

# PERSONAL SOFTWARE



ANNO 4 N. 32  
OTTOBRE 1985 - L. 4.800

LA PRIMA RIVISTA EUROPEA DI SOFTWARE PER PERSONAL COMPUTER

Spedizione in abbonamento postale gruppo III/70



**DESTROY  
THE CITY  
CON L'ATARI**

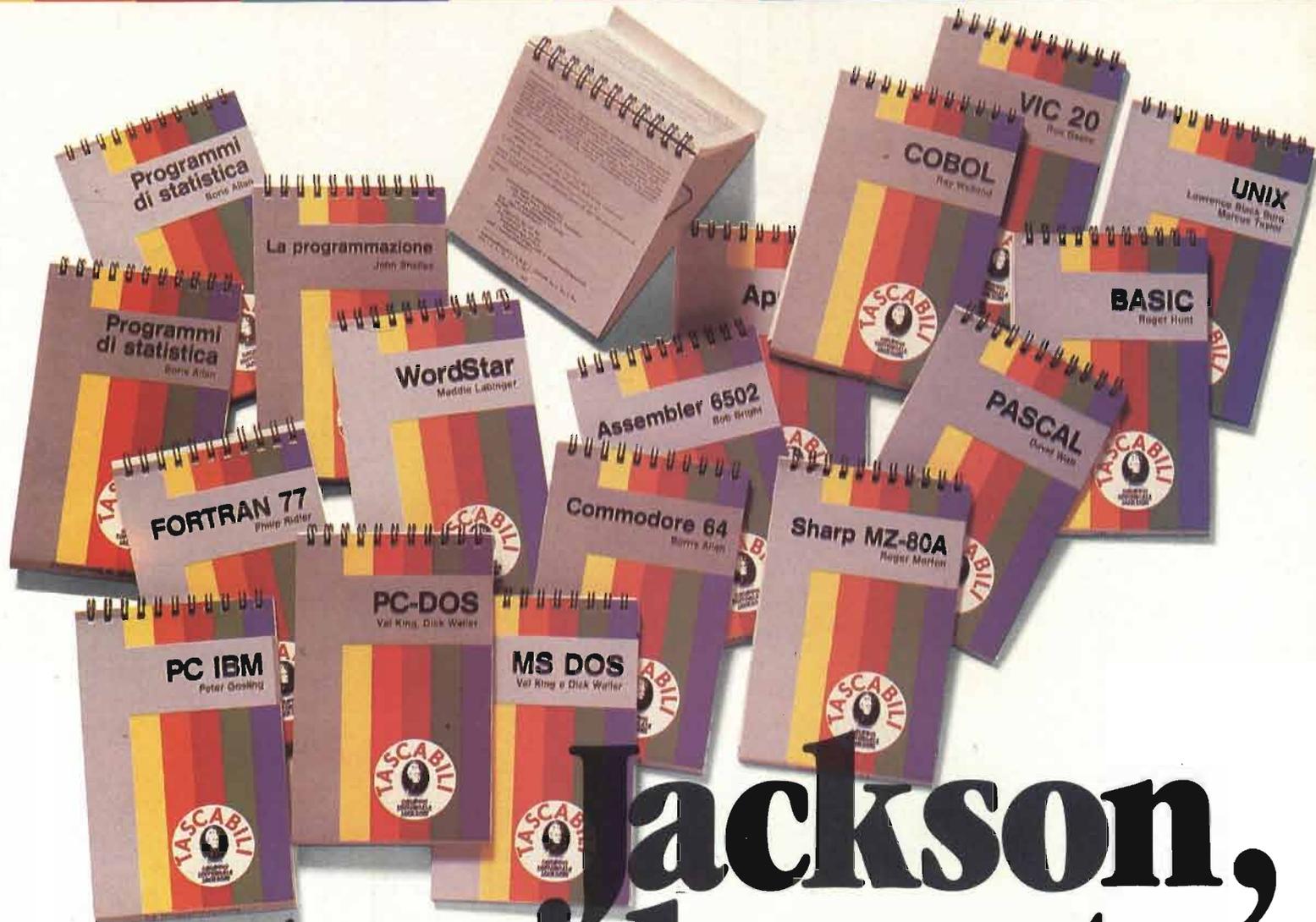
**UNO SGUARDO  
NELLA VDP RAM MSX**

**ELABORAZIONE A QUATTRO  
MANI CON L'APPLE**

**LA GESTIONE DEL VIDEO  
CON LO SPECTRUM**

**PROSPETTIVA E GRAFICA  
TRIDIMENSIONALE CON IL C64**

# i Tascabili



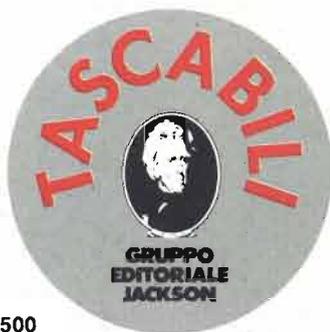
# Jackson, naturalmente.

## ELENCO DEI TITOLI DISPONIBILI

I tascabili Jackson sono uno strumento prezioso per chi lavora con il computer.

- SINCLAIR SPECTRUM cod. 017H
- VIC 20 cod. 005H
- COMMODORE 64 cod. 002H
- PC IBM cod. 018H
- APPLE IIc cod. 003H
- SHARP MZ80A cod. 014H
- LA PROGRAMMAZIONE cod. 004H
- WORD STAR cod. 008H
- UNIX cod. 009H
- LOGO cod. 020H
- MS-DOS cod. 019H
- PROGRAMMI DI STATISTICA cod. 015H
- CP/M cod. 011H
- PC-DOS cod. 012H

- BASIC cod. 007H
- ASSEMBLER Z80 cod. 016H
- ASSEMBLER 6502 cod. 013H
- COBOL cod. 001H
- FORTRAN 77 cod. 010H
- PASCAL cod. 006H



OGNI TASCABILE COSTA L. 8.500

ritagliare (o fotocopiare) e spedire in busta chiusa a:  
GRUPPO EDITORIALE JACKSON - Divisione Libri - Via Rosellini, 12 - 20124 Milano

### CEDOLA DI COMMISSIONE LIBRARIA

#### VOGLIATE SPEDIRMI

n° copie	codice	Titolo	Prezzo unitario	Prezzo totale
Totale				

Pagherò contrassegno al postino il prezzo indicato più L. 3.000 per contributo fisso spese di spedizione.

Condizioni di pagamento con esenzione del contributo spese di spedizione:

Allego assegno della Banca  Allego fotocopia del versamento sul c/c n. 11666203 a voi intestato

N° \_\_\_\_\_  Allego fotocopia di versamento su vaglia postale a voi intestato

Nome e Cognome \_\_\_\_\_

Via \_\_\_\_\_

Cap \_\_\_\_\_ Città \_\_\_\_\_

Prov. \_\_\_\_\_

Data \_\_\_\_\_ Firma \_\_\_\_\_

Spazio riservato alle Aziende Si richiede l'emissione di fattura

Partita I.V.A. \_\_\_\_\_

ORDINE  
MINIMO  
L. 50.000

# SOMMARIO



Questo mese: Destroy the city, un affascinante videogioco per Atari 800 XL.

ANNO 4  
N. 32  
OTTOBRE 1985

---

<b>ANIMAZIONE DI FUNZIONI TRIDIMENSIONALI</b> <i>di Carlo Cappelli</i>	ZX SPECTRUM	<b>12</b>
<b>DESTROY THE CITY CON L'ATARI</b> <i>di Daniela Cerù</i>	ATARI XL 800	<b>18</b>
<b>LA GESTIONE DEL VIDEO</b> <i>di Carlo Cappelli</i>	ZX SPECTRUM	<b>20</b>
<b>UNO SGUARDO NELLA VDP RAM</b> <i>di Sergio Borsani</i>	MSX	<b>45</b>
<b>PROSPETTIVA E GRAFICA TRIDIMENSIONALE</b> <i>di Stefano Checchini</i>	COMMODORE 64	<b>52</b>
<b>INTESTAZIONE CASSETTE</b> <i>di Paolo Rossi</i>	SHARP MZ-700	<b>63</b>
<b>ELABORAZIONI A QUATTRO MANI</b> <i>di Paolo Morocutti</i>	APPLE II	<b>66</b>
<b>LA SIMULAZIONE DINAMICA DI FENOMENI CONTINUI 2°</b> <i>di Franco Sardo</i>	COMMODORE 64	<b>73</b>

## RUBRICHE

---

<b>EDITORIALE</b> <i>di Riccardo Paolillo</i>	<b>5</b>
<b>POSTA</b>	<b>6</b>
<b>PERSONAL NEWS</b> <i>a cura di Marco Giacobazzi</i>	<b>8</b>
<b>PERSONAL MARKET</b>	<b>80</b>

**CON INSERTO SUPERBIT**  
64 PAGINE DI SOFTWARE PER IL TUO PERSONAL

# è in edicola il nuovo numero

*Bit, la prima  
e più diffusa rivista  
di personal computer  
e accessori*



**UNA  
PUBBLICAZIONE  
DEL  
GRUPPO  
EDITORIALE  
JACKSON**

SAN FRANCISCO-LONDRA-MILANO



**B**itest:  
Atari 520 ST

**S**harp  
PC-2500

**C**orso  
di Assembly  
per C 64

**T**empo!  
un gioco  
per Apple

**S**onata  
per MSX

**I**nteger  
Compiler  
per Spectrum

**L**a didattica  
e il C 64

**O**rgano  
a 9 ottave  
per Sharp

**SPECIALE:  
LINGUAGGI**

## *Crimini informatici*

**S**i parla spesso dei benefici apportati alla collettività dalla continua crescita telematica. Sono sicuramente rilevanti e ne abbiamo più volte messo in luce portata e conseguenze. Occorre però esaminare anche l'altra faccia della medaglia, e cioè cosa succede quando avvengono usi illeciti degli strumenti che la tecnologia mette a disposizione.

Infatti i casi di "crimini informatici" stanno avendo un peso non irrilevante in ambiente giudiziario ed è logico che si cerchi di studiare sistemi idonei a ridurne la nocività.

L'uso illegale dei calcolatori viene normalmente effettuato in ambiente bancario o comunque finanziario per accreditare fondi su conti correnti o direttamente "liquidi" in modo abusivo.

Alcuni casi clamorosi, come il colpo realizzato da una banda che falsificando le opportune tessere magnetiche è riuscita a fare incetta di denaro contante agli sportelli bancari automatici, sono stati ripresi con grande evidenza dagli organi di stampa.

Ma anche i colpi messi a segno presso banche e grandi aziende da impiegati e programmatori infedeli che, grazie al proprio terminale, sono riusciti ad intascare cifre considerevoli rappresentano un campanello d'allarme.

La percentuale di crimini di questo tipo, che vengono scoperti e puniti è bassissima. Si calcola che negli Stati Uniti si aggiri intorno ad un irrisorio 1%, e questo non per negligenza nelle indagini, ma semplicemente perché le aziende raggirate preferiscono non sporgere denuncia.

Infatti rendere di pubblico dominio questo tipo di eventi per molte aziende può costituire una pubblicità negativa ed una conseguente perdita di immagine ed affidabilità rispetto ad una clientela che esige sicurezza e riservatezza.

La soluzione, che per forza di cose viene ricercata, è di tipo preventivo più che repressivo; l'adozione di sistemi di protezione sempre più sofisticati può sicuramente contribuire a ridurre il numero di illeciti.

Anche in settore non economico è possibile effettuare degli abusi con il terminale. Le numerosissime banche dati esistenti possono essere consultate in modo indiscreto da persone non autorizzate. In questo modo dati particolarmente delicati riguardanti ad esempio le idee politiche e religiose dei cittadini possono essere velocemente consultati, modificati e confrontati per fini illeciti.

Anche in questo caso occorre adottare idonee misure di controllo e comunque ricordare anche ai più pessimisti che se non esistessero archivi elettronici non ci sarebbe la possibilità di avere veloci ed aggiornate informazioni che permettono di individuare i responsabili di reati comuni e di terrorismo, gravissimi per la collettività.

## Errori di italiano

Leggo ed apprezzo la vostra rivista fin dal primo numero, sono particolarmente interessata alla simulazione dinamica e mi ha fatto piacere leggere in proposito l'articolo pubblicato nel n. 30.

Mi sembra quasi superfluo segnalare l'errore in esso contenuto:  $V_{12}$  sta ovviamente per  $V_{12}$ , ma vorrei invece richiamare la vostra attenzione su qualche errore d'italiano riscontrato più avanti; precisamente a pag. 28, "digressione" sta per "digressione" e, ancora a pag. 28, "un'unico input" non richiede l'apostrofo.

Insegno matematica e fisica in un liceo, molti dei miei studenti leggono Personal Software, spesso ci scambiamo la rivista, mi dispiacerebbe se questa risultasse per alcuni versi diseducativa. So bene che gli errori di stampa sono pressoché inevitabili, ma credo che sia possibile una maggiore attenzione.

*Rita Serafini Tancredi  
Perugia*

La ringraziamo per l'attenzione e la fedeltà con cui segue Personal Software. Sapere di essere così seguiti in ambito scolastico ci spingerà a curare sempre più, oltre che i contenuti, anche la forma con cui sono espressi.

Anche se, come giustamente osserva, sviste tipografiche sono purtroppo all'ordine del giorno nel nostro lavoro.

## Mini-Utility C 64

Ho deciso di scrivervi per informarvi di una piccola utility che non ho mai visto pubblicata e che ritengo di grande utilità per tutti i possessori di C 64 con un disk drive.

A volte capitano dei programmi su nastro che non hanno il nome e, dato che

il registratore non è una memoria ad accesso casuale come il drive, non si presentano problemi per salvarli.

Quando però si cerca di passare questi programmi su disco ci si deve arrendere di fronte all'impossibilità di dare loro un nome.

Infatti se si dà un nome al programma si verifica un errore di "Out of Memory", ma senza fornire un nome per il programma, questo non può essere memorizzato su disco.

La mia utility risolve questo problema: essa dimensiona lo spazio di memoria a disposizione delle matrici, lasciando così all'utente memoria sufficiente per dare un nome al programma. Ovviamente è impossibile, dopo aver battuto questa utility, chiedere la memoria libera al BASIC pena il blocco del computer, ma vi assicuro, come del resto potrete facilmente verificare, che la utility funziona perfettamente.

Essa è brevissima: [Poke 49,2 : Poke 50,2]. Ora potrete dare un nome al programma e salvarlo senza alcun problema.

*N.B.* - In seguito per caricare il programma non sarà necessario nessun particolare accorgimento.

*Luca Arzeni  
Pescara*

Registriamo questa mini-utility che ci invia il lettore sperando che possa essere utile a quanti utilizzano sia il drive che l'unità a cassette per memorizzare i dati del proprio C 64.



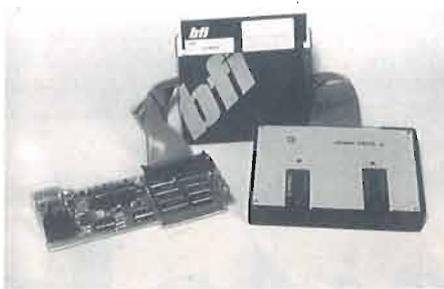
## Stampanti compatibili Mannesmann

**S**empre in ambiente Apple, segnaliamo l'arrivo delle stampanti ad aghi MT/85 e MT/86, dotate di interfaccia Imagewriter che le rende immediatamente collegabili anche al Macintosh. La velocità di stampa è di 180 caratteri al secondo, e diventa di 45 in Near Letter Quality, la grafica è compatibile IBM, la rumorosità (53 DBA) molto ridotta ed è possibile disporre di più fonti di caratteri.

*Mannesmann Tally S.r.l.  
Via Cadamosto, 3  
20094 Corsico (MI)  
Tel. 02-4502850*

## Programmatore universale su Apple

**L**a Elmarc ha introdotto sul mercato un sistema per la programmazione di EPROM via Apple II e compatibili. Il software è distribuito sui classici supporti da 5,25" e l'operatore dispone di una serie ben concepita di menu attraverso i quali scegliere le opzioni desiderate. È possibile manipolare i blocchi della memoria mediante un po-



tente editor; i collegamenti con l'esterno riguardano il disco, memorie già scritte e tutto quanto può essere collegato via linea seriale RS-232. L'Adapter EM09/A si rivolge alle EPROM TMS2708, TMS2716 e a tutte quelle a singola alimentazione delle serie 25XX e 27XX in tecnologia MOS e CMOS fino alle 27152.

*Elmarc S.r.l.  
Via Tiziano, 71  
60125 Ancona  
Tel. 071-81318*

## Qualcosa si muove in casa Atari

**L**a Casa americana ci propone un'importante anteprima: Atari e Activenture hanno sviluppato hardware (il sistema 520 ST) e software (Facts and Figures) per lo sfruttamento del Compact Disk Drive, la memoria di massa (a sola lettura) del futuro. Basata su dischi di materiale plastico, chiamati CD-ROM, impiega la tecnologia e gli stessi supporti che ci sono familiari nel mondo della musica, arrivando a registrare fino a 500 milioni di byte su un singolo disco. Ci piace pensare alla rivoluzione che la diffusione di queste memorie potrà rappresentare: non più scaffali colmi di testi, ma una biblioteca di minuscoli "dischi illustrati" a basso costo, aggiornabili periodicamente per sostituzione, acquistando magari in edicola un'intera enciclopedia con figure a colori contenuta su un paio di nuovi dischetti. Il software di gestione Activenture, Facts and Figures, consente di reperire le informazioni presenti in CD-ROM mediante l'interfaccia grafica GEM della Digital Research ed una serie di tavole d'accesso indicizzate per ricerche con vari criteri: argomento, autore, titolo, soggetto. Non siamo ancora in grado di comunicarvi disponibilità e prezzo del sistema. Questi dati sono invece ben noti per quanto concerne il modello 130 XE 128 Kbyte, ultimo rampollo Atari sbarcato in Italia, basato su 6502C con clock da 1,79 MHz. Con 380.000 lire più IVA è possibile portare a casa questo sistema con 128 Kby-



te di RAM, BASIC su ROM e compatibile con la serie 800XL. Per finire con Atari, informiamo i possessori dell'unità a disco 1050 corredata di sistema operativo DOS III che hanno diritto di richiedere la versione aggiornata dello stesso (la 2.5, ovvero quella del 130 XE), inviando la fotocopia del certificato di garanzia all'indirizzo sotto riportato.

*Atari Italia S.p.A.  
Via dei Lavoratori, 19  
20092 Cinisello B. (MI)*

## Un po' di tutto ...

- **Videobit** (via Console Marcello, 18/5 - Milano - Tel. 02-390516) distribuisce un'interfaccia per ZX Spectrum che si collega al connettore d'espansione del computer e comprende un sistema d'alimentazione, un sistema operativo (56 Kbyte di ROM) per la gestione della scheda, un debugger con programmatore di EPROM 2764-27128, 8 Kbyte di RAM di servizio, un controller per doppio disk drive (da 100 Kbyte l'uno), interfaccia stampante e RS-232, amplificatore suoni e tasto Reset.
- Si è svolta a Milano dal 27 Maggio al primo Giugno "La posta in gioco", una settimana sul gioco elettronico con mostre, conferenze e ampia messe di video e game da usare sul posto. Organizzata dal CRT (Centro di Ricerca per il Teatro) di Milano ha visto la partecipazione di una serie di case costruttrici e del Gruppo Editoriale Jackson che ha proposto, tramite la redazione della rivista *Video Giochi*, una

scelta delle coloratissime buste usate nella corrispondenza dai suoi lettori.

- La **NCR** (Viale Cassala, 22 - Milano) ha partecipato alla regata transatlantica Portofino-New York di quest'estate con un 12 metri condotto da Alberto Morerio e Lucia Pozzo e fornendo inoltre il computer di bordo del Rolly Go, barca appoggio della spedizione. I programmi del personal computer NCR hanno consentito di conoscere giorno per giorno la classifica reale, in tempo compensato, e la posizione delle imbarcazioni partecipanti.

- La **Remat Elettronica S.r.l.** (Via Monte Trina, 2 - Roma - Tel. 06-899007) produce e distribuisce una scheda di espansione grafica per i computer Sharp MZ-700. La risoluzione video diventa di 320 per 200 punti ed è possibile accedere ad una serie di routine in linguaggio macchina per sfruttarne a pieno le caratteristiche. La scheda costa 200.000 lire più IVA.

- A Parigi, dal 2 al 6 Dicembre, si terranno due saloni specializzati per l'informatica e la didattica. Il primo, è **Educatec '85**, specializzato in apparecchiature, tecniche e materiali per l'insegnamento. Formation '85 è invece rivolta alla formazione professionale, ai mestieri e alle tecniche del futuro.

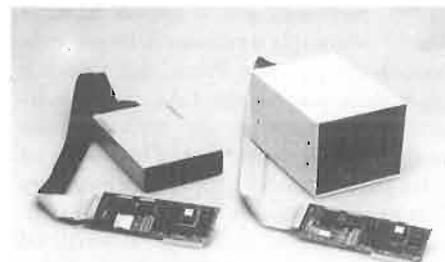
- A Firenze, dal 22 al 25 Novembre prossimo si terrà, nella sede espositiva della Fortezza da Basso, la terza edizione di "Exposer - Firenze Informatica".

Il salone dell'Ufficio e dell'Informatica anche quest'anno annovera fra le oltre 200 Ditte rappresentate le maggiori Case italiane ed estere.

## Cresce la memoria dell'Apple II

**P**er far fronte alle esigenze di chi tratta voluminosi data base o testi chilometrici, la Datatech propone due nuove unità per floppy da 5,25" compatibili con l'Apple II ed il IIe. I due drive, singolo e doppio, necessitano di un'apposita scheda controller, da inserire nello slot 6 al posto di quella standard e garantiscono una capacità di 640 Kbyte formattati per dischetto, arrivando quasi a quintuplicare le possibilità di memorizzazione del sistema. Con l'unità a

singolo drive SF-AP 960 (standard Shugart a 96 TPI) è possibile configurare un sistema che monta un lettore Apple e uno da 640 Kbyte. L'unità SF-AP 961 con alimentatore incorporato consente di disporre di 1.280 Kbyte in linea. Il kit di montaggio contiene anche un floppy con le utility di formattazione e copia per l'estensione del DOS. L'SF-AP 960 costa 770.000 lire,



mentre il modello superiore arriva a 1.300.000 lire, sempre IVA esclusa. Sono in arrivo unità da 3,5" con le stesse capacità.

*Datatech S.p.A.  
Palazzo Tr - Milanofiori  
20089 Rozzano (MI)  
Tel. 02-8243382*

## North-Holland per la didattica

**L**a casa editrice specializzata in periodici e testi per l'informatica ha aggiunto al suo nutrito catalogo una pubblicazione che uscirà ogni trimestre e riguarderà gli sviluppi della tecnologia che più coinvolgono gli aspetti educativi, nell'azienda come nella scuola. È possibile richiedere una copia omaggio di Education & Computing scrivendo al seguente indirizzo:

*Elsevier Science Publishers  
Attn. B. Wennekendonk  
P.O. Box 1991  
1000 BZ Amsterdam  
The Netherlands*

## Software in abbondanza per QL

**P**roseguito nella sua strategia di produzione di programmi professionali per il QL la Sinclair annuncia il rilascio della seconda versione del package Psion, più veloce e capace della precedente. Con questo software aggiornato è in arrivo anche Config.bas, per configurare l'hardware ed alcuni parametri del proprio sistema. Per l'Europa continentale la Sinclair ha pianificato inoltre la traduzione di manuali ed applicazioni rivolte all'utenza professionale: tra queste QL Cash-Trader, sistema di contabilità multitasking a finestre per gestire flussi di cassa e situazione IVA. Altri pacchetti, alcuni sviluppati in Italia, arri-



veranno entro l'anno e tra di essi si annuncia particolarmente interessante il programma di comunicazione tra QL ed IBM. Prima di chiudere ricordiamo che il QL ha partecipato all'ultima edizione del Giro d'Italia per aiutare i giornalisti a ricordare i dati statistici delle varie tappe.

Ricordiamo infine con piacere che ha avuto inizio con il numero di giugno la pubblicazione su *Compuscuola*, rivista del Gruppo Editoriale Jackson particolarmente attenta al legame tra didattica e computer, di un corso di educazione informatica rivolto agli insegnanti. L'esperienza da cui trae origine il corso, supportata da materiale di produzione Sinclair, vede infatti come attori principali gli insegnanti stessi e non gli esperti dei nuovi strumenti. Il programma, coordinato dal prof. Giancarlo Mauri dell'istituto di Cibernetica del-

l'Università di Milano, ha coinvolto tre docenti delle scuole medie di Milano e provincia.

*Rebit Computer*  
Viale Matteotti, 66  
20092 Cinisello Balsamo (MI)  
Tel. 02-6181801

## Microprofessor per l'8088

**L**a felice serie dei microcomputer didattici Microprofessor si amplia anche in Italia dell'ultimo prodotto, l'MPF-1/88, basato su uno dei micro più in voga del momento. Si tratta di un mezzo per imparare a conoscere a fondo le caratteristiche hardware e software del "cuore" dell'IBM PC e di tanti altri compatibili. Il nuovo sistema Multitech può convertire un programma da codice Assembler a linguaggio macchina durante la sua scrittura; trovare ed eliminare gli errori che si manifestano durante l'esecuzione di un programma; caricare e scaricare sorgenti su cassetta. La RAM di partenza è di 4 Kbyte, ma può essere espansa a 24, il display è di due righe da venti caratteri e la tastiera dispone di 59 tasti alfanumerici; sono presenti numerose



interfacce, tra cui una Centronics per il collegamento dello stampante.

