.

Gli sprite consentono di ottenere con facilità effetti piuttosto sofisticati come, ad esempio, simulazioni tridimensionali.

Gli sprite non sono disponibili nella modalità testo a 40 colonne.

In tutte le altre possono essere utilizzati sullo schermo fino a 32 sprite, con un massimo di quattro per ogni linea.

Se si supera questo limite, solo i quattro sprite a più alta priorità vengono ovisualizzati sulla linea.

Inoltre viene sollevata la condizionione di "quinto sprite" ed il suo numero viene posto nel registro di stato del VDP.

Ciascun sprite può essere di un unico colore e la sua priorità non può essere cambiata.

Il BASIC mette a disposizione dell'utente due istruzioni per controllare gli sprite.

La prima,

SPRITE\$(N)

viene usata per definire un banco di modelli di sprite e la seconda,

PUT SPRITE

determina la posizione, il colore e il numero di ogni sprite visualizzato.

Questo comandi agiscono su due tabelle nella RAM video che contengono tutti i dati di visualizzazione degli sprite.

La tabella dei modelli di sprite che parte dalla locazione 14336 della VRAM (RAM Video) - indipendentemente dalla modalità di visualizzazione - contiene la definizione della forma per tutti gli sprite.

La tabella degli attributi degli sprite contiene le coordinate X e Y, il colore e il modello di ogni sprite.

Gli sprite possono avere dimensioni sia di 8 x 8 pixel, sia di 16 x 16.

Inoltre possono essere visualizzati con dimensione doppia di quella di definizione - quindi si arriva ad un massimo di 32 x 32 pixel.

Il loro posizionamento è relativo all'angolo sinistro in alto dello schermo e possono essere mossi di un pixel alla volta.

La tabella dei modelli degli sprite può contenere fino a 256 blocchi di otto byte.

Se gli sprite hanno dimensioni di 8 x 8 pixel (non si possono usare contemporaneamente quelli a 8 e a 16) allora possono essere definiti 256 modelli di sprite differenti - il modello 0

DENTRO L' MSX (Parte Seconda)

dalla locazione 14336 alla 14343, il numero 1 dalla 14344 alla 14351 e così via.

Se invece si scelgono gli sprite a 16 pixel, ciascuno di essi viene definito usando quattro blocchi consecutivi di otto byte.

Ciascun blocco definisce una sezione pari a un quarto dello sprite, nella sequenza:

1 3

2 4

Dopo la definizione della forma di uno sprite, esso viene posto sullo schermo tramite un'entrata della tabella degli attributi degli sprite.

In tutti i modi di visualizzazione che consentono l'uso degli sprite, la tabella inizia alla medesima locazione della VRAM: 6912 e ha una lunghezza massima di 32 sprite x 4 byte = 128 byte.

Ciascuna entrata serve ad indicare la posizione, il colore e il numero di modello per un singolo sprite.

I primi due byte danno le coordinate Y e X mentre il terzo identifica la forma selezionando uno o più blocchi nella tabella dei modelli degli sprite.

L'ultimo byte contiene nel semibyte meno significativo il colore di sfondo dello sprite.

In aggiunta il bit più significativo consente di spostare a sinistra, quando è posto a 1, lo sprite di 32 pixel ("clock early bit").

Le collisioni tra sprite vengono gestite in BASIC usando le istruzioni

SPRITE ON e ON SPRITE.

IL GENERATORE PROGRAMMABILE DI SUONI

Per questa funzione viene utilizzato un chip Yamaha AY-3-8910 PSG.

Si tratta di un generatore programmabile di suoni basato su registri che rende disponibili tre voci su un'estensione di otto ottave.

Inoltre l'ingresso da joystick è gestito tramite due porte di I/O indipendenti, situate sulla scheda.

Ogni voce ha un controllo separato di volume e può produrre un suono pulito oppure un rumore.

Però la stessa frequenza di rumore e forma d'involuppo viene usata per tutti i canali - se due canali producono rumore, essi producono lo STESSO rumore; può essere regolato solo il volume.