*Zelco S.r.l.*  
P.le Cadorna, 13  
20123 Milano  
Tel. 02-804382

## Acquisizione dati via Apple o IBM

**I**l catalogo della Pertel si è recentemente arricchito della produzione 3D (Digital Design and Development), una società inglese specializzata in schede e rack modulari per la raccolta di dati, utilizzanti l'Apple II o l'IBM PC come unità di controllo intelligenti. L'integrazione al software di base compiuta dall'azienda italiana le permette offrire



versatili strumenti di lavoro ove siano richiesti A/D converter veloci 8-16 canali, 8-12 bit; D/A converter di precisione; controllori con relè opto-isolati o al mercurio o con mosfet di potenza; amplificatori per termocoppie; controllori TTL fino a 32 bit o per stepper motor a 4 fasi; sistemi di trasduzione lineare di precisione con risoluzione fino ad 1 micrometro.

*Pertel S.n.c.*  
Via Ormea, 99  
10126 Torino  
Tel. 011-655865

# Bit

La prima rivista europea di personal computer, software e accessori. Con test, novità, analisi del mercato... 11 numeri all'anno: L. 5.000 a numero  
Abbonamento: solo L. 43.000

**Quando l'informazione fa testo**

# Ecco i vincitori del Grande Concorso "VINCI 30 MSX PHILIPS"

- 1 - **MORESE Salvatore**  
Coop. Borgo Incoronata-Sc.l -  
71040 FOGGIA
- 2 - **PILLA Erminio**  
Via Cechov, 20 - 20151 MILANO
- 3 - **BATTAGLIA Vito**  
Via Pietragrossa Comp. A.R.G.I.-  
Sc.B3 - 90015 CEFALÙ (PA)
- 4 - **FIorentini Federica**  
Via Matilde Serao, 9 - 10141 TORINO
- 5 - **MAMELI Luigi**  
Viale S. Ignazio, 38 -  
09100 CAGLIARI
- 6 - **CERBONE Roberto**  
Via E. De Nicola, 39/5  
80059 TORRE DEL GRECO (NA)
- 7 - **MAGLIO Giuseppe**  
Via Bellini, 53 - 36030 POVOLARO (VI)
- 8 - **SIMIONATI Vittorio**  
Via Gorizia, 3 - 13051 BIELLA (VC)
- 9 - **PIERDICCA Claudio**  
Via Crivelli, 9 - 60125 ANCONA
- 10 - **GRAZZI Mauro**  
Via Ricciarelli, 19 - 44100 FERRARA
- 11 - **GREMMO Stelvio**  
Via Rosmini, 13 - 13051 BIELLA (VC)
- 12 - **FENU Franco**  
Via Vito Frazzi, 24  
50018 SCANDICCI (FI)
- 13 - **PETROZZI Mario**  
Via Trapani, 42 - 71042 CERIGNOLA
- 14 - **SANA Giovanni**  
Via Paderno, 25/i  
24068 SERIATE (BG)
- 15 - **BATACCHI Filippo**  
Via del Filarete, 28 - 50143 FIRENZE
- 16 - **COMPAGNO Domenico**  
Via F.sco Purpura, 23  
90127 PALERMO
- 17 - **MASTRANGELO  
Roberto** - Via Bizzarri, 31  
24042 CAPRIATE (BG)
- 18 - **SUMAN Cristiano**  
Via Industria, 10  
30010 CAMPONOGARA (VE)
- 19 - **GOLIA Romano**  
Via Vitt. Veneto, 7  
67019 CASALE DI SCOPPATO (AQ)
- 20 - **PIAZZA Giuseppe**  
Via A. Moro, 25/1  
33072 CASARSA D. DELIZIA(PN)
- 21 - **BELLOCCI Maura**  
Via Fattori, 73 - 10141 TORINO
- 22 - **FERRARI Pietro  
Giuseppe** - Via Gazzolo, 5  
25080 BOTTICINO MATTINA (BS)
- 23 - **DAMANTE Maurizio**  
Via del Serbatoio, 31  
00054 FIUMICINO (Roma)
- 24 - **LESENTENZE  
Bernardo** - Via Papa Benedetto  
XIII, 3 - 70124 BARI
- 25 - **KEDEROGLU Panagiotis**  
Via Monte Sabotino, 32  
20099 SESTO S.GIOVANNI (MI)
- 26 - **ANTINUCCI Amedeo**  
Via Principe di Piemonte, 22  
86100 CAMPOBASSO
- 27 - **LA BUA Antonino**  
Via Città di Palermo, 139  
90011 BAGHERIA (PA)
- 28 - **DOTTO Enrico**  
Via dei Mille, 15  
30038 SPINEA (VE)
- 29 - **BUSSOLA Renato**  
Via A. De Gasperi, 24  
37029 S. PIETRO IN CARIANO (VR)
- 30 - **CRIVELLI Luigi**  
Via Monte Fumaiolo, 44  
00139 ROMA

## CONGRATULAZIONI!!!



**GRUPPO  
EDITORIALE  
JACKSON**

Milano • San Francisco • Londra • Madrid

**P**er il suo funzionamento occorre caricare la routine in linguaggio macchina per la gestione delle immagini video presentata sul n. 31 di **Personal Software**. Potete memorizzare e richiamare fino a 5 grafici completi; ogni immagine occupa 6.144 byte.

Con questo programma potrete alterare velocemente le immagini sul video ottenendo delle interessanti animazioni simili a quelle della sigla di *Quark*.

All'inizio occorre caricare il linguaggio macchina precedentemente citato; fatto questo, dovete inserire il numero massimo di fotogrammi che desiderate, da 1 a 5; questa informazione serve per predisporre la memoria e inizializzare la routine in linguaggio macchina (per quanto riguarda il funzionamento di questa routine vi rimandiamo all'artico-

# Animazione di funzioni tridimensionali

Questo programma studia e memorizza i grafici delle funzioni tridimensionali

di Carlo Cappelli

Listato 1 - Il programma che esegue le animazioni.

```

10 REM
20 REM --inizializzazione--
30 REM
40 CLEAR 65300
50 BORDER 0: PAPER 0: INK 7: C
LS
60 PRINT "*****"
*****"
70 PRINT "* STUDIO E ANIMAZI
ONE DI *"
80 PRINT "* GRAFICI TRIDIME
NSIONALI *"
90 PRINT "*****"
*****"
100 PRINT "*****"CARICA IL L/M
PER LA GESTIONE DELLE IMMAGINI
OPPURE UN FILE DI FOTOGRAMMI"
110 LOAD "CODE
120 IF PEEK 64500<>0 THEN POKE
64501,1: LET n=PEEK 64500: GO TO
150
130 INPUT "numero di fotogrammi
(1-5)";n
140 IF n<1 OR n>5 THEN GO TO 13
0
150 POKE 64500,n
160 LET a$=CHR$ 0+CHR$ 0+CHR$ 3
2+CHR$ 22+CHR$ 0+CHR$ 1+CHR$ n
170 CLEAR USR 84052
180 LET n1=5*(PEEK 64501=1)
190 LET n=PEEK 64500
200 IF n1=0 THEN GO TO 250
210 PRINT " " elaborazione i
mmagine " " visualizzazione de
lle immagini memorizz
ate"
220 INPUT LINE c$
230 IF c$="2" THEN GO TO 1330

```

```

240 IF c$<>"1" THEN GO TO 220
250 REM
260 REM --inizializzazione
funzione
tridimensionale
270 REM
280 DIM v(50)
290 CLS
300 PRINT " GRAFICA TRIDIMEN
SIONALE"
310 INPUT "z = f(x,y) = ";f$
320 PRINT " ";f$
330 INPUT "xmin ";xmi
340 PRINT "xmin ";xmi
350 INPUT "xmax ";xma
360 PRINT "xmax ";xma
370 INPUT "ymin ";ymi
380 PRINT "ymin ";ymi
390 INPUT "ymax ";yma
400 PRINT "ymax ";yma
410 INPUT "zmin (autonorm=0) ";
zmi
420 PRINT "zmin ";zmi
430 INPUT "zmax (autonorm=0) ";
zma
440 PRINT "zmax ";zma
450 INPUT "step(3-25) ";s
460 INPUT "vuoi il piano di rif
erimento (s/n)";c$
470 CLS
480 LET sx=INT (195/s)
490 LET sy=INT (60/s)
500 LET a=s*SGR 3
510 LET dx=(xma-xmi)/sx
520 LET dy=(yma-ymi)/sy
530 IF zma<>zmi THEN GO TO 0690
540 REM
550 REM --calcolo oscillazine
della z
560 REM

```

lo in proposito). Ora inserite la funzione  $z=f(x,y)$  e i parametri entro i quali tale funzione deve essere studiata. Si deve inoltre stabilire l'ampiezza di oscillazione della  $z$ . In definitiva occorre precisare le dimensioni del parallelepipedo che contiene tutta la figura. A volte è difficile sapere a priori l'oscillazione della  $z$ , quindi, conviene fare in modo che sia il programma stesso a stabilirla; per ottenere questo risultato occorre inserire per  $z_{min}$  e per  $z_{max}$  il valore 0. Il disegno viene realizzato a rete, quindi dovete inserire l'ampiezza delle maglie dando uno step tra 3 e 20.

Infine, vi viene chiesto se desiderate il piano di riferimento o meno. Ricevuti questi dati, il computer traccia il grafico della funzione, facendo attenzione alle parti in vista e cancellando quelle nascoste. Terminato il grafico potete me-

morizzare la schermata o passare subito a comporne un'altra. I grafici memorizzati possono essere rivisti, animati e salvati su cassetta. Nell'animazione occorre inserire la sequenza con cui si desidera rivederli e l'intervallo fra una immagine e l'altra.

Quando salvate il file di immagini, viene salvata anche la routine in linguaggio macchina, quindi, all'inizio del programma, potete caricare il file invece della semplice routine. Una volta completata la sequenza di immagini, se volete cambiarne alcune, non dovete far altro che comporre e memorizzarle; a questo punto il programma, con una scritta rotante nella parte bassa del video, vi segnalerà che i fotogrammi sono al completo e vi chiederà il numero del fotogramma da sostituire con quello sullo schermo. Premete il tasto deside-

rato e il gioco è fatto. Se invece avete cambiato idea e non desiderate più memorizzare l'immagine sullo schermo, premete 0.

Le funzioni sono visualizzate in assonometria con angolo di vista principale di  $60^\circ$ .

La semplicità del programma è resa possibile dalla duttilità della funzione Draw e dai comandi Over e Inverse che permettono la rapida cancellazione di ciò che non si deve vedere. Nell'affrontare questo aspetto, ci siamo serviti di un particolare algoritmo che risolve il problema senza tener conto di concavità e intersezioni delle superficie.

La superficie è approssimata con una struttura a rete dove la distanza tra due punti della  $x$  dipende dalla variabile  $dx$  (dy per la  $y$ ). Tale variabile è direttamente proporzionale allo step inserito in in-

```

570 LET X=xmi: LET y=ymi
580 LET zmi=VAL f$: LET zma=zmi
590 FOR x=xmi TO xma STEP dx
600 FOR y=ymi TO yma STEP dy
610 LET z=VAL f$
620 IF z<zmi THEN LET zmi=z
630 IF z>zma THEN LET zma=z
640 NEXT y
650 NEXT x
660 REM
670 REM --elaborazione immagine
680 REM
690 LET mz=71/(zma-zmi)
700 LET x=xmi: LET j=1
710 FOR y=ymi TO yma STEP dy
720 LET z=INT ((VAL f$-zmi)*mz+
j*a-a+.5)
730 LET z=z*(z>0 AND z<=175)+17
5*(z>175)
740 LET v(j)=z
750 LET j=j+1
760 NEXT y
770 IF c$="n" THEN GO TO 0830
780 PLOT 0,0
790 DRAW 195,0
800 DRAW 50,104
810 DRAW -195,0
820 DRAW -50,-104
830 LET i=0
840 FOR x=xmi+dx TO xma STEP dx
850 LET i=i+1: LET j=sy+1
860 FOR y=yma TO ymi-dy/5 STEP
-dy
870 LET a1=z
880 LET z=INT ((VAL f$-zmi)*mz+
j*a-a+.5)
890 LET z=z*(z>0 AND z<=175)+17
5*(z>175)
900 IF y=yma THEN GO TO 1080
910 LET a2=s*(i+j)

```

```

920 LET a3=(v(j+1)-a1)/s
930 LET a4=(z-v(j+1))/s
940 FOR k=1 TO s
950 PLOT INVERSE 1; OVER 1;a2-k
,a1+a3*k
960 DRAW INVERSE 1;0,a4*k
970 NEXT k
980 LET a2=s*(i+j-2)
990 LET a3=(v(j+1)-v(j))/s
1000 FOR k=1 TO s
1010 PLOT INVERSE 1; OVER 1;a2+k
,v(j)+a3*k
1020 DRAW INVERSE 1;0,a4*k
1030 NEXT k
1040 PLOT s*(i+j),a1
1050 DRAW -s,v(j+1)-a1
1060 DRAW -s,v(j)-v(j+1)
1070 LET v(j+1)=a1
1080 LET j=j-1
1090 NEXT y
1100 DRAW s,z-v(1)
1110 LET v(1)=z
1120 NEXT x
1130 FOR j=1 TO sy
1140 DRAW s,v(j+1)-v(j)
1150 NEXT j
1160 REM
1170 REM --memorizzazione
1180 REM
1190 INPUT "lo memorizzi (s/n)?" ;
c$
1200 IF c$="n" OR c$="N" THEN GO
TO 210
1210 LET n1=n1+1: IF n1>n THEN L
ET n1=n: GO TO 1230
1220 GO TO 1020
1230 LET a$="          fotogramm
i al completo      premi 0 per rit
orno menu' o da 1 a "+STR$ n+" p
er inserimento

```

Seguito listato 1.

```

1240 FOR i=1 TO LEN a$-31
1250 PRINT #1;AT 1,0;a$(i TO i+3
1)
1260 PAUSE 8
1270 LET c$=INKEY$: IF c#<"0" OR
c#>STR$ n THEN NEXT i: GO TO 12
40
1280 IF c#="0" THEN GO TO 1330
1290 LET a$=CHR$ 0+CHR$ 0+CHR$ 3
2+CHR$ 22+CHR$ 0+CHR$ VAL c#+CHR
$ n
1300 RANDOMIZE USR 64052
1310 INPUT 1
1320 RANDOMIZE USR 64166
1330 CLS
1340 PRINT "***** 1 studio funz
ione" " " 2 visualizzazione immag
ini" " " 3 animazione immagini"
1350 PRINT " " 4 salva immagini"
" " 5 carica file di fotogrammi"
1360 INPUT LINE c#
1370 IF c#="1" THEN GO TO 280
1380 IF c#="2" THEN GO TO 1450
1390 IF c#="3" THEN GO TO 1560
1400 IF c#="4" THEN GO TO 1740
1405 IF c#="5" THEN GO TO 50
1410 GO TO 1360
1420 REM
1430 REM --visualizzazione
1440 REM
1450 CLS
1460 INPUT "numero fotogramma ch
e si vuole visualizzare "; LINE
c#
1470 IF c#<"0" OR c#>STR$ n THEN
GO TO 1340
1480 LET a$=CHR$ VAL c#
1490 POKE 64042,1: RANDOMIZE USR
64323
1500 RANDOMIZE USR 64368
1510 PRINT #1;"premi un tasto":
PAUSE 0
1520 GO TO 1330
1530 REM
1540 REM --animazione
1550 REM
1560 INPUT "inserisci la sequenz
a con cui vuoi vedere i fotogram
mi "; LINE c#
1570 IF LEN c#>255 THEN GO TO 13
40
1580 CLS
1590 LET s#=""
1600 FOR i=1 TO LEN c#
1605 IF VAL c$(i)>n THEN PRINT #
1:"ERRORE I FOTOGRAMMI SONO SOLO
";2: PAUSE 0: GO TO 1340
1610 LET a$=a#+CHR$ VAL c$(i)
1620 NEXT i
1630 POKE 64042,LEN a$
1640 RANDOMIZE USR 64323
1650 PRINT AT 10,2:"PER TORNARE
AL MENU PREMI 0"
1660 INPUT "pausa tra la immagin
i (1-20)";t

```

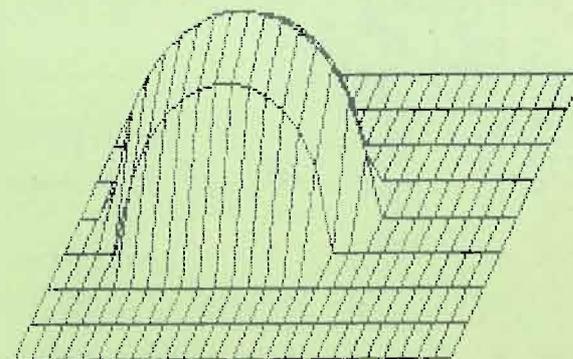
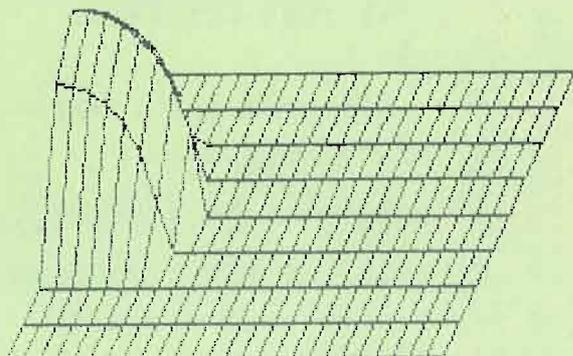
```

1670 IF INKEY#="0" THEN GO TO 13
30
1680 RANDOMIZE USR 64368
1690 FOR i=1 TO t: NEXT i
1700 GO TO 1670
1710 REM
1720 REM --salvataggio
1730 REM
1740 CLS
1745 PRINT "*****RICORDA:"
1750 PRINT "oltre alle immagini
viene salvata anche la rou
tine in linguaggio macchina"
1760 INPUT "nome file "; LINE c#
1770 SAVE c$CODE 63000-n*6144,n*
6144+2000
1780 GO TO 1330

```

Figura 1 - Esempio di fotogrammi da comporre per realizzare una interessante animazione con una semisfera. Per l'equazione, vedere l'articolo. I valori di xmin e xmax sono di -2 e 2 per tutti i 5 grafici, quelli di zmin e zmax sono zero, mentre per ottenere l'immagine:

- 1) ymin=0,3      ymax= 4,3
- 2) ymin=0,85     ymax= 3,15
- 3) ymin=-2        ymax= 2
- 4) ymin=-3,15    ymax= 0,85
- 5) ymin=-4,3     ymax= -0,3



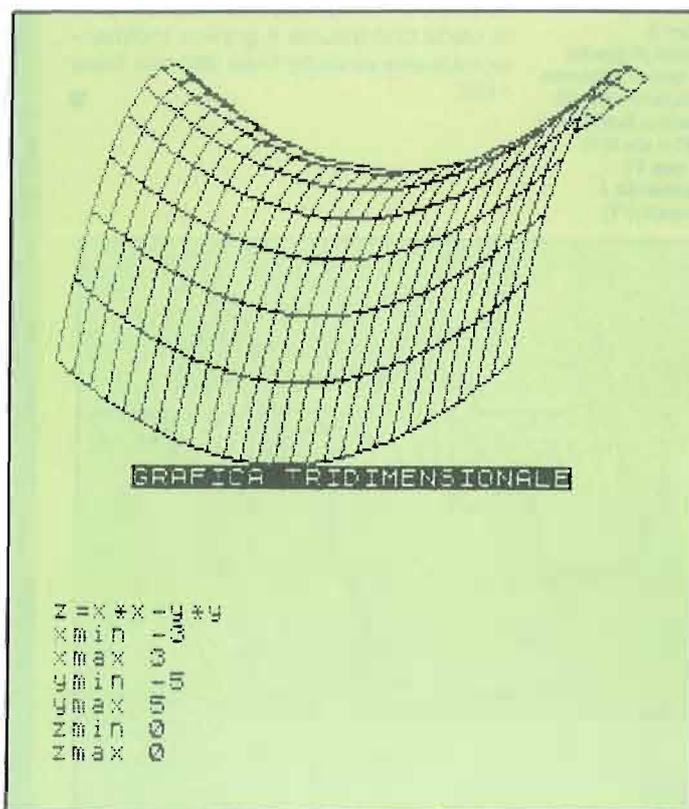
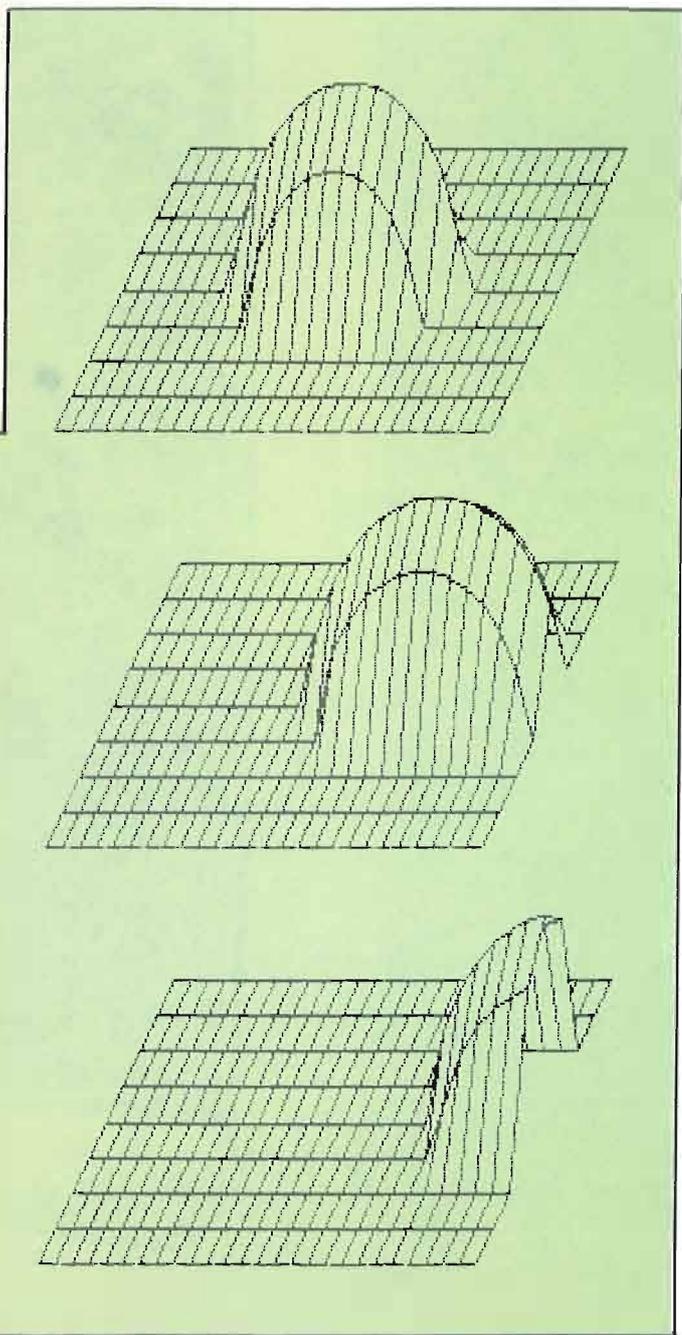
put; più piccolo è il suo valore e più preciso sarà il grafico (maggiore sarà però il tempo di elaborazione). Le variabili  $s_x$ ,  $s_y$ ,  $s_z$  servono per valutare il numero dei punti che si devono congiungere per ogni asse.

Per quanto riguarda le funzioni da inserire, bisogna fare alcune precisazioni: esse devono essere del tipo  $z = f(x,y)$  (in input si inseriscono solo  $f(x,y)$ ) e de-

vono avere come campo di definizione un dominio rettangolare; con un esempio capirete meglio cosa si intende dire. Consideriamo una sfera: essa non può essere direttamente inserita poiché la sua equazione è  $x^2 + y^2 + z^2 = 1$  (sfera con raggio unitario); tale equazione deve essere esplicitata rispetto a  $Z$ :  $Z = \text{SQR}(1-x^2-y^2)$  (semisfera positiva); ora si deve osservare che il campo di

definizione di una tale funzione non è certo rettangolare, cioè non è possibile determinare un rettangolo nel piano  $XY$  nei punti del quale la funzione risulti definita.

Infatti, considerando il rettangolo di vertici  $(1,1)$   $(1,-1)$   $(-1,1)$   $(-1,-1)$ , esso contiene non solo tutte le proiezioni dei punti della sfera, ma anche punti che non vi appartengono come il punto



**Figura 2 - Esempio di utilizzo del programma; la figura ottenuta è un parabolide a sella.**

$(1,1)$  per il quale risulta  $Z = \text{SQR}(-1)$  che non ha senso.

Nello studiare queste superfici (esempio: gli ellissoidi, di cui la sfera è un caso particolare), bisogna fare in modo che quando  $1-x^2-y^2 < 0$ , la funzione assuma un valore maggiore o uguale a zero. Si può operare come segue per disegnare la semisfera:  $Z = \text{SQR}((1-X * X - Y * Y) * (1-X * X - Y * Y) < 0)$  quando  $1-x^2-y^2$  è negativo l'espressione  $(1-X * X - Y * Y) < 0$  è uguale a zero. Visto che abbiamo analizzato a fondo questo tipo di superfici, possiamo comporne alcune in posizione diverse per simulare il loro spostamento (vedi figura 1).

Animazione di funzioni tridimensionali

## ZX SPECTRUM

A parte queste funzioni che richiedono particolari accorgimenti, il programma ben si adatta a studiarne tante altre (vedi figura 2).

Nel caso si incorra in qualche errore, non conviene dare il Run, altrimenti si deve ricaricare la routine in linguaggio macchina; in tal caso basta dare Goto 1330.

Se avete il 16 Kbyte potete utilizzare solo la parte che traccia il grafico tridimensionale che va dalla linea 260 alla linea 1150.

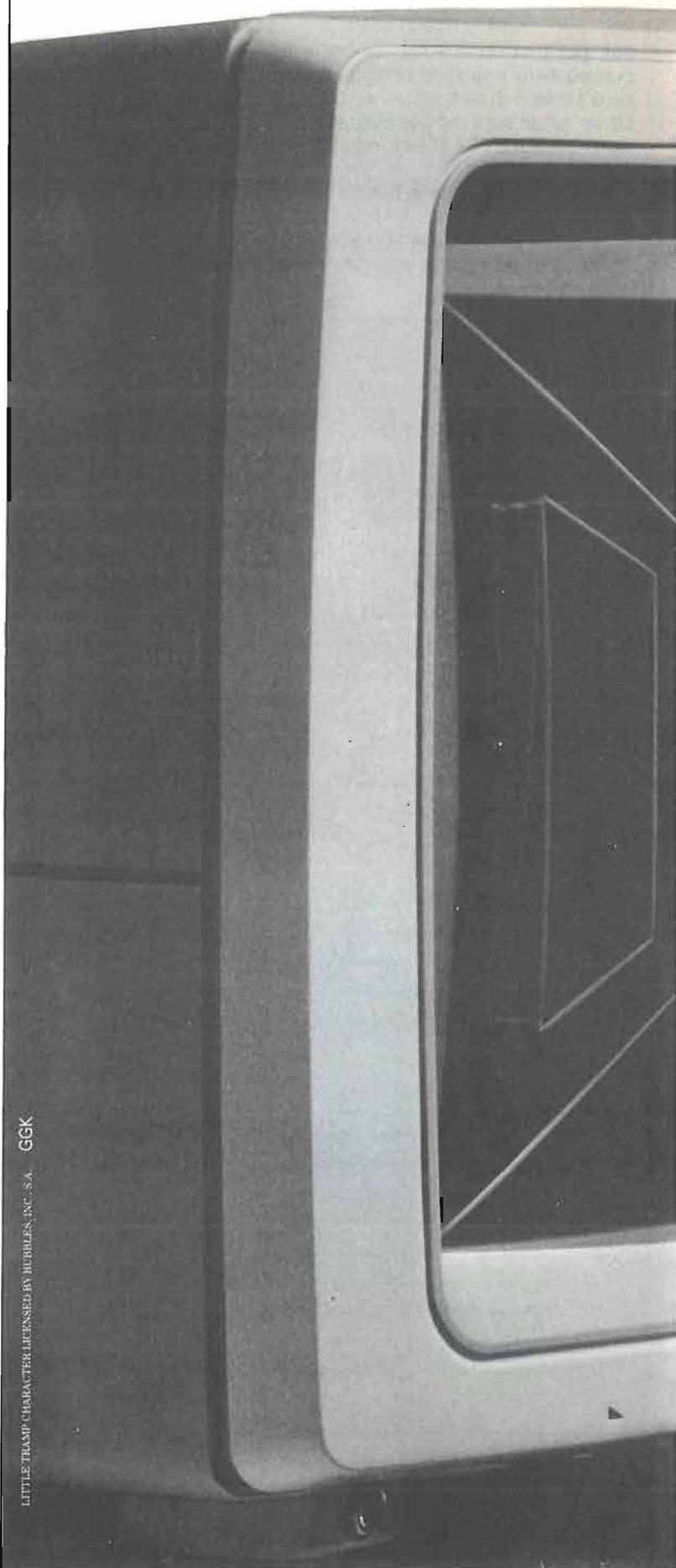
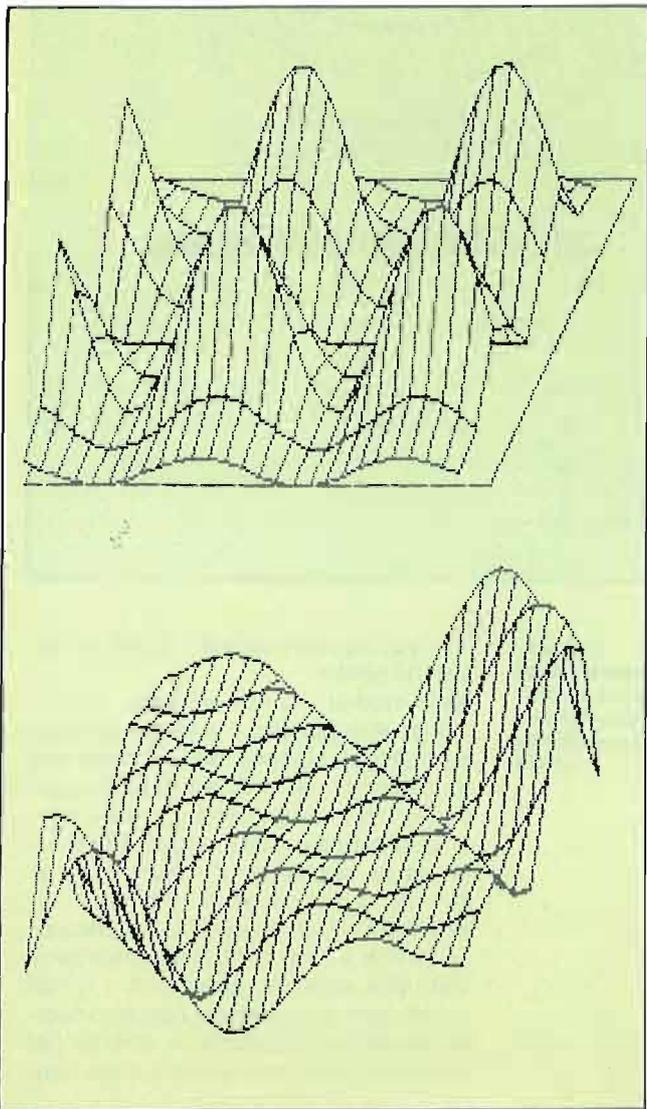
**Figura 3 - Esempi di grafici che ben si adattano ad essere animati.**

La prima funzione è:

$$z = (1 - \sin x) * (1 - \cos Y)$$

La seconda è:

$$z = \sin(X + Y)$$



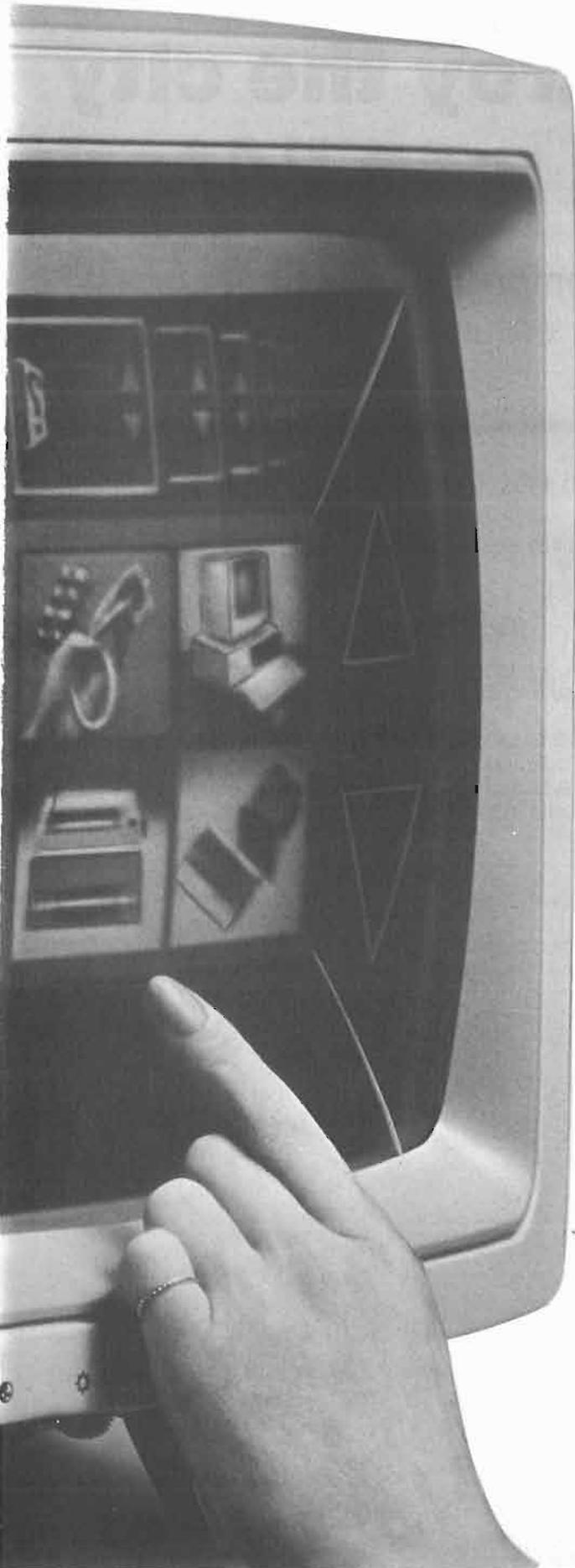
# più il Personal Computer IBM?

## Il Personal Computer Support Service, per esempio.

Il Concessionario IBM Personal Computer ha un servizio completamente nuovo da offrirti: il Personal Computer Support Service. È un sistema che ti permette di ottenere tutte le informazioni che vuoi sulla famiglia dei Personal Computer IBM. È composto da una serie di videodischi e da uno schermo con speciali caratteristiche, che risponde alle tue richieste al solo tocco di un dito. Basta toccare lo schermo e potrai scegliere direttamente dal menù quello che vuoi sapere in relazione al tuo settore di interesse e al tuo lavoro. In questo modo potrai ricevere qualunque tipo di informazione sulla famiglia dei Personal Computer IBM, oppure sui sistemi operativi, oppure sulle centinaia di programmi disponibili per i più disparati settori applicativi.

Ma c'è di più: il Personal Computer Support Service è collegato direttamente a una banca dati IBM europea. Così il Concessionario IBM Personal Computer potrà avere aggiornamenti in tempo reale su tutte le novità relative al Personal Computer IBM e ai suoi programmi e sarà in grado di rispondere subito alle più disparate domande su problemi specifici, che potrai essere tu stesso a porre. Grazie a questo servizio IBM il Concessionario IBM Personal Computer ti avrà fornito un'assistenza ancora migliore.

Se vuoi vedere da vicino ed usufruire del Personal Computer Support Service, vai dai Concessionari IBM (gli indirizzi sono sulle Pagine Gialle). Allora, hai visto che il Personal Computer IBM ha veramente qualcosa in più?



Desidero ricevere:

- Informazioni sul Personal Computer IBM e i suoi programmi.
- Gli indirizzi dei Concessionari IBM Personal Computer della mia regione
- Una visita o dimostrazione pratica di un concessionario

Nome e Cognome .....

Azienda .....

La mia attività è .....

Indirizzo .....

Spedisci questo tagliando a: IBM Italia Direzione Canali Esterni - Casella Post. 137 - 20090 Segrate Milano

**IBM**

**S**copo del gioco è atterrare, con un aereo, sulla città visualizzata dalla terza schermata del programma. Perché questo sia possibile, è necessario bombardare la città, distruggendola pressoché completamente.

La possibilità di scelta tra nove livelli di difficoltà, vi permette di variare la velocità a cui l'aereo si sposta sullo schermo. Le prime righe del programma ridefiniscono i caratteri usati per l'aereo, la bomba, i grattacieli. Nel frattempo c'è una schermata iniziale di presentazione del gioco (Gosub 2000). Alla linea 190 c'è l'istruzione per scegliere il livello di difficoltà, a seconda del quale variano sia la velocità d'esecuzione del gioco che le dimensioni della città (e poi anche il punteggio).

La costruzione della città avviene alle linee 240-280 e, conclusasi questa operazione, parte l'aeroplano (carattere % ridefinito). Nella linea 380 c'è l'istruzione di ritardo corrispondente al livello scelto, la 390 è attiva quando l'aereo raggiunge il margine destro del video. La riga 430 controlla se nella prossima posizione dell'aeroplano ci sono ostacoli; altrimenti la linea successiva verifica se è stato premuto il pulsante di fuoco del joystick o il tasto Start della tastiera, e in tal caso il programma rimanda alla routine di riga 1000.

Il loop principale è tutto racchiuso nelle linee 360-460 che abbiamo appena esaminato; gli altri blocchi di programma sono quelli delle righe 500-550 (la città è distrutta e potete atterrare), 700-910 (nel caso di insuccesso), 1100-1150 (distruzione parziale dei grattacieli) ed i già visti 1000-1060 (bomba sganciata) e 1000-2080 (intestazione).

Si potrebbero apportare ancora due modifiche; la prima nasce dalla constatazione che un giocatore che atterra a livello 1 può fare molti meno punti di uno che distrugge l'aeroplano dopo poco a livello 9. Si può quindi introdurre l'istruzione:

```
505 PTS = PTS + 1000 + INT(RND(1)*1001).
```

Un'ulteriore correzione può essere fatta alla riga 440, laddove compare:

```
...THEN BX = BY = Y + 1: eccetera,
```

si può inserire una variabile MK come segue:

# Destroy the city con l'Atari

Fremono i pulsanti per questo entusiasmante videogioco

di Daniela Cerù

```
...THEN MK = 3 + INT(RND(1) * 5):  
BX = X:BY = Y + 1...
```

Si provvederà quindi a modificare la linea 1120 come segue:

```
K = K + 1:IF K = MK THEN...  
con quel che segue.
```

In questo modo i piani dei grattacieli distrutti da una bomba varieranno da 3 a 7.

Un'ultima cosa: nella linea 2070 le scritture tra parentesi quadre indicano rispettivamente l'attivazione o la disattivazione del modo reverse (tasto Atari).

## Variabili presenti nel listato

**X, Y** - Coordinate dell'aeroplano.

**BX, BY** - Coordinate della bomba.

**Z** - Variabile di comodo, usata anche per rivelare il contenuto di una casella del video.

**L** - Livello di gioco.

**PTS** - Punti ottenuti.

**BM** - Uguale a 1 se il tasto di fuoco del joystick o lo Start sono stati premuti (altrimenti vale zero).

**I, J** - Indici dei cicli For-Next. ■

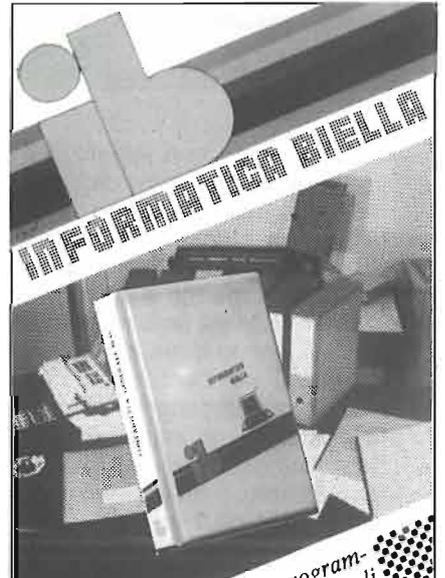
### Listato 1 - Il programma Destroy the city.

```
5 DATA 4,255,129,129,129,129,129,129,255,5,16,8,132,194,255,2,4,8
10 DATA 6,170,85,170,85,170,85,170,85,7,0,36,60,24,24,60,24,0,-1
15 GOSUB 2000
20 R=PEEK(106)-8
30 FOR I=0 TO 1023
40 POKE R*256+I,PEEK(57344+I)
50 NEXT I
60 READ X
70 IF X<0 THEN POKE 203,R:GOTO 150
80 FOR I=0 TO 7
90 READ Y
100 POKE R*256+(8*X)+I,Y
110 NEXT I
120 GOTO 60
150 OPEN #1,4,0,"K:"
160 GRAPHICS 0
170 SETCOLOR 2,3,0:SETCOLOR 4,8,0
180 POKE 752,1
190 PRINT "LIVELLO < 1-9 > ";
200 GET #1,L
210 IF L<49 OR L>57 THEN 200
220 L=L-48
230 POKE 756,PEEK(203)
240 PRINT CHR$(125):FOR I=1 TO 38
250 Y=INT(RND(1)*(L/2+8))+1
```

Destroy the city con l'Atari

## ATARI XL 800

```
260 FOR J=12 TO 22-Y STEP -1
270 POSITION I,J:PRINT "$"
280 NEXT J
290 FOR J=15 TO 0 STEP -1
300 SOUND 0,I*6+20,10,J
310 NEXT J
320 NEXT I
330 COLOR 94:L=27-L*3
340 PLOT 1,23:DRAWTO 39,23
350 X=1:Y=0:BM=0:PTS=0
360 POSITION X,Y:OX=X:OY=Y
370 PRINT "%";:X=X+1
380 FOR I=0 TO L:NEXT I
390 IF X=39 THEN X=1:Y=Y+1
400 IF X>25 AND Y=22 THEN 500
410 LOCATE X,Y,Z
420 POSITION OX,OY:PRINT " "
430 IF Z=ASC("$") THEN 700
440 IF (STRIG(0)=0 OR PEEK(53279)=6) AND BM=0 THEN BX=X:BY=Y+1:BM=1:K=0:
IF BY=23 THEN BY=22
450 IF BM=1 THEN GOSUB 1000
460 GOTO 360
500 POSITION 9,10:PRINT "OTTIMO ATTERRAGGIO !"
510 FOR I=0 TO 6000 STEP 20
520 SOUND 0,I,10,15
530 NEXT I
540 SOUND 0,0,0,0
550 GOTO 840
700 POSITION X-1,Y-1
710 PRINT CHR$(7);CHR$(124);CHR$(6)
720 POSITION X-1,Y
730 PRINT "-*-"
740 POSITION X-1,Y-1
750 PRINT CHR$(6);CHR$(124);CHR$(7)
760 FOR I=200 TO 10 STEP -10
770 FOR J=-5 TO 5
780 SOUND 0,J,120,15
790 NEXT J
800 COL=INT(RND(1)*15)
810 SETCOLOR 2,COL,2:SETCOLOR 4,COL,2
820 NEXT I
830 SOUND 0,0,0,0
840 POSITION 0,0:PRINT "HAI TOTALIZZATO ";PTS;" PUNTI"
850 IF MAX<PTS THEN MAX=PTS
860 POSITION 0,1:PRINT "IL PUNTEGGIO MASSIMO : ";MAX;" PUNTI"
870 POSITION 0,5:PRINT "Vuoi tentare ancora ?"
880 GET #1,Z
890 IF CHR$(Z)="$" THEN 160
900 IF CHR$(Z)<>"N" THEN 880
910 GRAPHICS 0:END
1000 POSITION BX,BY:PRINT " "
1010 BY=BY+1:IF BY=23 THEN BY=22
1020 LOCATE BX,BY,Z:POSITION BX,BY
1030 PRINT CHR$(39)
1040 IF Z=ASC("$") THEN GOSUB 1100
1050 IF BY=22 THEN POSITION BX,22:PRINT " ":BM=0
1060 RETURN
1100 POSITION BX,BY:PRINT "&":SOUND 0,230,120,15
1110 PTS=PTS+35-L:POSITION 0,0:PRINT PTS
1120 K=K+1:IF K=5 THEN BM=0:POSITION BX,BY:PRINT " "
1130 POSITION BX,BY:PRINT " "
1140 SOUND 0,0,0,0
1150 RETURN
2000 GRAPHICS 17:POSITION 2,1
2010 PRINT #6;"destroy the city"
2020 POSITION 0,4:PRINT #6;"VOI SIETE ALLA GUIDADI UN AEREOPLANO E"
2030 PRINT #6;"DOVRETE DISTRUGGERE LA CITTA`SOTTOSTANTE";
2040 PRINT #6;"PRIMA DI ATTERRARE EIMPADRONIRVENE."
2050 PRINT #6;"SCEGLIETEVI IL GRADODI DIFFICOLTA`PER"
2060 PRINT #6;"QUESTA IMPRESA E,MI RACCOMANDO,CALCOLATEBENE DOVE SPARATE"
2070 POSITION 2,19:PRINT #6;"[<REV.ON>]UN ATTIMO ANCORA[<REV.OFF>]"
2080 RETURN
```



**INFORMATICA BIELLA**

informatica biella crea programmi per ogni specifica esigenza e li realizza in collaborazione con esperti del settore verso il quale il programma è indirizzato.

informatica biella è garanzia di sicurezza, affidabilità e aggiornamento continuo del software.

**informatica biella propone:**

per APPLE II  
Contabilità generale 80CL Prodos  
Contabilità forfettaria  
multiaziendale  
Gestione Parrocchie  
(anche in MS/DOS)  
Gestione Alberghi  
Parcellazione  
Studi Legali

per MACINTOSH  
Contabilità generale  
Contabilità forfettaria  
multiaziendale

informatica biella è  
Rivenditore autorizzato  
Centro assistenza

apple computer inc.

informatica biella vi propone infine speciali interfacce adatti alle macchine per scrivere Olivetti e Adler.

Richiedete i programmi con il marchio **ib** presso i rivenditori autorizzati APPLE o direttamente a

**informatica biella**  
P.zza S. Paolo, 1 - 13051 Biella  
tel. (015) 29875 - 24181

**I**ncominciamo con l'analizzare le possibilità cromatiche dello Spectrum; come tutti sanno, questo computer è dotato di otto colori, ciascuno dei quali ha due tonalità di luminosità. Lo schermo è diviso in 768 caratteri (24 linee, 32 colonne) e ognuno di essi è dotato di un colore per lo sfondo e uno per l'inchiostro; inoltre, ogni carattere può godere di due luminosità e può essere lampeggiante. Lo sfondo, l'inchiostro, la luminosità e il lampeggiamento, costituiscono gli attributi di un carattere. I valori di tali attributi sono memorizzati a partire dalla locazione 22528 per un'estensione di 768 byte; ogni byte contiene gli attributi di un carattere.

Se eseguite delle Poke in tale zona di memoria, vedrete cambiare attributi a dei caratteri sul video; ciò avviene perché un particolare integrato dello Spectrum, la ULA, provvede periodicamente ad analizzare quelle locazioni e a fornire di conseguenza informazioni ad un altro integrato (LM 1889), che genererà l'immagine video.

Per modificare gli attributi in un determinato punto del video di coordinate x, y, dovete eseguire la Poke nella locazione numero  $22528 + X + Y \times 32$ .

Il numero da inserire dipende dal tipo di attributi che si desidera; tale valore si calcola, come illustrato a pagina 88 del manuale, nel seguente modo:  $Ink + Paper \star 8 + Bright \star 64 + Flash \star 128$ , dove Ink e Paper assumono valori da 0 a 7, mentre Bright e Flash assumono valori 0 o 1. Consultate a riguardo la tabella 1.

Ad esempio, affinché il carattere di coordinate 10,5 abbia lo sfondo rosso e l'inchiostro giallo, dovete fare Poke 22698,22

Sfruttando quanto detto, possiamo realizzare una routine in linguaggio macchina che gestisca il colore a nostro piacimento e che possa essere sfruttata in molti programmi.

Prima di passare ad analizzare la routine, vi illustriamo un programma scritto in BASIC atto alla sua memorizzazione (vedi listato 1).

Con questo caricatore dovete introdurre i codici decimali delle istruzioni Assembly e al termine un valore indicato con Check.

Questo numero rappresenta la somma dei codici di tutte le istruzioni inserite.

# La gestione del video

## Produciamo delle routine e dei trucchi per l'output su video

di Carlo Cappelli

Nel caso commettiate qualche errore è facile che la somma dei codici da voi inseriti differisca da quella effettiva; in tal caso, il programma vi segnalerà la presenza di qualche sbaglio. In questo modo si vengono a ridurre le possibilità di non funzionamento delle routine in linguaggio macchina. Detto questo, passiamo ad analizzare la routine: essa serve per modificare istantaneamente gli attributi di una porzione rettangolare di video di dimensioni a piacere, una cosa simile non è possibile col BASIC. Per analizzare questa routine dovete inserire i codici decimali del dump 1.

Il dump 1 è scritto per il 16 Kbyte; se avete il 48 Kbyte dovete inserire come RAMtop il valore di 63.999 e modificare i codici segnati con l'asterisco da 125 a 250; al termine inserite come Check il valore 12.928.

Chi volesse collocare la routine in posizioni diverse, deve modificare opportunamente i codici contrassegnati con l'asterisco. I parametri per il funzionamento del programma in linguaggio macchina non vengono inseriti con delle scomode Poke, ma semplicemente memorizzati nella stringa A\$ con la seguente sintassi:

PAPER								
nero	blu	rosso	viola	verde	azzurro	giallo	bianco	
0	8	16	24	32	40	48	56	nero
1	9	17	25	33	41	49	57	blu
2	10	18	26	34	42	50	58	rosso
3	11	19	27	35	43	51	59	viola
4	12	20	28	36	44	52	60	verde
5	13	21	29	37	45	53	61	azzurro
6	14	22	30	38	46	54	62	giallo
7	15	23	31	39	47	55	63	bianco

Tabella 1 - Valori da inserire nella mappa degli attributi per ottenere determinati sfondi e inchiostri: se volete che il carattere lampeggi dovete aggiungere a tale valore 128, mentre se desiderate aumentare la luminosità aggiungete 64.