Il PSG ha 16 registri interni di lettura/scrittura, due dei quali hanno funzioni di registri di memorizzazione dati per le due porte del joystick.

I restanti registri possono essere divisi in 5 gruppi; in base alla loro funzione:

R0 - R5:

controllo della frequenza di canale.

R6:

frequenza del generatore di rumore.

R7:

registro di canale per la scelta tra rumore e suono pulito. I due bit più significativi determinano la direzione del trasferimento dati delle due porte di I/O.

R10 - R12 (Ottave):

registri per la scelta tra ampiezza del suono controllata dall'involuppo e ampiezza del suono fissata.

R13 - R15 (ottave):

registri per la selezione della forma dell'involuppo e del periodo.

Le due porte di ingresso e uscita sono del tutto indipendenti dalle funzioni di generazione del suono del PSG.

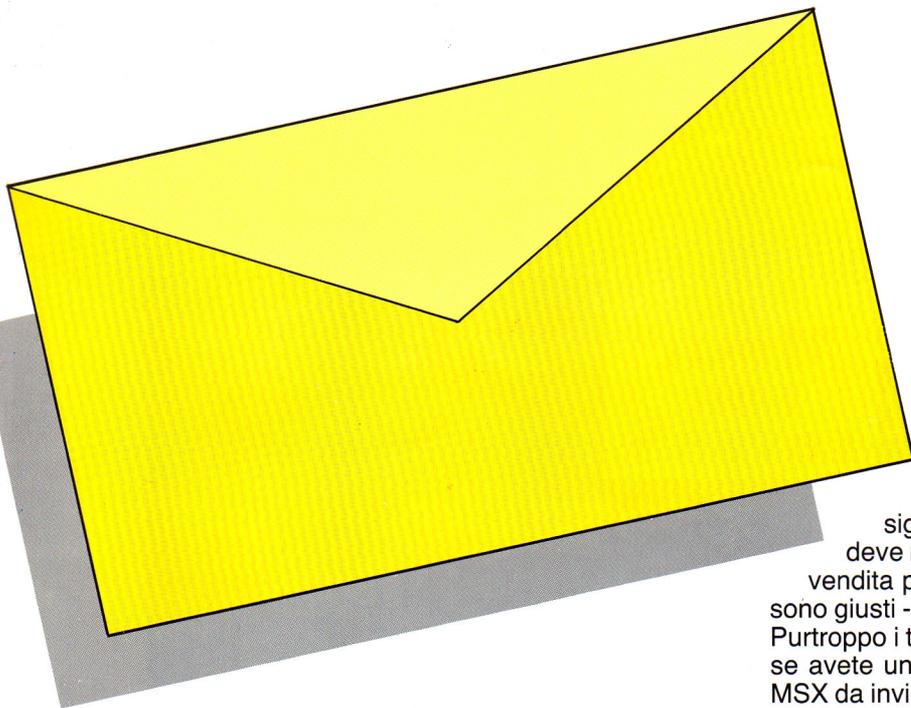
Il BASIC MSX comprende un buon numero di istruzioni per "fare musica" per le quali molti parametri sono fissati per default.

In combinazione con la capacità di generare interruzioni temporizzate, tutto quanto descritto permette di ottenere effetti musicali complessi con estrema facilità.

Questo è tutto per questa volta.

Nel prossimo numero cominceremo ad analizzare il funzionamento dell' MSX BASIC fino ad utilizzare le funzioni avanzate che mette a disposizione, tutto in funzione della programmazione Assembler.

POSTA



Alla cortese attenzione della redazione di C16/MSX. Sono proprietario di un Msx2 Philips NMS 8280 e vivo in una cittadina dove procurarsi dei programmi per lo standard Msx è molto difficile. Così, dopo aver letto la "POSTA" di uno scorso numero, ho pensato di scrivervi, in quanto avete risposto al sig. Bruno Negri che "per procurarsi software deve rivolgersi... a noi, grazie ad un nuovo servizio di vendita per corrispondenza di cui - se i tempi produttivi sono giusti - può trovare la pubblicità tra queste pagine". Purtroppo i tempi produttivi non erano giusti, così vi chiedo se avete un catalogo di giochi e di utilità per lo standard MSX da inviarmi. Vi ringrazio gentilmente.