# ZX SPECTRUM

Listato 1 - Programmino per caricare i codici del linguaggio macchina.

```

10 INPUT "RAMTOP=";A
20 CLEAR A: LET R=PEEK 23730+2
50 #PEEK 23731
30 PRINT "RAMTOP=";A
40 LET CHECK=0: LET I=R+1
40 INPUT A: IF A>500 THEN GO T
0 45 PRINT A;TAB 5;"=";I: POKE
I,A
1 50 LET CHECK=CHECK+A: LET I=I+
1 60 GO TO 40
70 IF A>>CHECK THEN PRINT FLAG
H 1;"ERRORE RICONTROLLA": STOP

80 PRINT ""CHECK =";A
90 PRINT ""ORA DEVI SALVARE L
A ROUTINE": INPUT "NOME FILE ";
LINE S#
100 PRINT "NOME FILE ";S#
110 SAVE S#CODE R+1,I-R+10
    
```

```

00000074 = 000107
00000075 = 000108
00000076 = 000109
* 00000077 = 000110
00000078 = 000111
00000079 = 000112
00000080 = 000113
00000081 = 000114
00000082 = 000115
* 00000083 = 000116
00000084 = 000117
00000085 = 000118
00000086 = 000119
00000087 = 000120
00000088 = 000121
00000089 = 000122
00000090 = 000123
* 00000091 = 000124
00000092 = 000125
00000093 = 000126
* 00000094 = 000127
00000095 = 000128
* 00000096 = 000129
00000097 = 000130
00000098 = 000131
00000099 = 000132
* 000100 = 000133
000101 = 000134
000102 = 000135
000103 =
000104 =
000105 =
000106 =
000107 =
000108 =
000109 =
000110 =
000111 =
000112 =
000113 =
000114 =
000115 =
000116 =
000117 =
000118 =
000119 =
000120 =
000121 =
000122 =
000123 =
000124 =
000125 =
000126 =
000127 =
000128 =
000129 =
000130 =
000131 =
000132 =
000133 =
000134 =
000135 =
CHECK =11678
    
```

```

RAMTOP=31000
00000000 = 00000000
00000001 = 00000001
00000002 = 00000002
00000003 = 00000003
00000004 = 00000004
00000005 = 00000005
00000006 = 00000006
00000007 = 00000007
00000008 = 00000008
00000009 = 00000009
00000010 = 00000010
00000011 = 00000011
00000012 = 00000012
00000013 = 00000013
00000014 = 00000014
00000015 = 00000015
00000016 = 00000016
00000017 = 00000017
00000018 = 00000018
00000019 = 00000019
00000020 = 00000020
00000021 = 00000021
00000022 = 00000022
00000023 = 00000023
00000024 = 00000024
00000025 = 00000025
00000026 = 00000026
00000027 = 00000027
00000028 = 00000028
00000029 = 00000029
00000030 = 00000030
00000031 = 00000031
00000032 = 00000032
00000033 = 00000033
00000034 = 00000034
00000035 = 00000035
00000036 = 00000036
00000037 = 00000037
00000038 = 00000038
00000039 = 00000039
00000040 = 00000040
00000041 = 00000041
00000042 = 00000042
00000043 = 00000043
00000044 = 00000044
00000045 = 00000045
00000046 = 00000046
00000047 = 00000047
00000048 = 00000048
00000049 = 00000049
00000050 = 00000050
00000051 = 00000051
00000052 = 00000052
00000053 = 00000053
00000054 = 00000054
00000055 = 00000055
00000056 = 00000056
00000057 = 00000057
00000058 = 00000058
00000059 = 00000059
00000060 = 00000060
00000061 = 00000061
00000062 = 00000062
00000063 = 00000063
00000064 = 00000064
00000065 = 00000065
00000066 = 00000066
00000067 = 00000067
00000068 = 00000068
00000069 = 00000069
00000070 = 00000070
00000071 = 00000071
00000072 = 00000072
00000073 = 00000073
00000074 = 00000074
00000075 = 00000075
00000076 = 00000076
00000077 = 00000077
00000078 = 00000078
00000079 = 00000079
00000080 = 00000080
00000081 = 00000081
00000082 = 00000082
00000083 = 00000083
00000084 = 00000084
00000085 = 00000085
00000086 = 00000086
00000087 = 00000087
00000088 = 00000088
00000089 = 00000089
00000090 = 00000090
00000091 = 00000091
00000092 = 00000092
00000093 = 00000093
00000094 = 00000094
00000095 = 00000095
00000096 = 00000096
00000097 = 00000097
00000098 = 00000098
00000099 = 00000099
000100 = 000100
000101 = 000101
000102 = 000102
000103 = 000103
000104 = 000104
000105 = 000105
000106 = 000106
000107 = 000107
000108 = 000108
000109 = 000109
000110 = 000110
000111 = 000111
000112 = 000112
000113 = 000113
000114 = 000114
000115 = 000115
000116 = 000116
000117 = 000117
000118 = 000118
000119 = 000119
000120 = 000120
000121 = 000121
000122 = 000122
000123 = 000123
000124 = 000124
000125 = 000125
000126 = 000126
000127 = 000127
000128 = 000128
000129 = 000129
000130 = 000130
000131 = 000131
000132 = 000132
000133 = 000133
000134 = 000134
000135 = 000135
    
```

Dump 1 - Codici decimali per realizzare la routine per la gestione del colore. La routine gira sia sul 16 Kbyte che sul 48 Kbyte, per quest'ultimo occorre modificare le locazioni segnate con l'asterisco nel modo illustrato nell'articolo.

```
LET A$ = CHR$ X + CHR$ Y + CHR$ LX
+ CHR$ LY + CHR$ F + CHR$ B
+ CHR$ P + CHR$ I
```

dove X e Y rappresentano le coordinate dello spigolo in alto a sinistra della superficie rettangolare che si vuole colorare; LX e LY devono esprimere la lunghezza dei lati di tale rettangolo; F esprime il lampeggiamento. Se si vuole che la superficie da noi definita lampeggi, si inserisca 1, altrimenti 0; B rappresenta la luminosità, ed è necessario inserire 0 o 1; P esprime il colore dello sfondo, mentre I quello dell'inchiostro, tali valori devono andare da 0 a 7 secondo i colori scelti.

Per attivare la routine, date Randomize Usr 32010 se avete il 16 Kbyte, mentre per il 48 Kbyte usate Randomize Usr 64010. Il programma del listato 2, ad esempio, colora un rettangolo con lo spigolo in alto a sinistra di coordinate X=3, Y=4 e avente i lati di lunghezza 18 e 8; dopo la chiamata alla routine lo sfondo della superficie rettangolare apparirà verde (P=4), mentre le scritte sa-



ranno bianche (I=7). Flash e Bright sono posti a zero. Il programma in linguaggio macchina non controlla l'esattezza dei dati inseriti, quindi, spetta a voi, assicurarvi che siano tali; soprattutto è importante che la superficie rettangolare non fuoriesca dallo schermo, quindi è opportuno definire i valori di X, Y, LX, LY in modo che non si verifichi una tale circostanza. Ad esempio, sarebbe errato dare a un rettangolo di spigolo 5,7 una lunghezza LX maggiore di 27 e un LY maggiore di 17 (si possono colorare le due righe dello schermo basso). Analizziamo ora il funzionamento della routine. Appena lanciata, essa provvede a trovare nell'area delle variabili la stringa A\$ di 8 caratteri. L'indirizzo di partenza di tale area è puntato dalla variabile di sistema VARS; da quel punto in poi si cerca la stringa il cui inizio è memorizzato come 65, 8, 0; 65 è il

codice di A\$ mentre 8 e 0 esprimono la lunghezza della stringa stessa. Nel caso in cui la stringa non sia in memoria o non abbia la giusta lunghezza e dopo aver analizzato tutta la RAM, il programma salta a una routine di errore della ROM (Call 5572). Trovata la stringa, ne trasferisce i primi 4 byte a partire dalla locazione 32000; a questo punto esegue i calcoli sulla posizione e sui colori, secondo le regole illustrate all'inizio dell'articolo, e infine procede alla stampa nell'area degli attributi. Tutto ciò in pochissimo tempo.

Se volete utilizzare nei vostri programmi la subroutine per la ricerca di una stringa, ricordiamo che essa inizia dalla locazione 32004 e termina a 32033, restituendo nel registro HL la posizione del contenuto della stringa A\$; se la vostra stringa non ha 8 caratteri, dovete modificare adeguatamente il valore del-

la locazione 32026.

Lo Spectrum possiede solo 8 colori, però, con alcuni artifici, possiamo ottenerne molti altri; il metodo che vi proponiamo è estremamente semplice: basta realizzare un carattere UDG a punti alterni (vedi listato 3), assegnando colori diversi allo sfondo e all'inchiostro; il carattere assumerà una colorazione somma delle due date. Questo effetto è dovuto al fatto che pixel estremamente vicini e di colori differenti, non risultano ben distinti al nostro occhio, dando così l'effetto desiderato. Per rendervi meglio conto di questo, provate il programma BASIC del listato 3.

Fino ad ora abbiamo trattato la parte alta dello schermo, adesso parleremo della parte bassa e del bordo. La variabile di sistema che contiene gli attributi di tali zone è BORDCR, e si trova nella locazione 23624. I primi tre bit di tale

**Listato 2 - Esempio di funzionamento della routine per la gestione del colore.**

```

10 REM color
20 IF PEEK 32030=126 THEN GO TO 120
30 PRINT "devi caricare il programma in linguaggio macchina per la gestione del colore"
40 PRINT "prepara il nastro e premi un tasto"
40 PAUSE 0
100 CLEAR 31999
110 LOAD ""CODE
120 PRINT "premi un tasto"
125 PAUSE 0
130 REM x,y,lx,ly,flash,bright,paper,ink
200 LET a$=CHR$ 3+CHR$ 4+CHR$ 18+CHR$ 8+CHR$ 0+CHR$ 0+CHR$ 4+CHR$ 7
210 RANDOMIZE USR 32010

```

**Listato 3 - Con questo semplice programma potete creare nuove colorazioni.**

```

10 REM **** MULTICOLORE ****
20 FOR i=0 TO 6 STEP 2
30 POKE USR "a"+i,BIN 01010101
40 POKE USR "a"+i+1,BIN 101010
50 NEXT i
60 INPUT "colore sfondo ";s
70 INPUT "colore inchiostro ";c
80 INPUT "luminosità (0-1) ";l
90 LET a$=CHR$ 144+CHR$ 144+CHR$ 144+CHR$ 144+CHR$ 144+CHR$ 144

```

```

100 LET a$=a$+a$+a$+a$
110 FOR i=1 TO 20
120 PRINT BRIGHT i; PAPER s; INK c;a$
130 NEXT i
150 INPUT "vuoi cambiare colore s/n ";s$
160 IF s$="s" OR s$="S" THEN CLS : GO TO 60

```

**Listato 4 - Con questo programma potete realizzare interessanti effetti visivi modificando velocemente i colori del border.**

```

10 REM **** border ****
20 CLEAR 31999
30 PRINT "questa routine è completamente rilocabile; tuttora è memorizzata a partire dalla locazione 32000"
40 DATA 243,14,6,8,255,120,211,254,16,251,13,32,246,251,201
50 FOR i=32000 TO 32014
60 READ a
70 POKE i,a
80 NEXT i
90 PRINT ""premi un tasto"
100 PAUSE 0
120 FOR i=1 TO 50
130 PAUSE 1
140 RANDOMIZE USR 32000
150 NEXT i
155 BORDER 7
160 PRINT ""vuoi rivedere l'effetto"
170 INPUT LINE s$
180 IF s$="s" OR s$="S" THEN GO TO 90

```

Seguito figura 2.

```

00000001 010 4 125 . . . . . jr nc,000004
00000004 000 0 . . . . . ld e,a
00000007 000 0 . . . . . add hl,0
00000010 000 0 . . . . . and nz,000007
00000013 004 025 . . . . . ld a,(000000)
00000016 004 125 . . . . . jr nc,000004
00000019 000 0 . . . . . ld e,a
00000022 000 0 . . . . . add hl,0
00000025 000 0 . . . . . add de,0
00000028 000 0 . . . . . add hl,0
00000031 000 0 . . . . . ld a,(000000)
00000034 000 0 . . . . . ld a,(000000),a
00000037 000 0 . . . . . ld a,(000000)
00000040 000 0 . . . . . ld a,(000000),a
00000043 000 0 . . . . . ld a,(000000)
00000046 000 0 . . . . . ld a,(000000),c
00000049 000 0 . . . . . inc hl
00000052 000 0 . . . . . ld a,02114
00000055 000 0 . . . . . ld a,0
00000058 000 0 . . . . . ld a,0
00000061 000 0 . . . . . ld a,0
00000064 000 0 . . . . . add b,e
00000067 000 0 . . . . . add e,a
00000070 000 0 . . . . . add hl,de
00000073 000 0 . . . . . ld a,(000000)
00000076 000 0 . . . . . dec a
00000079 000 0 . . . . . ret (000000),a
00000082 004 030 . . . . . jr 000110

```

Figura 3 - Listato Assembly del programma in linguaggio macchina per la gestione dei fotogrammi.

```

040000 0 . . . . . nop
040001 0 . . . . . nop
040002 0 . . . . . nop
040003 0 . . . . . nop
040004 0 . . . . . nop
040005 0 . . . . . nop
040006 0 . . . . . nop
040007 0 . . . . . nop
040008 0 . . . . . nop
040009 0 . . . . . nop
040010 0 . . . . . nop
040011 0 . . . . . nop
040012 0 . . . . . nop
040013 0 . . . . . nop
040014 0 . . . . . nop
040015 000 0 . . . . . ld a,0
040017 004 03 . . . . . jr 040021
040019 000 18 . . . . . ld a,18
040021 005 040 000 . . . . . jr 040040
040024 0 . . . . . nop
040025 040 070 000 . . . . . ld hl,(000007)
040026 000 090 000 . . . . . ld a,(000042)

040031 000 0 . . . . . cp h
040034 000 041 . . . . . jr c,040019
040035 000 0 . . . . . ld a,(hl)
040036 004 05 . . . . . inc hl
040037 000 0 . . . . . cp 0

```

```

040038 000 044 . . . . . jr nz,040028
040040 000 100 . . . . . ld a,(hl)
040041 000 40 . . . . . cp 0
040043 000 045 . . . . . jr nz,040034
040045 000 0 . . . . . inc hl
040048 000 0 . . . . . ld a,(hl)
040047 000 0 . . . . . and a,0
040048 000 0 . . . . . jr nz,040034
040050 000 0 . . . . . inc hl
040051 000 0 . . . . . ret
040057 000 0 . . . . . ld a,7
040054 000 40 000 . . . . . ld (040042),a
040057 000 050 000 . . . . . call 040025
040060 000 17 0 . . . . . ld bc,7
040063 000 17 0 . . . . . ld de,040000
040067 000 170 . . . . . ld ir
040071 000 0 . . . . . ld a,(040002)
040071 000 0 . . . . . ld b,a
040077 000 0 . . . . . ld a,(040003)
040077 000 0 . . . . . ld e,a
040077 000 0 . . . . . ld d,0
040081 000 0 . . . . . ld hl,0
040081 000 0 . . . . . add hl,de
040084 000 47 000 . . . . . ld (040007),hl
040087 000 0 . . . . . ld b,0
040090 000 0 . . . . . ld e,0
040091 000 0 . . . . . ex de,hl
040093 000 0 . . . . . add hl,de
040093 000 0 . . . . . djnz 040092
040094 000 4 000 . . . . . ld (040009),hl
040098 000 4 000 . . . . . ld a,(040004)
041001 007 0 . . . . . and a
041002 007 0 . . . . . jr z,041009
041004 007 01 7 050 . . . . . ld de,(040007)
. . . . . . . . . . . add hl,de
041009 000 34 11 000 . . . . . ld (040011),hl
041100 000 0 000 . . . . . ld a,(040006)
041105 000 71 . . . . . ld b,a
041106 000 0 . . . . . ex de,hl
041117 000 0 . . . . . ld hl,0
041120 000 0 . . . . . add hl,de
041121 000 050 . . . . . djnz 041120
041123 000 17 040 . . . . . ld de,00004
041125 000 0 . . . . . ex de,hl
041127 000 7 00 . . . . . sbc hl,de
041129 000 100 0 . . . . . ld a,(000054)
041130 000 0 . . . . . cp r
041133 000 19 050 . . . . . jr nc,040015
041136 000 0 . . . . . push hl
041137 000 0 . . . . . ld a,(040005)
041140 004 1 000 . . . . . cp 1
041140 000 0 . . . . . jr z,041153
041144 001 . . . . . dec a
041145 001 . . . . . ld b,a
041146 007 01 11 050 . . . . . ld de,(040011)
. . . . . . . . . . . add hl,de
041150 000 10 050 . . . . . djnz 041150
041153 004 13 050 . . . . . ld (040013),hl
041156 000 0 . . . . . pop bc
041157 000 11 . . . . . dec bc
041158 000 0 . . . . . ld a,(040006)
041161 000 0 050 . . . . . ld (040008),a
041164 000 1 . . . . . ret
041165 0 . . . . . nop
041166 000 0 . . . . . ld a,0

```

# ZX SPECTRUM

do attentamente il modo in cui vengono caricati gli screen. Le difficoltà nascono dal modo in cui sono suddivisi i blocchi: essi sono frazionati in otto fette di 256 byte ciascuna, e ogni fetta in fettine di 32 byte. Ogni carattere viene ad occupare 8 fettine una sopra l'altra sul video, ma distanti 256 byte le une dalle altre nella memoria di schermo. Questo fatto rende difficile la stampa sullo schermo. Per meglio capire le difficoltà da superare, potete provare le seguenti Poke. Con Poke 16384,255 stamperete un segmento lungo un carattere e alto un pixel (ricordate che 255 in binario è 11111111 se usate 240 al posto di 255 otterrete un segmento di lunghezza dimezzata rispetto al precedente; 240 in binario è infatti 11110000). Per visualizzare un segmento sotto al precedente, dovete fare Poke 16640,255; 256 byte

più avanti della precedente Poke. Se invece desiderate visualizzare il solito segmento a fianco del primo, fate Poke 16385,255.

Per ora abbiamo stampato sulla prima riga; per stampare sulla seconda, basta fare Poke 16416,255. Da queste osservazioni empiriche possiamo concludere che segmenti di pixel contigui orizzontalmente sullo schermo lo sono anche nella memoria di schermo; mentre la distanza tra due file di pixel contigue verticalmente è di 256 byte; infine, la distanza tra la prima fila di pixel di un carattere e la stessa del carattere successivo è di 32 byte.

Per lavorare sul secondo blocco dovete aggiungere a 16384,2048 cioè partire da 18432. Le regole di stampa sono le stesse del primo blocco; ad esempio, per visualizzare il solito segmento all'ini-

zio della nona riga provate Poke 18432,255.

Per eseguire la stessa operazione sul terzo blocco dovete operare a partire da 20480 (= 18432 + 2048); Poke 20480,255.

A questo punto possiamo trasformare in una relazione matematica quanto detto. Chiamiamo X e Y le coordinate del carattere da stampare, (con X compreso tra 0 e 31; Y tra 0 e 24) per sapere l'indirizzo in cui posizionare il primo segmento del carattere da stampare (i sette successivi segmenti disteranno dal primo 256 byte), basta eseguire il seguente calcolo:

$$16384 + X + \text{INT}(Y/8) \star 2048 + (Y - \text{INT}(Y/8) \star 8) \star 32$$

Normalmente in ogni Print il programma salta a una routine della ROM per

Seguito figura 3.

```

04168 50 160 251 ...ld (04416),a
04171 50 163 251 ...ld (04419),a
04174 50 37 251 ...ld (04203),a
04177 50 46 251 ...ld (04206),a
04180 50 6 250 ...ld a,(04006)
04183 50 7 250 ...and a
04184 50 10 250 ...jp z,04010
04187 50 0 04 ...ld hl,16384
04190 50 0 ...ld d,0
04193 50 0 250 ...ld a,(04000)
04196 50 0 ...ld e,a
04199 50 0 ...add hl,de
04202 50 1 250 ...ld a,(04001)
04205 50 0 ...ld e,a
04208 50 32 ...ld b,32
04211 50 0 ...add hl,de
04214 50 1 250 ...djnz 04203
04217 50 7 ...ld de,1700
04220 50 4 0 ...cp 0
04223 50 0 ...jr c,04219
04226 50 0 ...add hl,de
04229 50 4 16 ...cp 16
04232 50 1 ...jr c,04219
04235 50 0 ...add hl,de
04238 50 7 01 10 250 ...ld de,(04013)
04241 50 0 ...ld a,(04003)
04244 50 7 250 ...ld a,(04007),a
04247 50 0 250 ...ld a,(04002)
04250 50 15 251 ...ld (04415),a
04253 50 15 04 251 ...jp 04410
04256 50 0 ...nop
04259 50 0 ...nop
04262 50 5 0 0 ...ld (20010),a
04265 50 10 0 10 ...jp 4007
04268 50 0 ...nop
04271 50 0 ...nop
04274 50 0 ...nop

```

```

04040 50 0 ...nop
04043 50 0 ...nop
04046 50 0 ...nop
04049 50 0 ...nop
04052 50 0 ...nop
04055 50 4 250 ...ld a,(04004)
04058 50 107 0 ...and a
04061 50 40 5 ...jr z,04316
04064 50 1013 0 ...push de
04067 50 0 ...nop
04070 50 0 0 ...ld hl,22520
04073 50 1 250 ...ld a,(04001)
04076 50 0 ...ld e,a
04079 50 0 ...ld d,0
04082 50 0 2 ...ld b,32
04085 50 0 ...add hl,de
04088 50 1 250 ...djnz 04274
04091 50 0 250 ...ld e,a
04094 50 0 ...add hl,de
04097 50 0 250 ...ld a,(04003)
04100 50 7 250 ...ld (04007),a
04103 50 0 ...pop de
04106 50 2 250 ...ld a,(04002)
04109 50 0 ...ld c,a
04112 50 0 ...ex de,hl
04115 50 176 0 ...ldir
04118 50 0 ...ex de,hl
04121 50 0 ...push de
04124 50 0 ...ld e,a
04127 50 0 ...ld a,32
04130 50 0 ...ld d,0
04133 50 147 0 ...sub e,a
04136 50 5 ...ld e,a
04139 50 0 ...add hl,de
04142 50 0 ...pop de
04145 50 7 250 ...ld a,(04007)
04148 50 1 ...dec a
04151 50 7 250 ...ld (04007),a
04154 50 22 0 ...jr nz,04209

```

eseguire questi calcoli (vedi **Personal Software** n. 19).

Sfruttando tali routine non si riesce ad ottenere velocità incredibili e allora bisogna riformularle ex-novo in linguaggio macchina.

Questa necessità deve essere soddisfatta se vogliamo realizzare un programma per la gestione di fotogrammi e per le animazioni, oppure se vogliamo dotare il nostro Spectrum degli sprite.

Tutto il discorso iniziale ci servirà per meglio capire come realizzare tali programmi.

Incominciamo col parlare della routine per la gestione dei fotogrammi.

Sarebbe semplice trasferire in blocco tutta la memoria video, più complicato è trasferire in memoria e poi rivisualizzare una finestra (porzione rettangolare del-

lo schermo) con dimensioni a piacere. Il trasferimento in blocco dello schermo impone inutili sprechi di memoria con la conseguente limitazione sul numero di fotogrammi memorizzati.

Su un 16 Kbyte è possibile memorizzare un solo video completo, mentre è possibile conservare 3 finestre da 10 caratteri per 20; inoltre, il lavorare su finestre permette la gestione indipendente di varie porzioni di schermo. La routine permette inoltre di inserire la sequenza con cui i fotogrammi preparati devono ricomparire sullo schermo, permettendo così interessanti animazioni.

Per caricare il linguaggio macchina potete usare il programmino del listato 1; i codici da inserire sono quelli del dump 2 se avete il 48 Kbyte; mentre per il 16 Kbyte dovete inserire come RAMtop il

valore 31.999 e modificare le locazioni segnate con l'asterisco nel seguente modo: al posto di 250 inserite 125 e al posto di 251 126.

Al termine dovete inserire come check il numero 37.629.

I parametri da comunicare al linguaggio macchina devono essere memorizzati nella stringa A\$ rispettando la seguente sintassi:

```
LET A$=CHR$X + CHR$Y + CHR$LX
+CHR$ LY +CHR$ C +CHR$ I
+CHR$N
```

dove X e Y rappresentano le coordinate dello spigolo in alto a sinistra della finestra che desideriamo individuare; mentre LX e LY esprimono la lunghezza dei lati di tale finestra. Il parametro C deve valere 0 se non vi interessa memorizzare

Seguito figura 3.

```

64316 58 5 250 .....ld a, (64006)
64319 51 .....dec a
64320 50 5 250 .....ld a, (64006), a
64323 237 83 13 250 .....ld (64013), de
.....ld (64013), de
64327 201 .....ret
64328 205 25 250 .....call 64025
64331 34 9 250 .....ld (64009), hl
64334 58 42 250 .....ld a, (64042)
64337 50 5 250 .....ld (64005), a
64340 71 5 250 .....ld b, a
64341 58 5 250 .....ld a, (64005)
64344 0 .....nop
64345 04 .....ld e, (hl)
64346 107 .....cp e
64347 210 10 250 .....jp c, 64019
64350 355 .....inc hl
64351 116 .....djnz 64345
64353 202 250 .....ld a, 255
64355 50 180 251 .....ld (64416), a
64358 50 180 251 .....ld (64410), a
64361 50 37 251 .....ld (64003), a
64364 50 40 251 .....ld (64206), a
64367 201 .....ret
64368 42 9 250 .....ld hl, (64009)
64371 70 .....ld b, (hl)
64372 35 .....inc hl
64373 34 9 250 .....ld (64006), hl
64376 42 170 92 .....ld hl, (20730)
64379 35 .....inc hl
64380 207 91 11 250 .....ld de, (64011)
.....ld de, (64011)
64384 5 .....dec b
64385 12 9 .....ld a, b
64386 107 .....and a
64387 40 0 .....jr z, 64392
64389 25 .....add hl, de
64390 19 253 .....djnz 64389

```

```

64392 34 10 250 .....ld (64013), hl
64395 107 250 .....call 64187
64398 50 5 250 .....ld a, (64005)
64401 51 .....dec a
64402 50 5 250 .....ld (64005), a
64405 200 72 251 .....jp z, 64328
64408 201 .....ret
64409 0 .....nop
64410 200 9 .....push hl
64411 200 0 .....ld a, 0
64413 200 9 .....push hl
64414 14 15 .....ld c, 15
64416 200 5 .....ex de, hl
64417 200 7 176 .....ldir
64419 200 5 .....ex de, hl
64420 200 9 .....pop hl
64421 200 5 .....inc b
64422 201 .....dec a
64423 244 .....jr nz, 64413
64425 200 9 .....pop hl
64426 120 9 .....ld a, l
64427 200 4 204 .....cp 204
64429 50 9 .....jr c, 64440
64431 14 9 7 .....ld bc, 1024
64434 207 74 .....adc hl, bc
64436 50 0 .....ld b, 0
64438 24 5 .....jr 64446
64440 14 32 .....ld c, 32
64442 0 .....nop
64443 16 7 .....and a
64444 207 74 .....adc hl, bc
64446 50 7 250 .....ld a, (64007)
64449 51 .....dec a
64450 50 7 250 .....ld (64007), a
64453 30 211 .....jr nz, 64410
64455 105 253 250 .....jp 64253

```







i colori delle finestre (in questo modo si risparmia memoria), altrimenti ponete 1; il parametro I contiene il numero della finestra su cui lavorate; inizialmente deve valere 1, successivamente se dovete correggere qualche finestra, ponete in tale parametro il numero della finestra da modificare. Infine N esprime il numero di finestre che desiderate comporre. Dopo aver definito la stringa A\$, occorre memorizzare i dati e predisporre lo spazio per inserire le varie finestre; per

ottenere ciò, fate Clear Usr 64052 (per il 16 Kbyte Clear Usr 32052). A questo punto non vi resta che comporre dei disegni sul video e poi immagazzinarli con Randomize Usr 64166. Vi ricordiamo che non verrà memorizzato tutto lo schermo, ma solo quella porzione rettangolare contenuta nella finestra inizialmente definita (se avete il 16 Kbyte fate Randomize Usr 32166). Per immagazzinare l'intero schermo, basta assegnare a X e a Y il valore 0 e a

LX,32 e a LY,22 (se desiderate salvare le ultime due linee dello schermo basso, usate 24).

A questo punto non vi resta che inserire la sequenza con cui desiderate rivedere i fotogrammi preparati; tale sequenza va posta nella stringa A\$ nel modo seguente:

Let A\$ Chr\$A + Chr\$B + Chr\$C ...

A,B,C esprimono il numero del fotogramma che si vuol rivedere.

Seguito figura 4.

```

641008 213 ..... push de
641009 17 21 0 ..... ld de,21
641010 205 ..... add hl,de
641011 209 ..... pop de
641014 207 176 ..... ldir
641016 205 ..... ex de,hl
641017 04 255 ..... ld (hl),255
641019 201 ..... ret
641110 40 75 90 ..... ld hl,(20627)
641113 00 90 90 ..... ld a,(20642)
641116 100 ..... cp h
641117 210 0 250 ..... jp c,64006
641120 120 0 ..... ld a,(hl)
641121 05 ..... inc hl
641122 204 ..... cp 00
641124 00 243 ..... jr nz,64113
641126 120 0 ..... ld a,(hl)
641127 204 3 ..... cp 0
641129 00 245 ..... jr nz,64120
641131 05 ..... inc hl
641132 120 0 ..... ld a,(hl)
641133 107 ..... and a
641134 00 240 ..... jr nz,64120
641136 05 ..... inc hl
641137 195 221 251 ..... jp 64477
641140 17 0 250 ..... ld de,64000
641143 207 176 ..... ldir
641145 03 244 251 ..... ld hl,64500
641148 17 112 0 ..... ld de,112
641151 00 0 250 ..... ld a,(64000)
641154 107 ..... and a
641155 202 0 250 ..... jp z,64006
641158 71 ..... ld b,a
641159 00 1 250 ..... ld a,(64001)
641162 204 200 ..... cp 20
641164 210 0 250 ..... jp nc,64006
641167 00 2 250 ..... ld a,(64002)
641170 204 20 ..... cp 20
641172 210 0 250 ..... jp nc,64006
641175 205 ..... add hl,de
641175 10 253 ..... djnz 64175
641178 04 4 250 ..... ld (64004),hl
641181 17 54 0 ..... ld de,54
641184 205 ..... add hl,de
641185 220 ..... push hl
641188 221 205 ..... pop ix
641190 120 0 ..... ld a,(hl)
641193 204 205 ..... cp 205
641191 202 227 250 ..... jp z,64227

```

```

641194 04 ..... ld e,(hl)
641195 00 ..... inc hl
641198 00 ..... ld d,(hl)
641197 00 ..... inc hl
641198 00 ..... inc hl
641199 00 ..... inc hl
642000 00 ..... ex de,hl
642001 00 20 ..... ld a,205
642003 00 100 0 ..... ld (64410),a
642006 00 100 0 ..... ld (64420),a
642008 00 100 0 ..... ld (64440),a
642010 00 100 0 ..... ld (64445),a
642015 205 176 2051 ..... call 64430
642018 201 100 0 ..... ld h,(ix+3)
642021 201 120 0 ..... ld l,(ix+2)
642024 205 100 2051 ..... call 64414
642027 40 1 200 ..... ld hl,(64001)
642031 00 0 ..... ld e,l
642032 00 0 ..... ld d,0
642034 00 0 4 ..... ld hl,10004
642037 00 0 ..... add hl,de
642038 00 0 ..... ld b,02
642040 00 0 ..... ld e,a
642041 205 ..... add hl,de
642042 10 203 ..... djnz 64241
642044 200 ..... push hl
642045 17 0 24 ..... ld de,6144
642048 205 ..... add hl,de
642049 201 117 0 ..... ld (ix+2),l
642050 201 116 0 ..... ld (ix+3),h
642053 205 ..... pop hl
642056 17 0 7 ..... ld de,1702
642059 204 0 ..... cp 0
642061 200 0 ..... jr c,64260
642063 205 ..... add hl,de
642064 10 ..... cp 10
642066 1 ..... jr c,64260
642068 205 ..... add hl,de
642070 201 117 0 ..... ld (ix+0),l
642072 201 116 1 ..... ld (ix+1),h
642075 00 0 ..... ld a,0
642077 00 10 0 ..... ld (64410),a
642080 00 100 0 ..... ld (64420),a
642083 00 100 0 ..... ld (64440),a
642086 00 100 0 ..... ld (64445),a
642089 201 200 ..... push ix
642091 200 ..... pop de
642092 10 0 ..... inc de
642093 10 ..... inc de
642094 10 ..... inc de

```

## ZX SPECTRUM

Seguito figura 4.

```

044000 100 .....inc de
044001 176 051 .....call 644032
044002 102 0 .....ld h, (ix+3)
044003 110 0 .....ld l, (ix+2)
044004 158 051 .....call 64414
044005 110 0 .....ld l, (ix+0)
044011 102 1 .....ld h, (ix+1)
044014 237 91 4 250 .....
.....ld de, (64004)
044020 0 0 .....ld a, 0
044021 0 0 .....ld (64003), a
044022 158 0 .....ld a, 158
044023 107 051 .....ld (64003), a
044024 110 051 .....call 64374
044025 110 0 .....push de
044026 17 0 0 .....ld de, 20
044027 0 0 .....nop
044028 0 0 .....ld a, 220
044029 103 0 .....cp l
044030 48 0 .....jr nc, 64434
044041 17 0 7 .....ld de, 1801
044044 0 0 .....add hl, de
044045 0 0 .....pop de
044046 110 051 .....call 64374
044047 102 1 03 .....ld h, (ix+3)
044048 110 1 03 .....ld l, (ix+2)
044049 2 035 .....ld a, 235
044050 163 051 .....ld (64410), a
044051 165 051 .....ld (64422), a
044052 158 051 .....call 64414
044053 1 0 0 .....ld bc, 0
044054 58 3 250 .....ld a, (64003)
044070 70 .....ld c, a
044071 201 .....ret
044074 14 3 .....ld c, 3
044075 107 .....push bc
044076 220 .....push hl
044077 5 0 .....ld b, 0
044078 25 0 .....ld a, (de)
044081 182 .....or (hl)
044082 79 .....ld c, a
044083 110 .....ld (hl), a
044084 25 .....ld a, (de)
044085 145 .....sub c
044086 48 7 .....jr nz, 64395
044087 0 0 .....push hl
044088 0 0 0 .....ld hl, 64003
044089 0 0 0 .....set b, (hl)
044090 0 0 .....pop hl
044091 0 0 .....inc b
044092 100 .....inc de
044093 16 237 .....djnz 64380
044094 0 0 .....pop hl
044095 50 0 137 251 .....ld a, (64393)
044096 198 0 0 .....add a, 0
044097 107 051 .....ld (64393), a
044098 0 0 .....inc hl
044099 100 0 .....pop bc
044100 103 0 .....dec c
044111 0 0 .....ret z
044112 0 0 10 .....jr 64376
044114 0 0 .....ld a, 2
044115 1 0 0 .....ld bc, 3
044116 0 0 .....ex de, hl
04420 0 0 7 176 .....ldir

```

```

044200 0 0 .....ex de, hl
044201 1 0 0 .....ld bc, 20
044202 0 0 74 .....adc hl, bc
044203 0 0 1 .....dec a
044204 0 0 41 .....jr nz, 64416
044205 0 0 .....ret
044206 0 0 .....ld b, 2
044207 0 0 .....push bc
044208 0 0 .....push hl
044209 0 0 .....ld a, 0
044210 0 0 .....push hl
044211 0 0 0 .....ld bc, 0
044212 0 0 0 .....nop
044213 0 0 176 .....ldir
044214 0 0 .....nop
044215 0 0 .....pop hl
044216 0 0 .....inc b
044217 0 0 .....dec a
044218 0 0 240 .....jr nz, 64438
044219 0 0 .....pop hl
044220 0 0 .....ld a, l
044221 0 0 4 004 .....cp 004
044222 0 0 0 .....jr c, 64466
044223 1 0 0 7 .....ld bc, 1824
044224 0 0 74 .....adc hl, bc
044225 0 0 .....ld b, 0
044226 24 5 .....jr 64472
044227 14 32 .....ld c, 32
044228 0 0 .....nop
044229 16 7 .....and a
044230 26 7 74 .....adc hl, bc
044231 103 .....pop bc
044232 16 215 .....djnz 64434
044233 201 .....ret
044234 0 0 .....nop
044235 04 .....ld d, h
044236 93 .....ld e, l
044237 0 0 250 .....ld a, (64000)
044238 100 0 .....cp (hl)
044239 16 .....jr nz, 64501
044240 0 0 .....inc hl
044241 1 250 .....ld a, (64001)
044242 190 0 .....cp (hl)
044243 0 0 .....jr nz, 64501
044244 0 0 .....inc hl
044245 0 0 250 .....ld a, (64002)
044246 190 0 .....cp (hl)
044247 0 0 7 .....jr nz, 64501
044248 0 0 .....jr 64503
044249 0 0 5 .....ex de, hl
044250 1 0 0 .....ld bc, 3
044251 19 0 140 250 .....jp 64140
044252 0 0 .....ex de, hl
044253 0 0 .....push hl
044254 4 250 .....ld hl, (64004)
044255 17 54 0 .....ld de, 54
044256 0 0 .....add hl, de
044257 105 .....ld a, (hl)
044258 4 250 .....cp 255
044259 0 0 3 .....jr nz, 64525
044260 0 0 .....pop hl
044261 4 0 3 .....jr 64502
044262 0 0 .....pop de
044263 0 0 .....push hl
044264 0 0 5 .....pop ix
044265 19 5 52 251 .....jp 64308

```

**Listato 5 - Esempio di funzionamento del programma per la gestione dei fotogrammi.**

```

1 REM fotogrammi
5 IF PEEK 64030=92 THEN GO TO
20
10 PRINT "devi caricare la rou
tine in      linguaggio macchina
per la      gestione delle immag
ini" "prepara il nastro e premi
un tasto"
11 PAUSE 0: CLEAR 63999: LOAD
""CODE
15 REM DEFINIZIONE STRINGA A$
20 LET A#=CHR$ 3+CHR$ 4+CHR$ 2
9+CHR$ 15+CHR$ 1+CHR$ 1+CHR$ 5
25 REM DEFINIZIONE MEMORIA
30 CLEAR USR 64052
35 REM DEFINIZIONE IMMAGINE
40 FOR I=1 TO 5
50 CLS
60 INK 0: CIRCLE 80,80,I*10: I
NK 4: CIRCLE I*25+80,80,50
65 INK 0: PRINT AT 9,15;I;I;I;
70 I: INK 2: PAPER I: PRINT AT 8,
71;"KIKIKIKIKI"
80 PAPER 7
90 REM MEMORIZZAZIONE IMMAGINE
00 RANDOMIZE USR 64166
95 NEXT I
05 REM DEFINIZIONE SEQUENZA

```

```

100 LET A#=CHR$ 1+CHR$ 2+CHR$ 3
+CHR$ 4+CHR$ 5+CHR$ 4+CHR$ 3+CHR
# 2
110 POKE 64042,LEN A$
115 REM MEMORIZZAZIONE SEQUENZA
120 RANDOMIZE USR 64328
125 REM PREPARAZIONE IMMAGINE
130 RANDOMIZE USR 64368
140 PAUSE 5
150 GO TO 130
160 REM DEFINIZIONE FOTOGRAMMA S
170 LET A#=CHR$ 3+CHR$ 4+CHR$ 2
9+CHR$ 15+CHR$ 1+CHR$ 3+CHR$ 5
180 RANDOMIZE USR 64052
190 REM NUOVA IMMAGINE
200 CIRCLE 100,80,50: PRINT AT
10,9;"NUOVA IMMAGINE"
210 REM MEMORIZZAZIONE
220 RANDOMIZE USR 64166
225 REM SEQUENZA ALLOCAZIONE
230 CLS: PRINT "VISUALIZZAZINE
NUOVA IMMAGINE"
240 PAUSE 30
250 LET A#=CHR$ 3
255 POKE 64042,LEN A$
260 RANDOMIZE USR 64328
270 RANDOMIZE USR 64368
280 INPUT "VUOI RIVEDERE LA SEQ
UENZA";S$
290 IF S$="s" OR S$="S" THEN GO
TO 100

```

CARATTERE	1° COLONNA	2° COLONNA	3° COLONNA	
	16384		16415	1° RIGA
	16640		16671	
	16896			
	17152			
	17408			
	17664			
	17920			
	18176			2° RIGA
	16416			
	16672			
	16928			

**Tabella 3 - Struttura delle prime due righe del display file; ogni carattere occupa otto byte distanziati 256 byte l'uno dall'altro.**

Poi fate Poke 64042 (per il 16 Kbyte 32042). Per trasferire i dati della stringa nell'area di lavoro, date Randomize Usr 64328. Infine, per rivedere le immagini da voi preparate, usate Randomize Usr 64368 (per il 16 Kbyte le chiamate sono rispettivamente a 32328 e a 32368).

Se desiderate modificare il fotogramma numero n dovete inserire nella stringa iniziale al posto di I il numero n (come già accennato) e poi dare Randomize Usr 64052. Per visualizzare i fotogrammi in una posizione diversa da quella iniziale, occorre modificare i parametri X e Y della stringa A\$ nel modo desiderato; successivamente inserite un Randomize Usr 64052 per introdurre nell'area di lavoro i dati.

Il programma in linguaggio macchina è composto da 4 routine.

La prima (Call 64052), provvede a individuare la stringa A\$ secondo la procedura precedentemente illustrata; poi ne trasferisce il contenuto a partire dalla locazione 64000; fatto questo, calcola la quantità di memoria per ogni fotogramma (il colore occupa un numero di byte pari a LX★ LY, mentre l'immagine 8★







**DA OGGI È MUSICA PER TUTTI**

# A TUTTA MUSICA CON

# SOUND



Musica dal computer, musica vera, completa, definitiva: da ascoltare con gli amici. Ed è musica tua!

Con l'unità periferica SOUND BUGGY e la tastierina musicale SIEL da appoggiare sopra la tastiera alfanumerica, il tuo Commodore 64 si trasforma in una autentica band e tu diventi subito concertista, compositore e arrangiatore.

### *Musicista in pochi giorni*

Se sei già esperto di musica, SOUND BUGGY ti porterà alla perfezione. Se sei principiante, ti troverai in pochi giorni di fronte a un miracolo: comporrà musica tua e potrai ascoltarla in una perfetta registrazione elettronica collegabile a ogni impianto stereo, videotelevisivo, monitor C64.

### *Un pacchetto di programmi che concentra anni di studio*

Grazie al consistente, straordinario software di Sound Buggy, potrai eseguire o comporre su 24 ritmi (12 preregistrati), disponendo di ben 28 timbri strumentali (14 preregistrati), potendo correggere, migliorare, variare e disporre quindi per la registrazione definitiva.

Tramite interfaccia MIDI, SOUND BUGGY comunica anche con gli altri strumenti musicali elettronici (expander, sintetizzatori, sequencer ecc.). SOUND BUGGY è un prodigio dell'elettronica al servizio della creatività.

Spia luminosa di corretta alimentazione

Volume per la batteria elettronica

Volume per la parte orchestrale

Volume generale



### **MAXIPRESTAZIONE IN MINISPAZIO**

*Il tuo laboratorio musicale, completo e perfetto, è tutto qui: il Commodore 64, il SOUND BUGGY, la minitastiera SIEL, il software.*

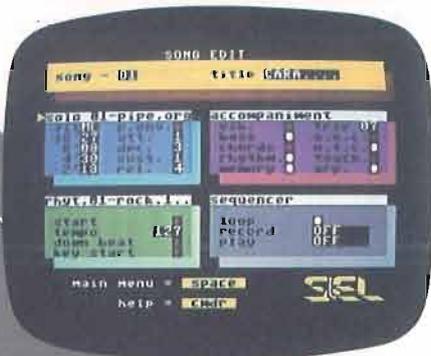
## SIEL®

LA QUALITÀ DELLA MUSICA

# BUGGY

## Main Menu

Massima semplicità d'impiego del software grazie alla standardizzazione dei comandi. Il Main Menu presenta il quadro complessivo delle funzioni musicali utilizzabili premendo i comandi indicati.



## Rhythm Menu

Visualizza tutti i 24 ritmi e mostra come SOUND BUGGY li esegue alla batteria elettronica. La batteria è programmabile per: Bass Drum (grancassa), Cymbal Short (piatto aperto), Cymbal Long (piatto chiuso), Snare Drum (tamburo rullante) e Rim Shot (cerchio rullante).

## Solo Menu

Elenca tutti i 28 timbri e visualizza i parametri con cui il sint SOUND BUGGY li sviluppa.

## Song Edit

È il cuore del SOUND BUGGY e permette di dirigere l'"orchestra". Infatti consente di intervenire in tempo reale, modificando i parametri relativi a: Solo, Rhythm, Accompaniment, Sequencer.

## CEDOLA PRIVILEGIATA DI ACQUISTO

### SOUND BUGGY

Indicare con una X il  che interessa



**SIEL**

Si desidero acquistare SOUND BUGGY, la vostra unità periferica per C64. Speditemela contrassegno completa del pacchetto software sia su disco che su cassetta e libretto istruzioni al prezzo speciale di Lire 185.000 (incluse L. 27.650 IVA e L. 3.750 di

spese postali). È inteso che il mio SOUND BUGGY sarà coperto da Garanzia per 1 anno.

Sono interessato alla unità periferica per C64 SOUND BUGGY e vi prego di valermi fornire informazioni più dettagliate.

## 12 MESI DI GARANZIA

NOME \_\_\_\_\_ COGNOME \_\_\_\_\_

INDIRIZZO \_\_\_\_\_

CAP \_\_\_\_\_ LOCALITÀ \_\_\_\_\_

DATA \_\_\_\_\_ FIRMA \_\_\_\_\_

(per i minori occorre quella del genitore)

DATI FACOLTATIVI

ETÀ \_\_\_\_\_ PROFESSIONE \_\_\_\_\_

TITOLO DI STUDIO \_\_\_\_\_

Compila e spedisce l'unità Cedola Privilegiata. È l'unico mezzo per ricevere SOUND BUGGY, completo di tutti i programmi, su disco e cassetta, a un prezzo speciale, direttamente a domicilio e con 12 mesi di Garanzia.

**AFFRETTATI!!**

Il prezzo è bloccato solo fino al 31-12-'85.



# SOUND BUGGY



## TUTTA LA MUSICA A COMANDO!

A tua disposizione hai gli strumenti, i ritmi, i tempi, gli accordi, gli arpeggi, gli effetti mixati, come una grande, insperata tavolozza di suoni. Ma tutto questo resterebbe muto senza l'intervento della tua creatività: sotto la guida del tuo estro musicale, seguendo le facilissime istruzioni dei programmi, prenderà vita sotto le tue mani dapprima la musica che vuoi produrre, poi la tua musica.

### Sequencer

È una funzione polifonica che ti permette di comporre e registrare in tempo reale fino a 9 song in linea, archivarle su disco e creare così una discoteca da richiamare a tuo piacere.

### Rhythm

Un'intera batteria elettronica al tuo comando con tre parametri modificabili: Start, Tempo, Key-Start. C'è anche il Down Beat che, come un metronomo, visualizza le battute musicali.



### Accordi

Grazie a due sofisticate tecniche di sviluppo, basta un dito per esplorare tutte le possibilità armoniche di base e creare accordi automatici.



### Orchestrazioni

Creato l'accordo il SOUND BUGGY si fa in quattro: basso, batteria, arrangiamento con arpeggio e accompagnamento ritmato trasformandoti in direttore d'orchestra.



### Ha anche una grande tastiera.

Con la piccola tastiera sovrapposta a quella del computer, SOUND BUGGY ti dà tutto ciò che puoi chiedere al tuo gusto musicale. Ma con la grande tastiera SIEL CMK 49 a passo professionale, l'elettronica ti offre una ricchezza musicale assoluta, con la possibilità di impostare la tua musica a due mani su un'unica tastiera, come nel classico pianoforte.

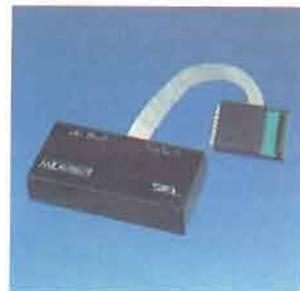


### Con CMK 49 puoi diventare un vero tastierista.

Qualsiasi altra tastiera collegabile al tuo Commodore 64 diventa "magica" con SOUND BUGGY: perché il suo software è universale.

# MIDI

(Musical Instruments Digital Interface)



Grazie a questo linguaggio comune tra sintetizzatori professionali, SOUND BUGGY è in grado di parlare con 4 tastiere professionali o expander creando delle orchestrazioni di eccezionale qualità timbrica. L'importanza di questo fattore è ben conosciuta dal musicista esperto; per chi lo vuol diventare SOUND BUGGY sarà una meravigliosa scoperta.

## AVVISO IMPORTANTE

Il SOUND BUGGY è acquistabile esclusivamente tramite il coupon privilegiato di questa offerta. Tutti gli altri strumenti elettronici SIEL sono in vendita presso i migliori Negozi di strumenti musicali. Se siete interessati rivolgetevi a loro. Oppure, se desiderate altre informazioni, scrivete a "Filodiretto SIEL" Casella Postale 10823 20124 Milano MI.

# SIEL

SIEL S.p.a.  
Stampa propagandistica  
Semestrale n. I. II semestre  
1985. Sped. Abb. Postale V  
Gruppo. Vendite per  
Corrispondenza. Proposta,  
accettazione, prezzo e invio del  
prodotto si realizzeranno  
attraverso i Servizi Postali.  
C.C.P. n. 12829636.

# COUPON GARANZIA 12 MESI

Compila e spedisce  
in busta chiusa a:

"Filodiretto SIEL"

# SIEL

SOCIETÀ INDUSTRIE ELETTRONICHE s.p.a.  
CASELLA POSTALE 10823  
20124 Milano MI



Listato 6 - Esempio di funzionamento del programma in linguaggio macchina per la gestione degli sprite.

```

1 REM sprite
2 IF PEEK 64030=35 THEN GO TO
20
3 PRINT "devi caricare il pro
gramma in linguaggio macchina p
er la gestione degli sprit
e".
4 PRINT "prepara il nastro e premi u
n
tasto"
4 PAUSE 0
10 CLEAR 63999
11 LOAD ""CODE
20 POKE 64058,1
30 DIM a$(18,3)
35 PRINT "inserisci delle lett
ere a caso tre per volta"
35 REM DEFINIZIONE CUALE DI
CASCHE SPRITE
40 FOR i=1 TO 18
50 INPUT a$(i): PRINT i,a$(i)
60 NEXT i
70 LET a$(17)=CHR$ 56+CHR$ 47+
CHR$ 56: LET a$(18)=a$(17)
70 REM MEMORIZZAZIONE
80 RANDOMIZE USR 64010
85 PRINT "PREMI UN TASTO": PAU
SE 0
90 CLS : PRINT ""LA CIFRA IN
ALTO A SINISTRA ESPRIME LA
COLLISIONE.PER MUOVERTI USA IL C
URSORE"
100 LET b#=CHR$ 1+CHR$ 10+CHR$
10
110 LET x=10: LET y=10
120 LET x=x+(INKEY#="8")-(INKEY
#="5")
121 LET y=y+(INKEY#="7")-(INKEY
#="6")
130 LET b$(2)=CHR$ x: LET b$(3)
=CHR$ y
135 REM MEMORIZZAZIONE
140 LET k=USR 64110
145 PRINT AT 0,0;k;" "
150 PAUSE 5
160 GO TO 120

```

Listato 7 - Listato del programma per la realizzazione degli sprite.

```

9000 BORDER 0: PAPER 0: INK 7: C
LS : IF PEEK 64092+PEEK 64346<>4
14 OR PEEK 23730+256*PEEK 23731>
64000 THEN GO TO 9750
9001 PRINT "*****"
*****
9002 PRINT "*" SPRITE EDIT
OR
*****
9003 PRINT "*****"
*****
9004 PRINT ""Con questo progr
amma puoi definire i 8 spr
ite che hai a disposizione.