Mario MARIOLA - Nettuno (RM)

I tempi produttivi, come ha visto, non erano giusti così siamo in ritardo, ma tra breve riceverà a casa un catalogo completo dei prodotti offerti dal Gruppo Editoriale International Education.



Spett. Gruppo Editoriale...

Vorrei sapere se il TotoMSX su dico (i programmi per fare 13 al Totocalcio) esiste anche su cassetta, perché ho un computer che funziona con quest'ultima. Se affermativo, inviati i programmi su cassetta (in contrassegno) al seguente indirizzo...

Carlo ROSTAGNO - Torralba (SS)

Non è la prima volta che ci sentiamo rivolgere questa domanda, ma la risposta è no.

Purtroppo non abbiamo disponibile la versione su cassetta di TotoMSX, ma potete trovare qualcosa all'Msx Club Italia.

L'indirizzo è ormai consueto:

**MSX CLUB ITALIA - C.P. 34
20075 LODI CENTRO (MI).**



Spettabile C16/MSX, sono un grande affezionato alla vostra mega rivista, e vi chiedo se nel prossimo numero di C16/MSX potreste mettere i seguenti giochi: Berretti Berdi, Carrier Command, Thundercats, Yes, prime minister for you, MASK, Xybots e infine un programma per trasportare i giochi da cassetta a disco.

Per migliorare la rivista vi consiglio di mettere la rubrica il MERCATO DI HARDWARE E SOFTWARE; mettere più pagine di recensione giochi rappresentando la schermata del gioco, mettere anche delle pagine dove vengono pubblicati listati di giochi o programmi vario.

Infine vi chiedo di stampare nella copertina le schermate dei giochi spiegando in modo sintetico di che cosa "tratta".

Continuate così che siete forti!!

Andrea MANUZZATO - Breganze (VI)

Grazie per i tuoi consigli che cercheremo di seguire nei limiti del possibile.

Per quanto riguarda i giochi non è facile trovare la versione Msx ma, se esistono, le troveremo.

Ti ringraziamo molto per gli stupendi disegni che ci hai mandato.

Spettabile Redazione di C16/MSX, dall'ultimo numero acquistato in edicola ho potuto vedere come sia cambiata in meglio la vostra rivista, e ho scritto questa lettera per rispondere alla vostra richiesta di consigli per migliorare ulteriormente C16/MSX. Sono un possessore di un MSX nuovo tipo, cioè quello in cui bisogna battere POKE 65535,170 prima di caricare dei programmi in L/M. A questo proposito, vorrei pregarvi di pubblicare dei giochi o altri programmi che FUNZIONINO in queste macchine (nel numero di luglio, SOCCER III e WESTPANIK non funzionavano).

Premesso questo, gradirei molto che la vostra rivista parlasse molto di più sull'MSX, riguardo in particolar modo il linguaggio macchina. Si potrebbero inserire anche delle recensioni di programmi "utili" (assemblatori, word processor, ecc.) e delle notizie sullo standard, sia in Italia (!) che all'estero. Ho anche letto che istituirete un servizio di vendita programmi per corrispondenza; l'unica richiesta è che i programmi vengano venduti ai prezzi praticati dai club (max 10.000 a programma)!

Spero possiate ancora migliorare la vostra rivista, e colgo l'occasione per porgervi i miei più cordiali saluti.