```

Ogni sprite e' co  
mposto da 24\*16 pixel"  
9005 PRINT "" per elaborare lo  
sprite"" per caricare o visu  
alizzare gli sprite"" per mod  
ificare uno sprite in memoria

```

9005 INPUT LINE s#
9007 IF s#="2" THEN GO TO 9800
9010 INPUT "INSERISCI IL NUMERO
DELLO SPRITE ";num
9011 IF num<=0 OR num>=8 THEN GO
TO 9010
9015 POKE 64058,num: CLS
9020 FOR I=3 TO 28: PRINT PAPER
6: AT 0,I;" ": PRINT AT 17,I: PAP
ER 6;" ": NEXT I
9030 FOR I=0 TO 17: PRINT AT I,3
: PAPER 6;" ": PRINT AT I,28: PA
PER 6;" ": NEXT I
9040 PRINT AT 19,0;"Per spostart
i usa i tasti del cursore
9=stampa
0=cancella"

```

```

9050 PRINT #1;"Al termine premi
p"
9060 LET k=1: FOR i=1 TO 16: FOR
j=4 TO 27: PRINT AT i,j: PAPER
7: BRIGHT k;" ": LET k=1-k: NEXT
j: LET k=1-k: NEXT i
9070 IF s#="3" THEN GO SUB 9910
9080 LET x=4: LET y=1: LET n=120
: GO TO 9095
9090 LET k=PEEK 23560
9091 POKE 23560,0
9092 IF k>56 OR k<53 THEN GO TO
9120
9095 POKE 22528+x+y*32,n
9096 BEEP .02,1
9100 LET x=x+(k=56 AND x<27)-(k=
53 AND x>4)
9105 LET y=y+(k=54 AND y<16)-(k=
55 AND y>1)
9107 LET n=PEEK (22528+x+y*32)
9110 POKE 22528+x+y*32,0
9111 GO TO 9090
9120 IF k=57 THEN LET n=0: BEEP
.05,10
9130 IF k=48 THEN LET n=56+64*(I
NT ((x+y)/2)-(x+y)/2<>0): BEEP .
05,5
9140 IF k=80 OR k=112 THEN GO TO
9200
9160 GO TO 9090
9200 LET c#=""

```

```

9201 INPUT 1: PRINT AT 19,0;c#;c
#;c#
9203 PRINT AT 20,4;"ATTENDI UN A
TTIMO"
9205 DIM a$(18,3)
9220 FOR i=1 TO 16: LET x=4: FOR
k=1 TO 3: LET n=0: FOR j=7 TO 0
STEP -1
9230 IF PEEK (22528+x+i*32)=0 TH
EN LET n=n+2+j
9240 LET x=x+1: NEXT j: LET a$(i

```

## ZX SPECTRUM

te fate invece Poke 31290, num). Per memorizzare la matrice fate Randomize Usr 64010 (per il 16 Kbyte Usr 31242). A questo punto, per stampare e spostare il vostro sprite, inserite in B\$ il numero dello sprite e la sua posizione e fate Randomize Usr 64110 (per il 16 Kbyte Usr 31342). Variando la posizione di stampa farete muovere il vostro sprite. Il linguaggio macchina provvede a cancellare l'immagine nella vecchia posizione e a stamparla nella nuova; inoltre, è attivato l'effetto Over, cioè se lo sprite va sopra un'altra immagine, si sovrappone ad essa senza cancellarla.

Vi è anche la possibilità di sapere su quale dei sei caratteri che compongono lo sprite è avvenuta la sovrapposizione o collisione, basta fare al posto di Randomize Usr 64110, Let k= Usr 64110; in k vi sarà un numero compreso tra 0 e 63

che esprime una tale evenienza. Per interpretare tale valore bisogna trasformarlo in binario e studiare le prime sei cifre: se sono tutte 0 non vi è alcuna collisione, se sono tutte a uno (cioè  $k=63$ ), significa che vi è sovrapposizione totale sui sei caratteri che compongono lo sprite; se la prima cifra vale uno, significa che vi è collisione sul primo carattere; se la seconda cifra è a uno, vi è collisione sul secondo carattere ecc..

Ogni sprite occupa 112 byte; 54 per il disegno e il colore, altri 54 per salvare l'immagine video su cui lo si va a sovrapporre e, infine, 4 byte nei quali è memorizzata la posizione nella memoria video, dello sprite. Essi sono memorizzati a partire dalla locazione 64612. Analizziamo ora il linguaggio macchina. La prima routine (Call 64010), cerca la matrice stringa A\$(18,3) che è memo-

rizzata nel seguente modo: 193 (codice di A\$ matrice); 59, (occupazione di memoria); 2 (numero di dimensioni); 18,0 (numero di righe); 3,0 (numero di colonne) .... testo stringa.

Poi, sapendo il numero dello sprite, trasferisce il contenuto della stringa a partire dalla locazione 64500 + 112 per numero sprite.

La seconda routine (Call 64110) cerca la stringa B\$ di tre caratteri, memorizzata come 66 3 0; controlla poi se lo sprite deve essere stampato nella stessa posizione o meno e in tal caso esegue solo la stampa dello sprite. In caso contrario, cancella lo sprite e ripristina l'immagine precedente, poi calcola le nuove coordinate nella memoria di schermo secondo le formule viste in precedenza; a questo punto salva la porzione di schermo su cui stamperà lo sprite; infine, av-

## Seguito Ilistato 7.

```

K)=CHR$ n: NEXT k: NEXT i
9250 LET a$(17)=CHR$ 7+CHR$ 7+CHR$ 7
R# 7: LET a$(18)=a$(17)
9260 PRINT AT 20,0;c$
9270 RANDOMIZE USR 64010
9280 LET b#=CHR$ num+CHR$ 12+CHR$ 20
9290 RANDOMIZE USR 64110
9300 INPUT "vuoi modificarlo (s/n)";s$
9310 IF s#="s" OR s#="S" THEN GO TO 9030
9400 CLS: PRINT "Ora devi colorarlo. Lo sprite è diviso in 6 settori per ognuno di essi devi definire lo sfondo, l'inchiostro e la luminosità"
9410 LET b#=CHR$ num+CHR$ 13+CHR$ 12: RANDOMIZE USR 64110
9420 PRINT AT 11,13;"123";AT 14,13;"456"
9435 FOR i=1 TO 6
9437 PRINT AT 21,3;"SETTORE ";i
9440 INPUT "sfondo (0-7)";s
9441 IF s<0 OR s>7 THEN GO TO 9440
9445 INPUT "inchiostro (0-7)";i
9446 IF in<0 OR in>7 THEN GO TO 9445
9448 INPUT "luminosità (0 o 1)";l
9449 IF l<>0 AND l<>1 THEN GO TO 9448
9450 LET s=in+8*s+64+l
9455 IF i<4 THEN POKE 22924+i,s: LET a$(17,i)=CHR$ s

```

```

9460 IF i>3 THEN POKE 22953+i,s: LET a$(18,i-3)=CHR$ s
9470 INPUT "vuoi cambiare i colori (s/n)";s$
9480 IF s#="s" OR s#="S" THEN GO TO 9440
9490 NEXT i
9500 CLS
9510 RANDOMIZE USR 64010: LET b#=CHR$ num+CHR$ 13+CHR$ 3: RANDOMIZE USR 64110
9519 PRINT AT 5,0:"lo sprite numero ";num;" è stato completato"
9520 PRINT "MENU"" sa lvataggio"" elaborazione nuovo sprite"" fine"
9530 INPUT LINE s$
9540 IF s#="3" THEN STOP
9550 IF s#="2" THEN GO TO 9000
9570 IF s#="1" THEN GO TO 9600
9580 GO TO 9530
9590 GO SUB 9900: CLS: PRINT "***SPRITE EDITOR***"
9610 PRINT "MENU"
9620 PRINT " salvataggio dei 6 sprite e della routine per gestirli"" salvataggio solo dei 6 sprite"" salvataggio di uno sprite"
9630 INPUT LINE s$
9640 IF s#="1" THEN LET k=64000: LET l=1300: GO TO 9700
9650 IF s#="2" THEN LET k=64600: LET l=700: GO TO 9700
9660 IF s#<>"3" THEN GO TO 9630
9670 INPUT "numero dello sprite da salvare ";n
9680 IF n<1 OR n>6 THEN GO TO 9670: LET l=112

```

viene la stampa dello sprite e l'analisi della collisione. La subroutine di stampa nella memoria degli attributi avviene dalla locazione 64414, mentre quella per le immagini avviene dalla locazione 64432; infine, la stampa con Over per gli sprite avviene a partire dalla locazione 64374. Se desiderate togliere l'effetto Over, dovete modificare le seguenti locazioni:

```
64381 porre 119 (LD (HL),A)
64382 " 182 (OR (HL) )
64383 " 79 (LD C,A)
```

Nel caso desideriate che lo sprite non ripristini l'immagine su cui precedentemente si era sovrapposto, perché dovette ricominciare il gioco, o perché volete che lo sprite lasci una traccia mentre si sposta, dovete fare la seguente Poke: Poke 64554+112★ numero sprite, 255 (se avete il 16 Kbyte fate Poke

31786+112★ numero sprite, 255).

Lo sprite può essere posizionato su tutto lo schermo eccetto che sulla trentaduesima colonna; può anche essere stampato sulle ultime due righe dello schermo basso. Nel caso si cerchi di posizionarlo fuori schermo, si otterrà un messaggio di errore. Per capire meglio quanto detto, provate ad inserire il programma BASIC del listato 6 qui, dopo aver composto casualmente lo sprite, potete spostarlo su tutto lo schermo; in alto a sinistra, viene visualizzato lo stato di collisione.

Per realizzare i vostri sprite facilmente, vi consigliamo di inserire il programma del listato 7; con tale programma potete disegnare, colorare, correggere, visualizzare, salvare e caricare da cassetta i vostri sprite. Il programma è predisposto per funzionare su un 48 Kbyte, chi invece possedesse un 16 Kbyte deve

apportare ad esso delle modifiche: tutti gli argomenti delle istruzioni Peek, Poke, Clear, Load"Code, devono essere decrementati di 32.768, solo nel caso in cui risultino superiori a 60.000.

Tutte le cifre superiori a 60.000 devono quindi subire un tale decremento. Ad esempio la linea 9000 diventa:

```
9000 BORDER 0: PAPER 0: INK 7: CLS :
IF PEEK 31324+PEEK 31578<> 414
OR PEEK 23730 +256xPEEK 23731>
31232 THEN GO SUB 9750
```

#### Seguito listato 7.

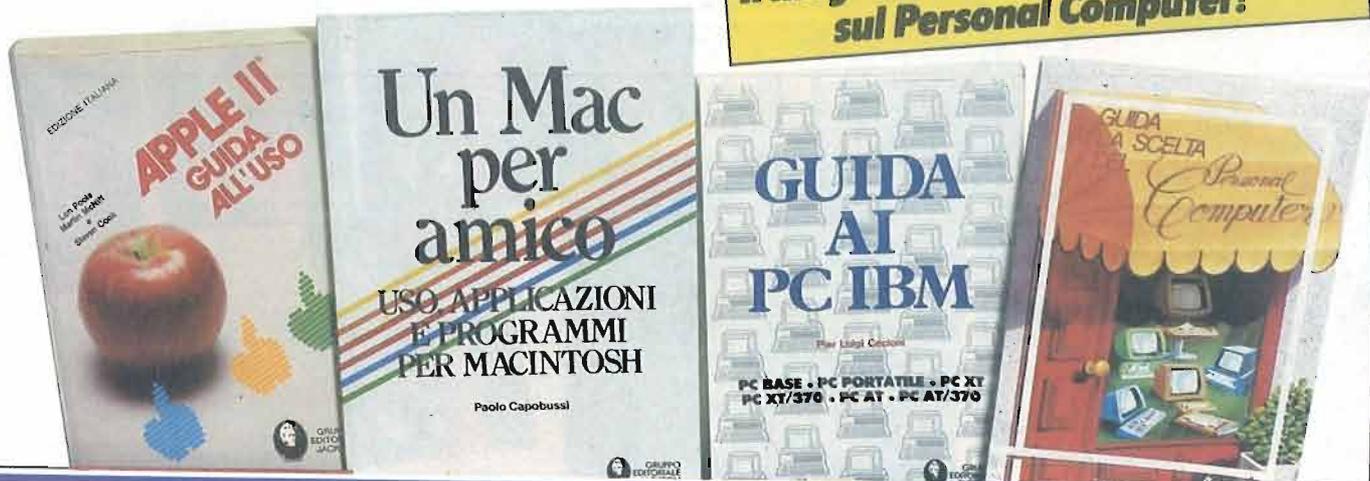
```
9690 LET k=64600+l*(n-1)
9700 INPUT "nome file ";s#
9710 SAVE s#CODE k,l
9720 CLS : GO TO 9520
9750 PRINT "ERRORE LA RAMTOP NON
E' POSIZIONATA ADEGUATA
MENTE O LA ROUTINE PER LA
GESTIONE DEGLI SPRITE NON E' I
N MEMORIA" "BISOGNA RIPETERE IL
CARICAMENTO" "POSIZIONA IL NA
STRO E PREMI UN TASTO"
9755 PAUSE 0
9760 CLEAR 63999: LOAD "CODE 64
000
9795 GO TO 9001
9800 CLS
9805 PRINT " menu" " visualizza
sprite in memoria" " carica
set sprite" " elaborazione
sprite" " fine"
9810 INPUT LINE s#
9815 IF s#="4" THEN STOP
9820 IF s#(">"1" AND s#(">"2" AND
s#(">"3" THEN GO TO 9810
9830 GO TO 9830+20*VAL s#
9850 GO SUB 9900: CLS : LET y=3:
FOR i=1 TO 6
9855 LET b#=CHR$ i+CHR$ (i+4)+CH
R$ y: RANDOMIZE USR 64110: NEXT
i
9860 PRINT #1:"premi un tasto":
PAUSE 0: INPUT i: GO TO 9800
9870 CLS : INPUT "nome file ";s#
9880 LOAD s#CODE 64600
9885 GO TO 9805
9890 GO TO 9000
9900 FOR i=64556 TO 65226 STEP 1
```

```
12: POKE i,255: NEXT i: RETURN
9910 INPUT "numero sprite da mod
ificare";s
9920 IF s<1 OR s>6 THEN GO TO 99
10
9930 LET n=64612+112*(s-1)
9940 FOR i=0 TO 47: LET k=PEEK (
i+n): IF k=0 THEN GO TO 9950
9945 FOR j=7 TO 0 STEP -1: IF k-
2↑j>=0 THEN LET k=k-2↑j: GO TO 9
955
9946 IF k=0 THEN GO TO 9960
9950 NEXT j
9960 NEXT i
9961 PRINT #1;"al termine premi
n"
9962 RETURN
9965 LET y=1: LET s=i: IF i>23 T
HEN LET y=9: LET s=i-24
9970 LET x=11-j+8*(s>7)+8*(s>15)
: LET y=y+s-INT (s/8)*8: PRINT P
APER 0: INK 0:AT y,x:" "
9975 GO TO 9946
```

# per hobby e per professione

\* Anche i non abbonati possono ordinare libri, naturalmente senza sconto alcuno.

**Il meglio di quel che c'è da leggere sul Personal Computer!**



250 volumi per approfondire, dalle più diverse angolazioni, le tematiche legate al mondo del computer e della microelettronica.

Per ordinare i volumi preferiti utilizzare l'apposito modulo di conto corrente postale blu o il coupon.



**GRUPPO EDITORIALE JACKSON**

**S.R. TROST**  
**Programmi utili per IBM PC**  
65 programmi collaudati e pronti all'uso. 192 pagine  
Cod. 564 D • Lire 15.000 / Abbonati Lire 12.000

**N. BRÉAUD - POULIQUEN**  
**Apple Memo**  
Regole e "malizie" per l'utilizzo dell'Apple. 146 pagine  
Cod. 340 H • Lire 15.000 / Abbonati Lire 12.000

**M. BROWNE**  
**Unità a dischi per Personal Computer**  
Tecniche di gestione dei file su disco. 156 pagine  
Cod. 300 P • Lire 15.000 / Abbonati Lire 12.000

**J.W. COFFRON**  
**Apple tutto fare**  
Il controllo di processo realizzato con l'Apple su applicazioni di tipo "domestico". 219 pagine  
Cod. 334 D • Lire 18.000 / Abbonati Lire 14.400

**POOLE - MC NIFF - COOK**  
**Apple II - Guida all'uso**  
Tutto quello che bisogna sapere sull'Apple. 400 pagine  
Cod. 331 P • Lire 31.000 / Abbonati Lire 24.800

**P. CAPOBUSSI**  
**Un Mac per amico** **NOVITA**  
Uso, applicazioni e programmi per Macintosh. 94 pagine  
Cod. 424 P • Lire 12.000 / Abbonati Lire 9.600

**P. CECIONI**  
**Guida ai PC IBM** **NOVITA**  
Tutto su tutti i Personal Computer del più famoso costruttore. 176 pagine  
Cod. 421 P • Lire 16.500 / Abbonati Lire 13.200

**F. SAMISH**  
**Guida alla scelta del Personal Computer**  
Caso per caso; per non sbagliare, per non pentirsi. 128 pagine  
Cod. 400 P • Lire 12.000 / Abbonati Lire 9.600

**P. LOTIGIE**  
**Un Personal Computer firmato IBM**  
Hardware, software e sistemi operativi. 160 pagine  
Cod. 404 H • Lire 15.000 / Abbonati Lire 12.000

**E. TONTI**  
**Didattica con il Personal Computer**  
Esercizi al calcolatore da fare in classe. 156 pagine  
Cod. 400 A • Lire 24.000 / Abbonati Lire 19.200

**M.-ROSACLOT**  
**La fisica con il computer: la dinamica**  
L'evoluzione temporale di un sistema fisico, anche complesso, attraverso la simulazione. 268 pagine  
Cod. 550 A • Lire 50.000 / Abbonati Lire 40.000

**JACKSON FA LEGGE  
LEGGI JACKSON**

# per hobby e per professione



R. BONELLI - D. GIANNI  
**M 20 la programmazione BASIC - PCOS**  
 Esempi e applicazioni. 360 pagine  
 Cod. 401 A • Lire 30.000 / Abbonati Lire 24.000

R. BONELLI  
**Il primo libro per M 24: MS DOS e GW BASIC**  
 Un'introduzione esauriente al sistema e al suo utilizzo.  
 152 pagine  
 Cod. 401 P • Lire 24.000 / Abbonati Lire 19.200

M. MANGIA  
**Olivetti M.10: guida all'uso**  
 Il più piccolo, versatile, straordinario Olivetti. 192 pagine  
 Cod. 401 B • Lire 18.000 / Abbonati Lire 14.400

G. MARANO  
**Epson HX 20**  
 Le meraviglie del primo calcolatore portatile con stampante incorporata. 230 pagine  
 Cod. 345 D • Lire 20.000 / Abbonati Lire 16.000

G. MARANO  
**Macintosh: 120 idee per il nuovo computer Apple**  
 Un libro di applicazioni, disegnato con il MacPaint, scritto con il MacWrite. 126 pagine  
 Cod. 402 D • Lire 30.000 / Abbonati Lire 24.000

## ENCICLOPEDIA

**ABC Personal Computer**  
 In tre volumi rilegati un pratico ed esauriente corso di BASIC. 680 pagine complessive + un volume Dizionario d'Informatica.  
 Cod. 160 A • Lire 150.000 / Abbonati Lire 120.000

**ABC QUIZ**  
 Test ed esercizi pratici di BASIC. 280 pagine  
 Cod. CIABC1 • Lire 40.000 / Abbonati Lire 32.000

## TASCABILI

P. GOSLING **NOVITA'**  
**PC IBM**  
 Una guida indispensabile nell'utilizzo del PC IBM.  
 74 pagine  
 Cod. 018 H • Lire 8.500 / Abbonati Lire 6.800

KING-WALLER **NOVITA'**  
**MS/DOS**  
 Tutto, e in breve, sul sistema operativo più diffuso.  
 54 pagine  
 Cod. 019 H • Lire 8.500 / Abbonati Lire 6.800

KING-WALLER **NOVITA'**  
**PC/DOS**  
 I comandi e le utility del sistema operativo del PC IBM.  
 58 pagine  
 Cod. 012 H • Lire 8.500 / Abbonati Lire 6.800

BLACKBURN-TAYLOR **NOVITA'**  
**CP/M**  
 Tutti i segreti e comandi del CP/M. 70 pagine  
 Cod. 011 H • Lire 8.500 / Abbonati Lire 6.800



**GRUPPO EDITORIALE JACKSON**

**JACKSON FA LEGGE  
 LEGGI JACKSON**



**Gli abbonati JACKSON hanno diritto, fino al 28-2-86, ad uno sconto del 20% su tutti i volumi.**

**N**ello standard MSX è stato adottato un microprocessore video familiare ai possessori del TI99/4A, il TI TMS-9929A o uno analogo (vedi TM9918 della Sony). Il modo in cui è organizzata la memoria in questi computer non si discosta molto da quello, ad esempio, di un Commodore 64. Per di più, mentre in quest'ultimo si deve necessariamente esercitare una lunga pratica con le Peek e le Poke, con il TI 99/4A non si può accedere normalmente alla VDP RAM, a meno che si sia espanso il sistema con un modulo Editor/Assembler. Il "texano", tuttavia, ha a disposizione una marcia in più per quanto riguarda alcuni modi d'accesso alla memoria-video: ci riferiamo alle istruzioni Call Char, Call Charpat e Call Charset, con le quali può modificare la mappa di memoria dove è conservata la forma dei caratteri, potendo così ridefinirli a suo piacimento. Perché dovrebbe essere necessario creare dei caratteri personalizzati

## Uno sguardo nella VDP RAM

### Comprendiamo meglio l'organizzazione della RAM

di Sergio Borsani

consiste nell'imparare come è organizzata la VDP RAM, cioè la memoria riservata al video, e modificare la forma (in alcuni manuali si parla di profilo o, con termine originale, di pattern) di alcuni caratteri. Sarà il loro accostamento, co-

ria provoca il panico, ed effettivamente non si deve usare questa istruzione (o una Vpoke) in modo indiscriminato, perché c'è il rischio di mandare in tilt il computer e dover procedere a un reset di sistema. Tuttavia la cosa non deve spaventare, perché la memoria è organizzata in modo molto semplice e lineare.

Consideriamo la VDP RAM di un qualsiasi sistema MSX. Ci sono due segmenti principali: il primo contiene i codici ASCII dei caratteri presenti in ogni locazione del video, il secondo contiene la forma dei caratteri (profilo o pattern). I dati sono posti sequenzialmente ed ogni locazione è contraddistinta da un indirizzo; conoscendo il primo e l'ultimo indirizzo di ogni tabella si può alterarne senza difficoltà il contenuto.

Poiché l'MSX-BASIC prevede due modi testo e due modi grafici, è opportuno riferirci a uno di questi in particolare. Partiamo dal modo testo 2, che è tra tutti quello che presenta minori difficoltà. Esso si seleziona con il comando/istruzione Screen 1 e comporta un video suddiviso in 24 righe di 32 caratteri, per un totale di 768 posizioni (il normale video del TI). Tali posizioni sono numerate dalla n. 0 alla n. 767, a partire dall'indirizzo contenuto nella variabile Base(5). Così, leggendo la mappa di memoria da Base(5) a Base(5) + 767 con l'istruzione Vpeek, troveremo i codici ASCII dei caratteri presenti sullo schermo, cominciando dall'angolo in alto a sinistra e procedendo nel senso della scrittura, da sinistra a destra e dall'alto in basso.



quando il set MSX è così ampio da comprendere anche lettere dell'alfabeto greco o simboli matematici, come quello che indica l'infinito, o l'esponente 2 o le frazioni 1/4, 1/2 e 3/4? Vi siete mai chiesti come si può creare un titolo a caratteri cubitali o un disegno per una scenografia? Utilizzare degli sprite non è consigliabile, dal momento che non ne possono coesistere più di quattro sulla medesima riga orizzontale. Il modo corretto per affrontare il problema

me in un puzzle o in un mosaico, a creare l'immagine desiderata.

Molto spesso a chi ha poca dimestichezza con la programmazione l'idea di usare una Poke per inserire un valore direttamente in una locazione di memo-

Quindi, invece di usare l'istruzione Print potremmo inserire un codice ASCII in una di queste locazioni usando l'istruzione Vpoke. Ad esempio, Vpoke Base(5)+384,42 fa apparire un asterisco nel bel mezzo dello schermo. Il computer possiede una tabella dalla quale può trarre la forma del carattere di codice ASCII 42 e naturalmente anche quella di tutti gli altri presenti sulla tastiera. La tabella dei profili è un po' più difficile da capire e per questo è necessario sapere come sono fatti i caratteri che appaiono sullo schermo.

Ogni simbolo è formato da tanti puntini luminosi (pixel); per ognuno ce ne sono 64, disposti a formare un quadratino di 8 x 8 punti. La figura 1, più di ogni altra spiegazione, farà capire come la diversa disposizione dei punti possa far apparire i caratteri che siamo soliti vedere sullo schermo. Tuttavia, se potessimo guardare direttamente con i nostri occhi il contenuto della memoria, non vedremmo niente di simile; da quel man-

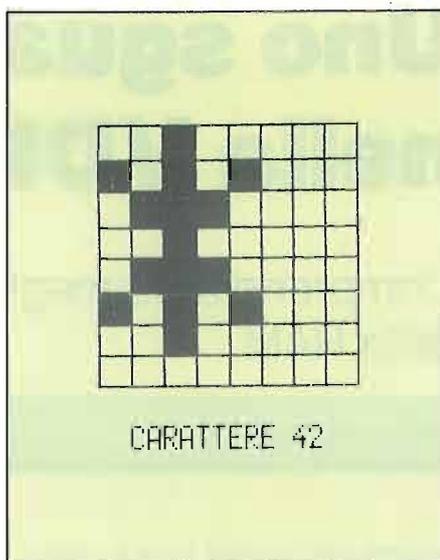


Figura 1 - Ogni carattere che appare sullo schermo è definito in una matrice di 8 x 8 punti. A un punto corrisponde un bit di memoria; sono quindi necessari 8 byte per memorizzare un carattere.

gia-numeri che è, il computer trasforma tutto in una sequenza di 0 e di 1, di stati On e di stati Off. Un byte è sufficiente a contenere l'equivalente binario di una serie di otto punti e quindi sono necessari 8 byte per un intero carattere.

La tabella dei profili inizia alla locazione Base (7) e si estende fino alla Base (7) + 2047. La forma del carattere di codice ASCII 42 è memorizzata negli 8 byte che vanno da Base(7)+42 ★ 8 a Base(7)+42 ★ 8+7. Indirizzando dei particolari valori decimali in queste locazioni di memoria, saremmo in grado di cambiare la forma dell'asterisco (codice ASCII 42), ma prima dobbiamo sapere quale numero far corrispondere a una particolare sequenza di punti.

Per chi ha qualche conoscenza sul sistema binario di numerazione, e non sono pochi, se si pensa che oggi lo si insegna comunemente nella prima Media, basterà dire di scrivere uno 0 per ogni punto spento ed un 1 per ogni punto acceso; si otterrà così un numero

**Listato 1 - Programma di prova per ottenere la sigla MSX in grande.**

```

10 REM CARATTERI PROGRAMMABILI
20 REM MOD0 TESTO 2
30 SCREEN 1:WIDTH 32:COLOR 15,4,4
40 B1=BASE(5):B2=BASE(7)
50 FOR W=1600 TO 1879:READ N
60 IF N<0 OR N>255 THEN 90
70 VPOKE B2+W,N: NEXT W
80 READ N
90 FOR RO=10 TO 13:FOR CO=10 TO 21
100 READ N:IF N<32 OR N>234 THEN END
110 LO=B1+(RO-1)*32+CO:VPOKE LO,N
120 NEXT CO:NEXT RO
130 END
140 DATA 15,51,103,79,255,255,255,255
150 DATA 0,192,224,240,248,252,254,255
160 DATA 0,3,7,15,31,63,127,255
170 DATA 240,156,62,254,255,255,255,255
180 DATA 0,3,7,14,15,31,31,31
190 DATA 255,128,15,127,255,255,255,255
200 DATA 255,0,255,255,255,255,255,255
210 DATA 255,15,255,255,255,255,255,255
220 DATA 248,254,255,255,255,255,255,255
230 DATA 0,0,0,128,192,224,240,248
240 DATA 0,0,0,1,3,7,15,31
250 DATA 60,102,207,191,255,255,254,252
260 DATA 255,255,255,255,255,255,255,255
270 DATA 255,255,255,255,255,63,31,15
280 DATA 255,255,255,255,255,252,248,240

```

```

290 DATA 31,31,31,31,31,31,31,15
300 DATA 240,224,224,240,255,255,255,255
310 DATA 0,0,0,0,255,255,255,255
320 DATA 0,0,0,0,224,248,252,254
330 DATA 63,15,7,3,1,0,0,0
340 DATA 252,254,255,255,255,255,127,63
350 DATA 63,127,255,255,255,255,254,252
360 DATA 248,240,224,192,128,0,0,0
370 DATA 7,3,0,0,0,0,0,0
380 DATA 224,192,0,0,0,0,0,0
390 DATA 15,7,3,0,0,0,0,128
400 DATA 255,255,255,255,0,0,0,0
410 DATA 255,255,255,255,1,0,0,1
420 DATA 254,255,255,255,255,255,255,255
430 DATA 255,255,255,255,255,126,126,60
440 DATA 255,255,255,127,127,63,31,7
450 DATA 255,255,255,254,254,252,248,224
460 DATA 63,127,255,255,255,255,126,60
470 DATA 31,15,7,3,1,0,0,0
480 DATA 252,254,255,255,255,255,126,60
490 DATA 999
500 DATA 200,201,202,203,204,205
510 DATA 206,207,208,209,210,211
520 DATA 212,213,214,212,215,216
530 DATA 217,218,219,220,221,222
540 DATA 212,223,224,212,225,226
550 DATA 227,228,210,221,220,209
560 DATA 229,32,32,230,206,212
570 DATA 212,231,232,222,233,234
580 DATA 999

```

# Il tuo "cervello" in più è Giapponese

I giapponesi sono piccoli? Alcuni sì. Ma sono anche molto robusti, veloci di riflessi e dotati di un'intelligenza viva e poliedrica. Proprio come **TOSHIBA T1100**, il personal computer così piccolo e maneggevole da entrare comodamente in una 24 ore, ma così potente da poter competere con i personal da tavolo di prezzo notevolmente superiore. Perché **TOSHIBA T1100**, oltre ad avere un rapporto prezzo-prestazioni eccezionale è tecnologicamente superiore e **totalmente compatibile con il PC IBM®** e con il suo software.

Con una memoria di 256 Kbyte, espandibile a 512 Kbyte, un video a cristalli liquidi con una risoluzione grafica di 640x200 pixel, oltre all'interfaccia standard RGB e video composito, un disk drive interno da 720 Kbyte, la possibilità di un secondo drive esterno e un'autonomia di oltre otto ore con le batterie ricaricabili, **TOSHIBA T1100** è molto di più di un personal portatile. È un vero cervello in più, ma non pensa avidamente al vostro denaro.

TL&DB



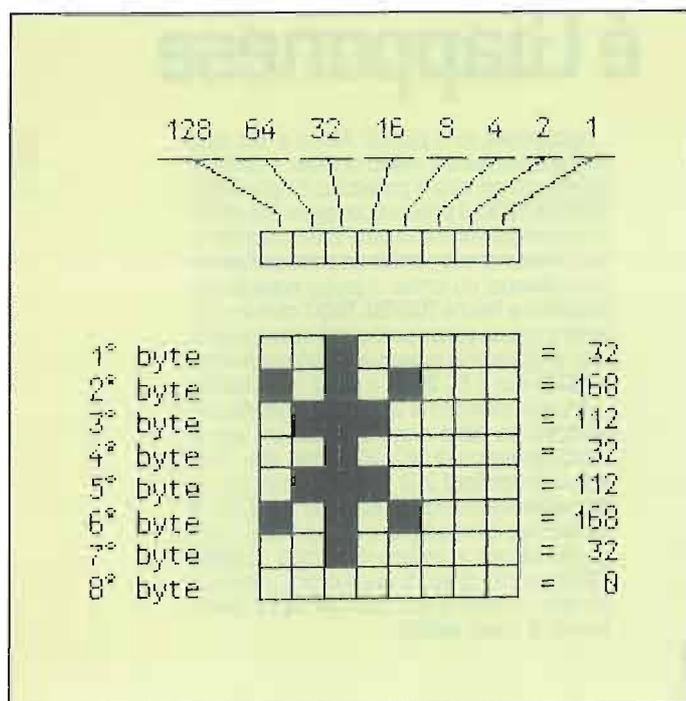
## Toshiba T1100



**TOSHIBA**  
**COMPUTER**  
**Made in Japan**

IBM e IBM PC sono marchi registrati dalla International Business Machines Corporation

<b>TIBER</b>	PC Attrezzature Ufficio SpA Via Madonna del Riposo, 127 00165 Roma - Tel. 06/6236741
Desidero maggiori informazioni sul TOSHIBA T1100	
Nome _____	
Indirizzo _____	
Telefono _____	



**Figura 2 - Ai punti che formano il carattere (foreground) corrispondono bit in posizione On, mentre agli altri (background) corrispondono bit nello stato Off. Poiché ogni bit ha un valore decimale che dipende dalle potenze di 2, non è difficile ricavare il numero decimale associato a una qualsiasi disposizione di 8 punti.**

binario, dal quale si ricaverà il corrispondente decimale. Per chi invece è ancorato a nozioni matematiche più tradizionali, senza la pretesa di trattare esaurientemente l'argomento, daremo una semplice regola pratica. Disegnata una fila di 8 punti, in parte accesi e in parte spenti, assegnare ad essi i seguenti valori, cominciando da destra e procedendo verso sinistra: per ogni punto spento valore 0; per quelli accesi, 1, se il punto è nella prima posizione, 2, se è nella seconda, 4 nella terza, 8 nella quarta, 16 nella quinta, 32 nella sesta, 64 nella settima e 128 nell'ottava; al termine sommare tutti i valori così ottenuti (vedi figura 2).

Il listato 1 è un programma di prova per ottenere la sigla MSX in formato gigante, tale da richiedere un totale di  $12 \times 4 = 48$  normali caratteri. La scritta è stata ottenuta prima su un foglio simile alla carta millimetrata; da questa si sono potuti ottenere i valori decimali contenuti nelle istruzioni Data.

**Listato 2 - Ora la sigla MSX è stampata nel modo testo 1.**

```

10 REM CARATTERI PROGRAMMABILI
20 REM MODO TESTO 1
30 SCREEN 0:WIDTH 40:COLOR 15,4,4
40 B1=BASE(0):B2=BASE(2)
50 FOR W=1600 TO 1879 : READ N
60 IF N<0 OR N>255 THEN 90
70 VPOKE B2+W,N : NEXT W
80 READ N
90 FOR RO=10 TO 13 : FOR CO=14 TO 25
100 READ N : IF N<32 OR N>234 THEN END
110 LO=B1+(RO-1)*40+CO : VPOKE LO,N
120 NEXT CO : NEXT RO
130 END
140 DATA 60,68,140,188,188,252,252,252
150 DATA 0,192,224,240,248,252,252,252
160 DATA 0,12,24,60,124,252,252,252
170 DATA 240,24,124,252,252,252,252,252
180 DATA 4,8,24,20,20,28,28,28
190 DATA 252,0,124,252,252,252,240,224
200 DATA 252,0,252,252,252,252,0,0
210 DATA 252,60,252,252,252,252,0,0
220 DATA 240,252,252,252,252,252,252,124
230 DATA 0,0,128,128,192,224,224,240
240 DATA 0,0,4,4,12,28,28,60
250 DATA 112,152,188,124,252,252,248,248
260 DATA 252,252,252,252,252,252,252,252
270 DATA 252,252,60,28,12,0,0,0
280 DATA 252,252,240,224,192,0,0,0

```

```

290 DATA 28,28,28,28,28,28,28,28
300 DATA 224,224,224,224,224,240,252,252
310 DATA 0,0,0,0,0,0,252,252
320 DATA 0,0,0,0,0,0,240,248
330 DATA 60,60,28,12,12,4,0,0
340 DATA 248,252,252,252,252,252,252,124
350 DATA 124,252,252,252,252,252,252,248
360 DATA 240,224,224,192,128,128,0,0
370 DATA 28,28,12,4,0,0,0,0
380 DATA 252,252,252,252,0,0,0,0
390 DATA 252,252,252,252,4,0,0,0
400 DATA 252,252,252,252,252,252,120,48
410 DATA 252,252,252,252,252,252,124,60
420 DATA 0,128,252,252,252,252,252,252
430 DATA 0,0,252,252,252,252,252,252
440 DATA 0,4,252,252,252,252,252,252
450 DATA 252,252,252,252,252,252,248,240
460 DATA 60,124,124,124,124,124,60,28
470 DATA 240,248,248,252,252,252,248,112
480 DATA 252,254,255,255,255,255,126,60
490 DATA 999
500 DATA 200,201,202,203,204,205
510 DATA 206,207,208,209,210,211
520 DATA 212,213,214,212,215,216
530 DATA 217,218,219,220,221,222
540 DATA 212,32,32,212,223,224
550 DATA 225,212,210,221,220,209
560 DATA 226,32,32,227,228,229
570 DATA 230,231,232,222,219,233
580 DATA 999

```



# PAINTSTAR

## Paintstar colora le tue idee

Paintstar rende facile creare immagini a colori sullo schermo dell'Apple II con il semplice uso di un joystick. Paintstar è comandato a icone, simili a quelle del Macintosh, e ti offre una completa scelta di comandi grafici, con risposta immediata sul video.

Puoi disegnare a mano libera, o usare linee, rettangoli, cerchi, ellissi. Puoi usare una penna od un pennello della forma e del colore desiderati.

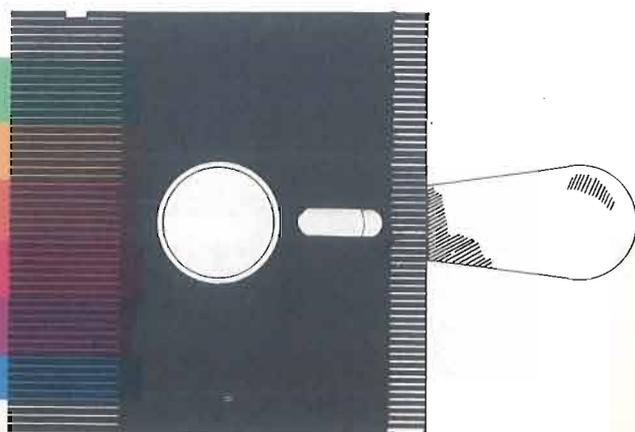
## Riesce meglio con Paintstar

Paintstar ti offre gli strumenti per disegnare con poco lavoro e molti risultati. Puoi spostare oggetti e copiarli. Puoi riempire aree del colore desiderato, oppure con un retino di tuo disegno. Puoi scrivere, usando caratteri dello stile preferito

E, naturalmente, puoi anche stampare il risultato del tuo lavoro, oltre a registrarlo su disco.

Paintstar ti dà molto con poco.

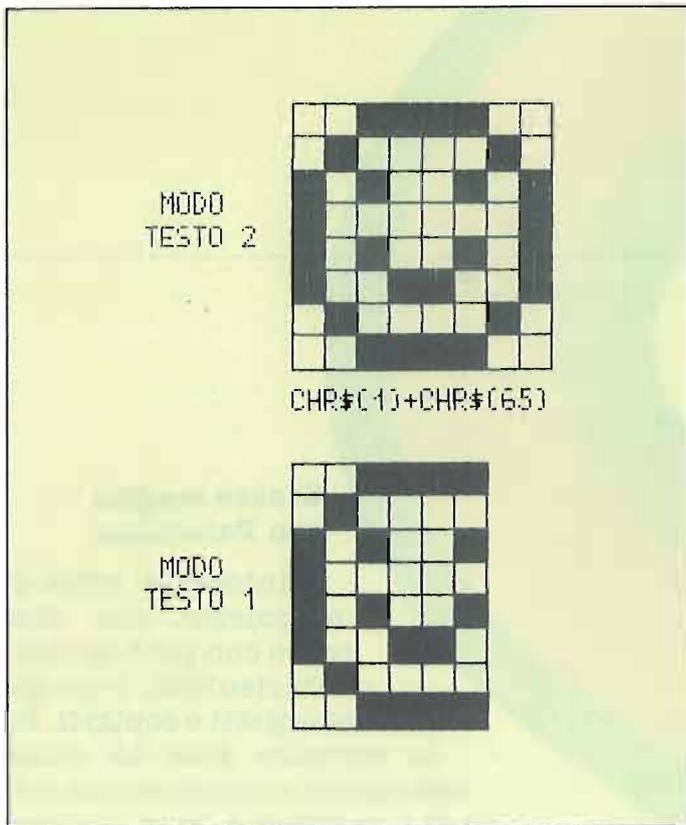
Paintstar è per Apple IIc e IIe. Richiede joystick.



software di qualità  
**J.soft**

Viale Restelli, 5 - 20124 Milano - Tel. 02/6888228 - 683797 - 6880841 - 6880842 - 6880843

L'acquisto di software originale significa: garanzia di prodotto, possibilità di sostituzione anche nel caso di versioni aggiornate o di versioni nazionalizzate. Esigete sempre software originale.



**Figura 3 - Nel modo testo 1 per consentire la stampa di 40 caratteri per riga vengono mostrate solo le prime 6 colonne di punti ed alcuni simboli grafici vengono così troncati; nel modo testo 2 la matrice è di 8 x 8 punti.**

da Base (2) e si estende fino a Base (2) + 2047, come in precedenza, dato che il numero dei caratteri è sempre lo stesso.

Un'ulteriore differenza è dovuta al fatto che per far rientrare 40 caratteri in una riga questi devono essere più stretti dei precedenti.

Nel modo testo 2 i simboli vengono definiti in una matrice di 8 x 8 punti: normalmente le prime cinque colonne di punti contengono il carattere, mentre le ultime tre garantiscono la spaziatura tra un carattere e l'altro.

Nel modo-testo 1 i caratteri vengono sempre definiti in una matrice di 8 x 8 punti, ma solo le prime 6 colonne vengono visualizzate sullo schermo, mentre le ultime due vanno perdute. Si può rendere evidente questo fatto stampando, ad esempio, il simbolo grafico che riproduce la faccia di un omino nei due modi testo; si noterà come in un caso la faccia non sia completa (vedi figura 3). Bisogna quindi far attenzione nel caricare in memoria i valori decimali con le VPoke, poiché per ogni byte sono significativi solo i primi 6 bit di sinistra, con i valori rispettivamente di 128, 64, 32, 16, 8 e 4; i due bit di destra possono indifferentemente esser posti in On o in Off.

Il listato 2 contiene un programma dimostrativo per stampare la sigla MSX nel modo testo 1.

Un ultimo consiglio. Se necessario è preferibile ridefinire i caratteri con numero di codice ASCII più alto e riservare i simboli alfanumerici per i testi e per i messaggi che solitamente accompagnano ogni programma.

Non si tocchi il carattere 32, cioè lo space, per non incorrere in strane visioni delle quali possono godere solo i miracolati. I numeri hanno un codice che va dal 48 (lo zero) al 57 (il nove), le lettere maiuscole occupano l'intervallo da 65 a 90, mentre le minuscole quello da 97 a 122: la sorte quindi tocca a quei simboli che hanno un codice ASCII maggiore, di solito dal 128 in poi e non oltre il 255, ultimo carattere del set.

Il programma consiste essenzialmente in due cicli For-Next, linee 50-80 e 90-120, e due gruppi corrispondenti di istruzioni Data. Il primo ciclo modifica la forma dei caratteri a partire da quello di codice 200; la linea 60 controlla che il valore sia lecito, cioè tra 0 e 255, per prevenire errori di trascrizione o di battitura che non sono infrequenti. Il secondo ciclo legge l'altro gruppo di istruzioni Data, contenenti i numeri dei codici ASCII dei caratteri modificati e li carica

nella mappa di memoria del video, dopo che la linea 110 ha calcolato la giusta locazione (LO) a partire dal numero di riga (RO) e di colonna (CO).

Nel modo testo 1, ottenibile con l'istruzione Screen 0, le cose non vanno molto diversamente. Lo schermo viene diviso in 24 righe di 40 caratteri ciascuna; in tal modo la mappa di memoria consta di 24 x 40 = 960 locazioni, iniziando dall'indirizzo contenuto nella variabile Base(0). La tabella dei profili dei caratteri inizia



**Figura 4 - I due programmi dimostrativi, di cui sono stati riportati i listati, ridefiniscono alcuni caratteri, modificando il contenuto della tabella dei profili, in modo da creare la sigla MSX in formato gigante.**

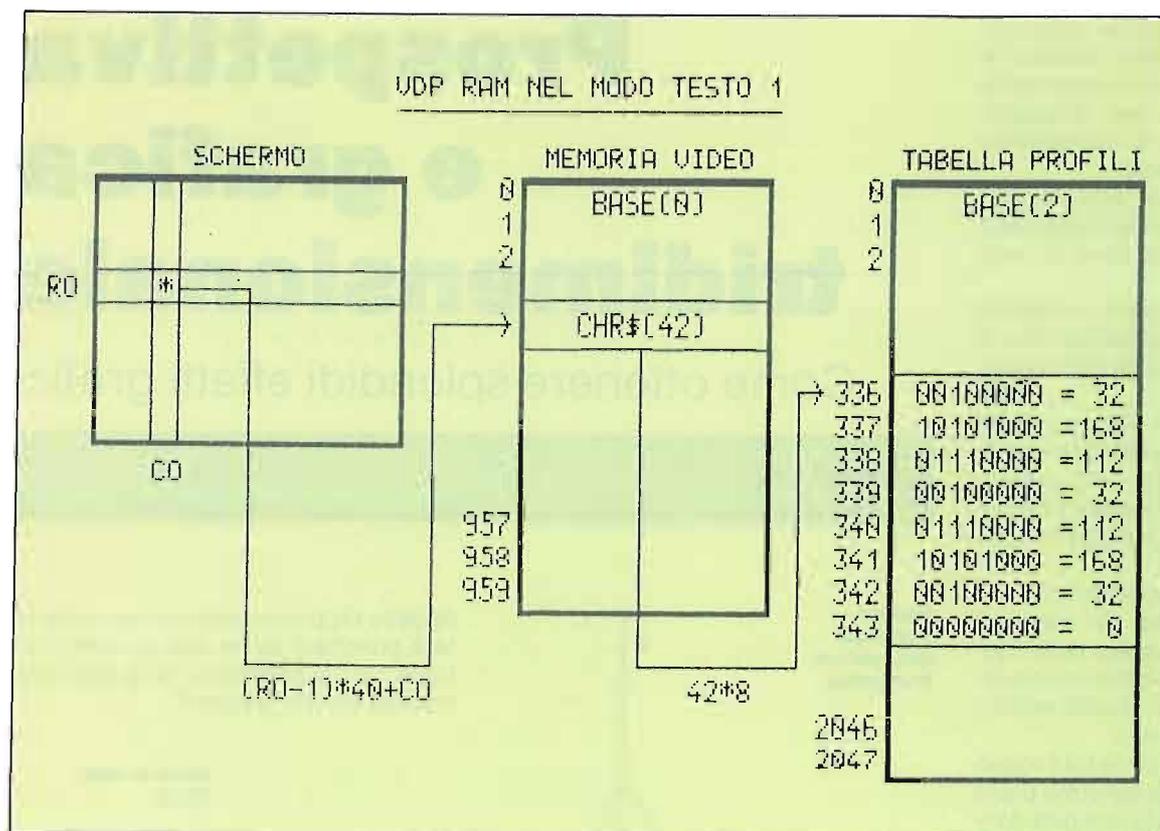


Figura 5 - Le parti più importanti della VDP RAM sono la mappa di memoria del video e la tabella dei profili, cioè delle forme dei caratteri. Ogni elemento della mappa contiene il numero di codice ASCII del carattere presente in una data posizione dello schermo e ad ogni carattere corrisponde una sequenza di 8 byte nella tabella dei profili.

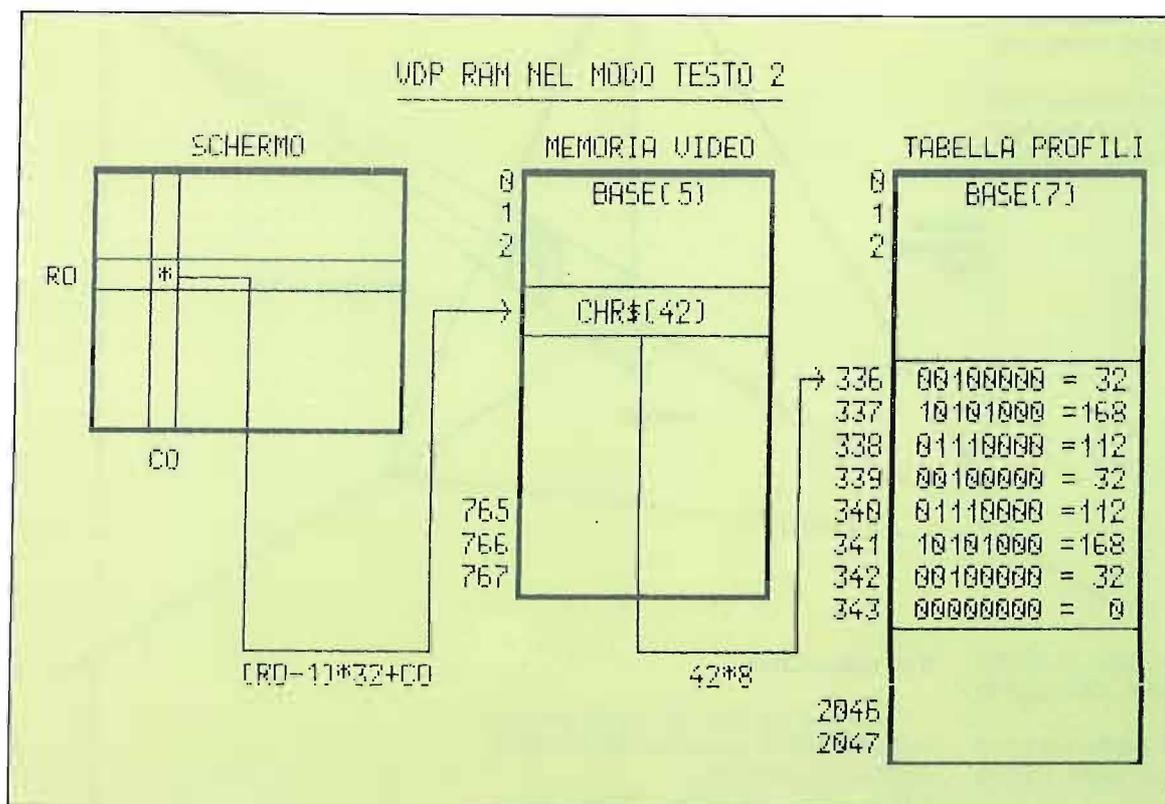


Figura 6 - Nel modo testo 2 la mappa di memoria del video è più limitata. In compenso i simboli grafici vengono visualizzati in modo più completo ed esiste una maggiore libertà nell'uso dei colori, in virtù di una tabella di memoria non riprodotta nella figura.

**L'** uso degli home computer per programmi didattici si sta estendendo soprattutto grazie alla loro diffusione capillare nelle famiglie e ultimamente anche in numerose scuole.

Particolare importanza assumono i programmi grafici, principalmente per il supporto che l'immagine visiva fornisce all'intuizione.

In quest'ottica si pongono numerose routine di utilità, le quali permettono la rappresentazione di figure tridimensionali sullo schermo.

Anche questo articolo si occupa del problema della rappresentazione tridimensionale, presentando un programma implementato sul C 64 con l'utilizzo dell'ormai insostituibile tool Simons' BASIC.

L'idea è semplice; supponiamo di porre un oggetto (un dado, per esempio) sopra un tavolo. Ogni vertice della figura verrà osservato dal vostro occhio situato in un punto dello spazio esterno all'oggetto (punto di vista).

Immaginate ora di frapporre tra l'oggetto e il vostro occhio uno schermo piano trasparente, in modo tale che guardando l'oggetto, possiate tracciare sul medesimo i vertici così dove vi appaiono e successivamente i lati che collegano i suddetti vertici.

Otterrete così (figura 1) un disegno bidimensionale sul piano convenzionalmente posto come schermo, che con-

serverà per il vostro occhio lo stesso aspetto configurazionale dell'oggetto tridimensionale.