Un lettore

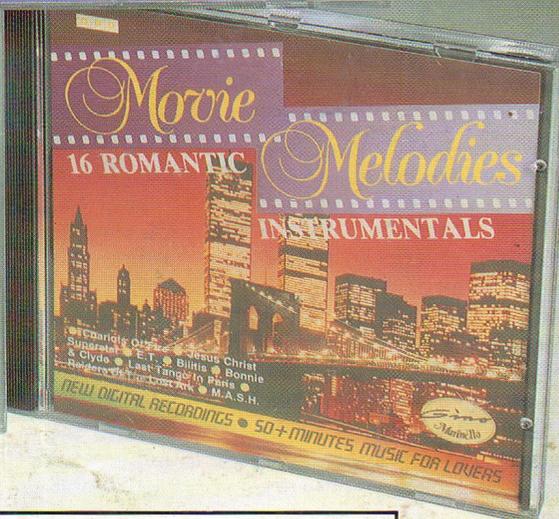
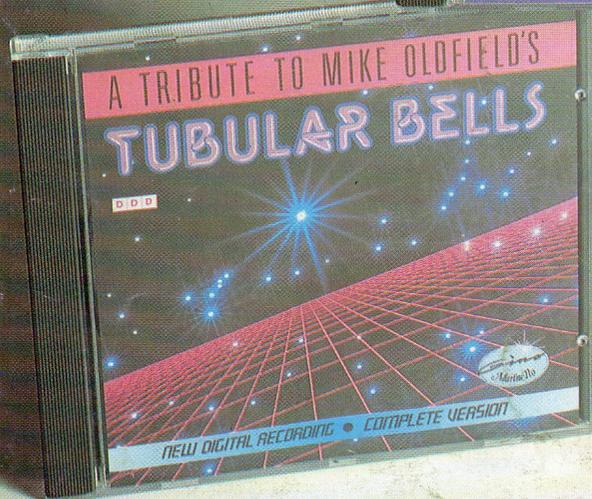
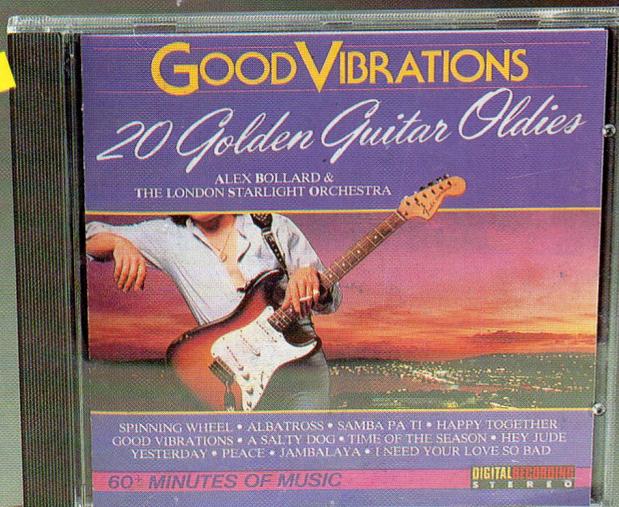
Per quanto riguarda i problemi con i programmi, se analizzate i caricatori Basic dei programmi L/M potete vedere che la poke c'è, ma se il programma non "gira" in alcun modo le cause possono essere due: la cassetta non funzionante o un'incompatibilità software tra i diversi modelli Msx. Questo vuol dire che un programma può funzionare bene su un Msx ma non su di un'altro. Per il resto l'unica cosa che ancora non abbiamo fatto tra quelle da te richiesteci sono le recensioni che, speriamo presto, cercheremo di inserire nella rivista. Per quanto riguarda la vendita di software per corrispondenza sta per essere avviata e riceverai presto il primo catalogo.

MUSIC FOR LOVERS

MUSICHE PER SOGNARE...
... insieme
SERIE "STRUMENTALI"

3 COMPACT DISC
AL PREZZO DI L. 34.900

- MOVIE MELODIES
CD 01030
- TABULAR BELLS
DGC 1029
- GOOD VIBRATIONS
CD 86028



Desidero ricevere l'offerta "MUSIC FOR LOVERS" codice CD8
Allego assegno ricevuta versamento
+ L. 2.500 quale contributo spese postali

NOME _____ COGNOME _____
VIA _____ N. _____
C.A.P. _____ CITTÀ _____
Firma _____

Compilare il coupon allegando ricevuta (o fotocopia) del versamento effettuato sul C/C n. 11319209 intestato a Gruppo Editoriale International Education srl oppure assegno non trasferibile e spedire a:

Gruppo Editoriale International Education srl
viale Famagosta 75
20142 Milano