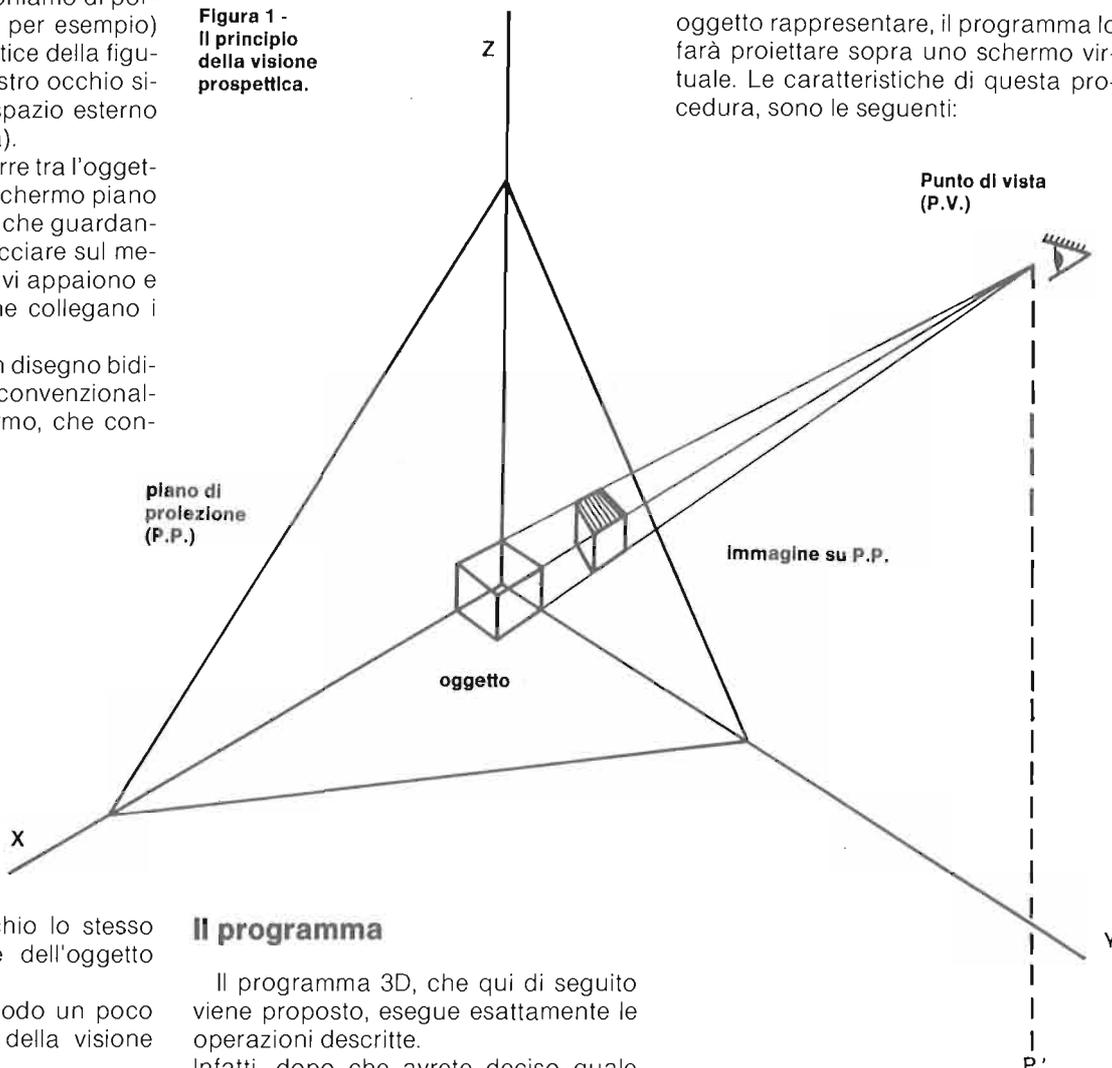
È questo, descritto in modo un poco semplificato, il principio della visione prospettica.

# Prospettiva e grafica tridimensionale

## Come ottenere splendidi effetti grafici

di Stefano Checchini

**Figura 1 - Il principio della visione prospettica.**

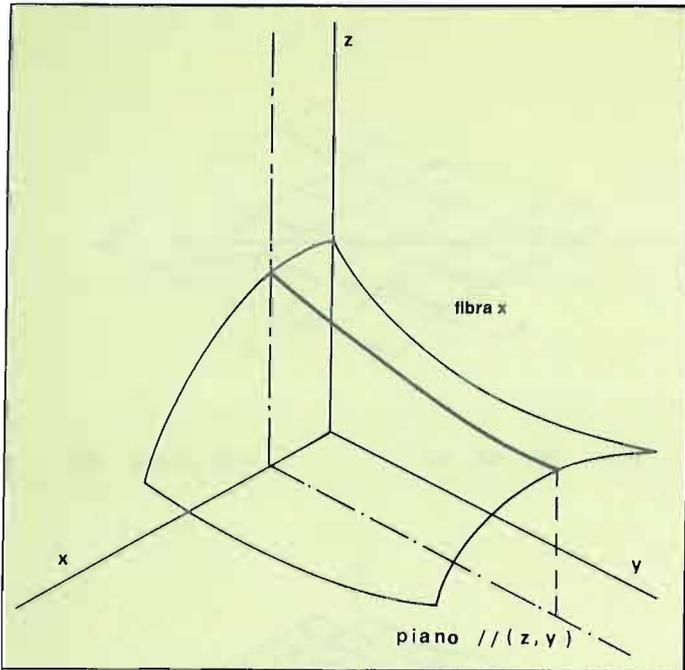


oggetto rappresentare, il programma lo farà proiettare sopra uno schermo virtuale. Le caratteristiche di questa procedura, sono le seguenti:

### Il programma

Il programma 3D, che qui di seguito viene proposto, esegue esattamente le operazioni descritte.

Infatti, dopo che avrete deciso quale

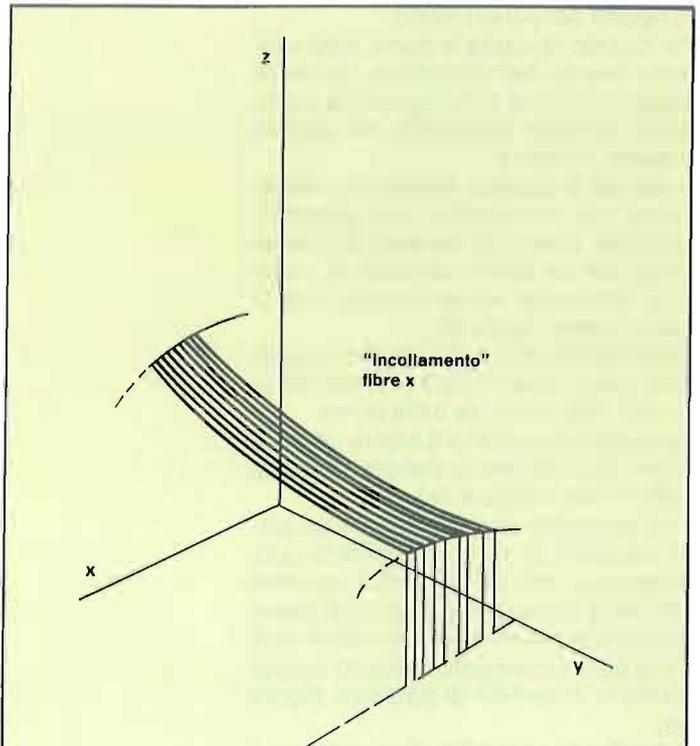


**Figura 2 -**  
Rappresentazione  
di superfici funzionali  
mediante  
l'individuazione  
delle fibre x.

delle figure poliedriche.

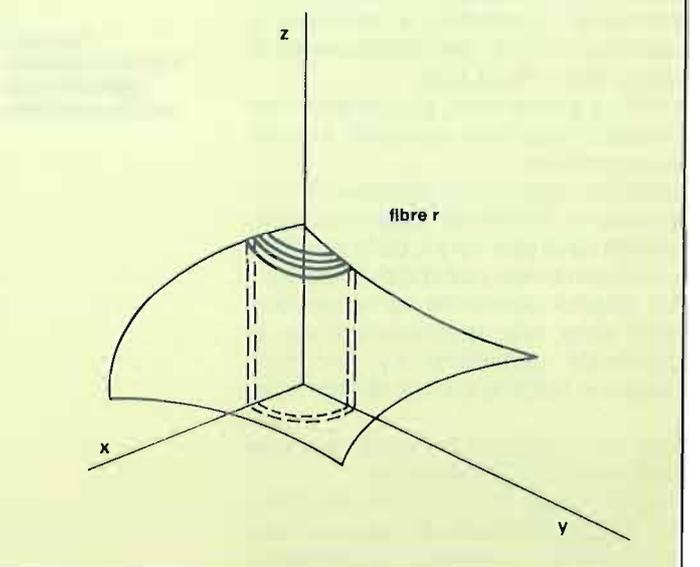
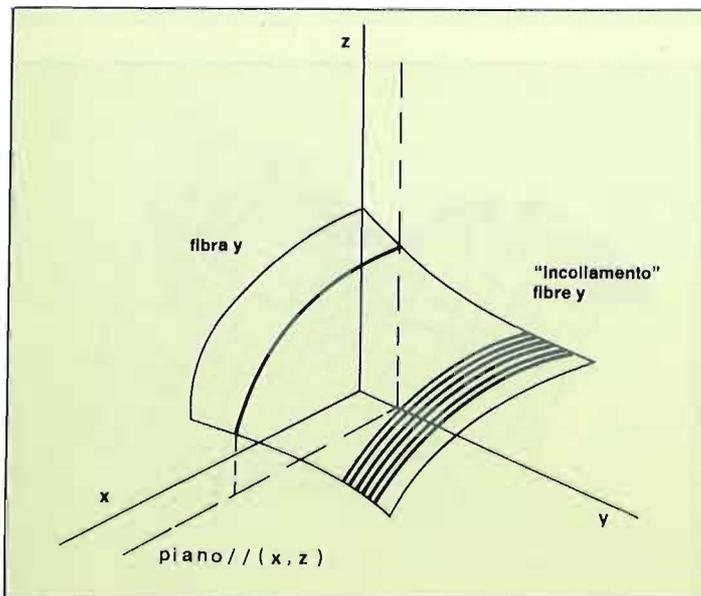
L'effetto di questa impostazione è che l'oggetto vi apparirà piccolo o grande a seconda di come spostiate la posizione del piano virtuale, e inoltre potrete osservarlo da ogni posizione.

Non sempre l'immagine sarà "elegante", soprattutto se scegliete i parametri a caso, ma dopo qualche esperimento



**Figura 3a,b,c -**  
Incollamento  
della fibra x, y ed r.

- 1) il punto di vista è scelto dall'operatore e può essere un qualsiasi punto dello spazio (con la sola esclusione dei punti che sono sulla verticale dell'oggetto);
- 2) il piano virtuale può essere posto in qualsiasi punto tra l'oggetto e il punto di vista;
- 3) il riferimento tridimensionale è mantenuto al centro dello schermo virtuale;
- 4) non vi sono limiti al numero di vertici



potrete osservare il vostro oggetto dall'angolo visuale più opportuno.

## Superfici e solidi

La vista prospettica è stata utilizzata per osservare tre tipi fondamentali di oggetti tridimensionali:

- a) superfici funzionali [funzioni  $z = f(x,y)$ ];
- b) superfici empiriche [ per punti ];
- c) oggetti poliedrici [solidi].

Per quanto riguarda le prime, esse vengono inserite dall'utilizzatore, tramite la legge funzionale, nel programma e vengono tracciate scegliendo tra quattro possibili modalità.

A questo proposito ricordiamo brevemente che, considerata una superficie qualsiasi, possiamo pensare di intersecarla con un piano parallelo al piano  $(z,y)$ , ottenendo come risultato una linea spaziale (figura 2).

Chiameremo tale linea una fibra  $x$ . Ogni altro piano parallelo al primo individuerà una fibra  $x$  diversa dalla prima.

La superficie può allora essere pensata come l'accostamento (incollamento) di tutte le fibre  $x$  (figura 3a).

Ma possiamo anche condurre un piano parallelo al piano coordinato  $(x,z)$ , ottenendo una nuova curva spaziale che verrà detta fibra  $y$ ; ogni altro piano parallelo al primo individua tutte le fibre  $y$  che per incollamento formano nuovamente la superficie di partenza (figura 3b).

Vi è infine la possibilità di creare fibre  $r$ , ottenute intersecando la superficie con tanti cilindri concentrici, e ricavando la superficie ancora per incollamento di tutte le fibre  $r$  (figura 3c).

Questo a grandi linee, supponendo per le superfici una certa regolarità, è quello che si può fare.

Nel nostro caso si può scegliere da programma di vedere la superficie come formata dalle sole fibre  $x$ , dalle sole fibre  $y$ , o da entrambe, nonché dalle fibre  $r$ .

Le quattro possibilità corrispondono ovviamente alla rappresentazione in coordinate cartesiane  $x,y$  (singolarmente o in coppia) e in coordinate polari.

Tutto ciò consente uno studio flessibile delle superfici tridimensionali.

Ma in molti casi non si ha a disposizione una legge funzionale per rappresentare la superficie; si ha invece un insieme di

Figura 4a,b -  
Rappresentazione  
di valori di quota  
utilizzando  
un reticolo  $x,y$ .

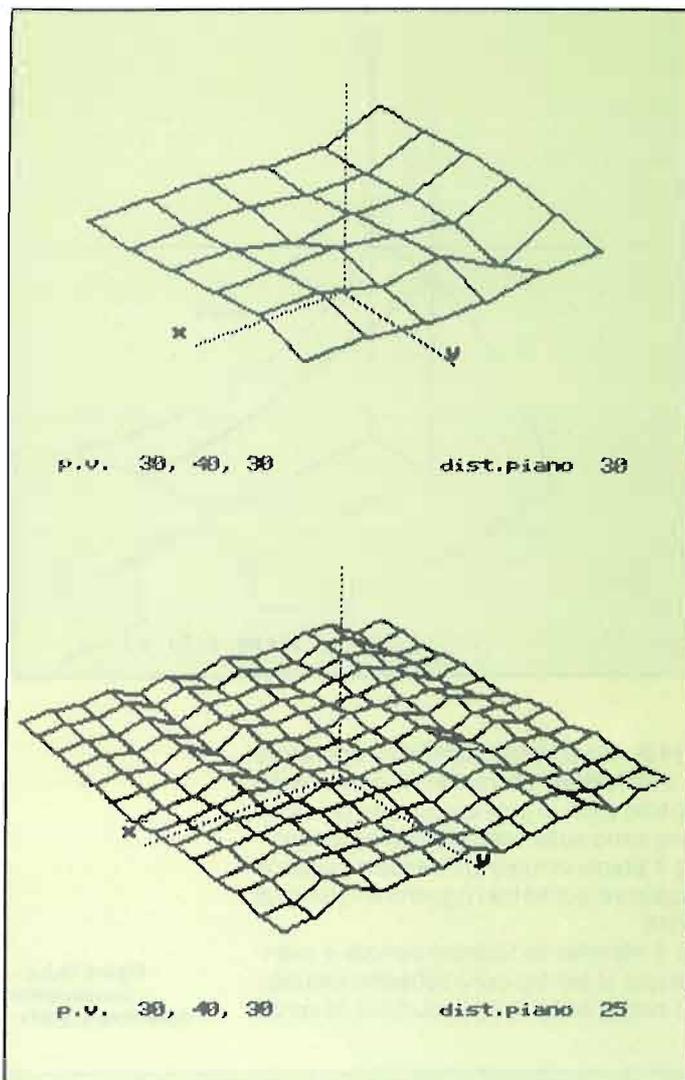
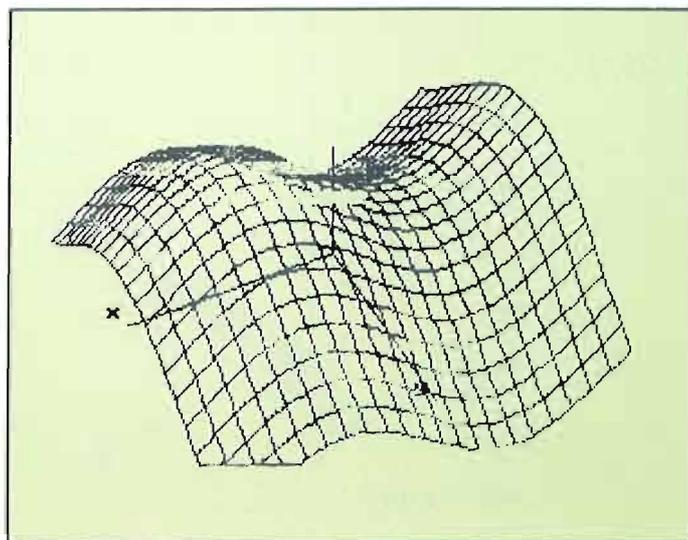


Figura 5 -  
Funzione empirica  
approssimante  
una più complessa



# COMMODORE 64

misure ottenute praticamente (si pensi alle quote di una certa zona geografica).

In questo caso è possibile che si abbia a disposizione un insieme di valori di quota misurati secondo un certo reticolo (x,y).

Il programma dà la possibilità di inserire i valori misurati tramite un reticolo cartesiano, e di tracciare la superficie relativa.

Facciamo un esempio.

Supponiamo di avere a disposizione un reticolo 6 x 6, posizionato al centro del sistema di riferimento. Abbiamo 49 differenti quote da memorizzare.

Se utilizziamo come parametri la massima espansione x e y del reticolo (che supporremo sempre simmetrico rispetto ad entrambi gli assi) e i quarantanove valori di quota, il programma è in grado di tracciare la superficie corrispondente, utilizzando il reticolo (x,y) (figura

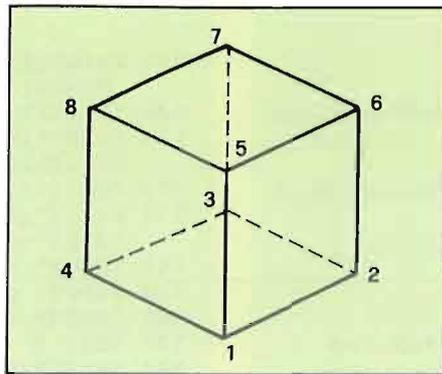


Figura 6 - Numerazione degli 8 vertici di un cubo.

4a,b).

Uso interessante di questa modalità di rappresentazione, può essere quello di verificare la forma di una funzione empirica  $P(x,y)$  approssimante di una più complessa funzione  $F(x,y)$ . In figura 5 si

vede un esempio.

Un capitolo a parte, per l'uso che può essere fatto, è quello riguardante gli oggetti poliedrici.

La maggior difficoltà in questi casi sta nel fatto che occorre comunicare non solo la posizione dei vertici, ma anche come sono collegati.

Il problema è stato risolto nel modo seguente.

Il numero dei vertici, a scelta dall'utilizzatore, non ha limiti se non quelli imposti dalla memoria, dal buon senso e dalla pazienza.

In realtà se il numero dei vertici è inferiore a 20, il programma esegue una routine semplificata che aiuta l'utente, se è superiore occorre digitare in modo diretto le informazioni richieste.

Per intenderci, nel primo caso, dopo aver dichiarato il numero dei vertici e aver caricato le rispettive coordinate; compare una matrice di incidenza che deve

## Listato 1 - Il programma 3D.

```

1 PRINT "{CLR}":
  PRINT "{ 2 SPAZI} [<M>] [<*>] {RVS}
  { 4 SPAZI} [<*>] {OFF} [<M>] [<*>] {RVS}
  { 3 SPAZI} [<*>] {OFF} { 2 SPAZI}"
2 PRINT "{ 2 SPAZI} [<M>] O [< 3 Y>] P [<M>]
O [< 3 Y>] M { 2 SPAZI}"
3 PRINT "{ 3 SPAZI} [<*>] L [< 2 P>] [<N>]
 [<M>] [<H>] { 4 SPAZI} M"
4 PRINT "{ 4 SPAZI} N N N [<M>] [<H>] OM
 [<N>]"
6 PRINT "{ 4 SPAZI} [<*>] [<H>] [<Y>] M [<M>]
 [<H>] [<H>] [<N>] [<N>]"
7 PRINT "{ 5 SPAZI} P M
 [<H>] [<N>] [<N>]"
8 PRINT "{ 5 SPAZI} N [<M>] [<N>] [<M>] [<H>]
LN [<M>}"
9 PRINT " [<M>] [<*>] {RVS} { 3 SPAZI} {OFF}
E [<N>] [<M>] [<H>] { 4 SPAZI} N"
10 PRINT " [<M>] O [< 2 Y>] { 2 SPAZI} N
 [<*>] L [< 3 P>] N Z Z"
11 PRINT "{ 2 SPAZI} [<*>] L [< 3 P>] N BY CHEC
CHINI"
13 PAUSE 5
14 REM -----MENU'-----
15 ON ERROR:GOTO 6000
20 PRINT "{CLR}";:
25 PRINT "{RVS}{BLU} PROCEDURE { 30 SPAZI}
":PRINT "{ 6 GIU' }"
30 PRINT "{RED}1) FUNZIONI IN COORDINATE
CARTESIANE":PRINT
32 PRINT "{RED}2) SUPERFICI NON FUNZIONAL
I (A PUNTI)":PRINT

```

```

35 PRINT " {RED}3) POLIEDRI { 24 SPAZI}":PR
INT
40 PRINT:PRINT:INPUT " {RVS}QUALE PROCEDUR
A ?":PR : PRINT "{CLR}";:PRINT:POKE 680
,INT(PR)
41 PR=INT(PR)
45 ON INT(PR) GOTO 400,900,500
50 REM-----INIZIALIZZAZIONE-----
53 UX=40:UY=40
55 PRINT "{CLR}":PRINT "{BLU} INTRODUCI LE
COORDINATE DEL PUNTO DI":PRINT " VISTA
!":PRINT
60 PRINT " {RVS}{BLK}";X0;:PRINTTAB(4) "
{OFF}>";:INPUT " {RVS}COORDINATA X=";X0
65 PRINT " {RVS}{BLK}";Y0;:PRINTTAB(4) "
{OFF}>";:INPUT " {RVS}COORDINATA Y=";Y0
75 PRINT " {RVS}{BLK}";Z0;:PRINTTAB(4) "
{OFF}>";:INPUT " {RVS}COORDINATA Z=";Z0
:PRINT
80 PRINT:PRINT " {RVS}{BLK}";DS;:PRINTTAB(
4) "{OFF}>";:INPUT " DISTANZA PIANO DI P
ROIEZIONE";DS
82 IF DS>SQR(X0^2+Y0^2+Z0^2) THEN PRINT "
{RVS}DISTANZA OLTRE IL PUNTO DI VISTA"
:GOTO 80
90 PRINT
94 PRINT " DIMENSIONI DOMINIO DI DEFINIZIO
NE":PRINT " {RVS}{BLK} VALORI ATTUALI
{OFF} {RVS}";A1;B1
95 PRINT
96 INPUT " DIMENSIONE X=";A1:INPUT " DIMENS
IONE Y=";B1:INPUT " DIMENSIONE Z=";C1

```

Seguito Ilistato 3D.

```

97 IF DS<SQR(A1↑2+B1↑2) THEN PRINT "{RVS} DIMENSIONI TROPPO AMPIE PER IL P.VISTA":GOTO 96
98 REM UX=(160/A1 OR 20):REM UY=(160/B1 OR 20)
100 REM---CALCOLI PER IL PIANO-----
      DI PROIEZIONE
110 P0=X0*X0+Y0*Y0:P1=SQR(P0)
120 D0=SQR(X0*X0+Y0*Y0+Z0*Z0)
130 XS=X0*(D0*DS)/P0:YS=Y0*(D0*DS)/P0
140 D1=SQR(XS*XS+YS*YS)
150 XC=X0*DS/D0:YC=Y0*DS/D0:ZC=Z0*DS/D0
155 REM---TRACCIA DEL RIFERIMENTO-----
160 HIRES 1,0
170 Y=0:Z=0
175 X=A1
180 GOSUB 4100
190 FOR V=0 TO 1 STEP .03:
      PLOT 160+V*PX,100+V*PY,1:NEXT
195 TEXT150+PX,90+PY,"{B}X",1,1,6
200 X=0:Y=B1:Z=0
210 GOSUB 4100
220 FOR V=0 TO 1 STEP .03:
      PLOT 160+V*PX,100+V*PY,1:NEXT
225 TEXT155+PX,90+PY,"{B}Y",1,1,6
230 X=0:Y=0:Z=C1
240 GOSUB 4100
250 FOR V=0 TO 1 STEP .03:
      PLOT 160,100-V*98,1:NEXT
265 TEXT15,192,"{B} P.V. "+STR$(X0)+","+STR$(Y0)+","+STR$(Z0),1,1,6
270 TEXT185,192,"{B} DIST.PIANO "+STR$(D0),1,1,6
275 PAUSE 5:
280 ON PEEK(680) GOTO 1600,1600,3000
400 REM-----INPUT DELLA FUNZIONE-----
405 PRINT "{CLR}";:PRINT "{ 3 GIU' } {DES} {RVS} INTRODUCI LA FORMULA DELLA FUNZIONE ":PRINT:PRINT
410 INPUT "{ 2 SPAZI } F(X,Y) = ";FU$
420 POKE 198,3:POKE 631,19:POKE 632,13:POKE 633,13
430 PRINT "{CLR} [<8>] 480 DEF FNA(X) = "FU$":RETURN"
440 PRINT "[<8>] GOTO 450":SYS 42115
450 GOSUB 480
460 BT=0: GOTO 50
480 DEF FNA(X) = 3*SIN(X↑2+Y↑2)/(X↑2+Y↑2):RETURN
500 REM-----ROUTINE POLIEDRI-----
505 CSET 0:PRINT "{CLR}";:PRINT "{RVS} {RED} POLIEDRI TRIDIMENSIONALI { 15 SPAZI } {OFF}"
510 PRINT "POLIEDRO DI PIU' DI 20 VERTICI? (S/N) {RVS}";:INPUT N$:IF N$="S" THEN 810
515 PRINT
520 DIM PO(20,2),IN(20,20),CO(20,3) :

```

```

525 PRINT:INPUT "{RVS} {RED} LA FIGURA E' IN MEMORIA? (S/N)";MM$:PRINT
526 IF LEFT$(MM$,1)="S" THEN 4700
528 PRINT:INPUT "QUANTI SONO I VERTICI ?";NV:PRINT
530 FOR I=1 TO NV
535 PRINT "{RVS} COORDINATE DEL VERTICE {OFF}";I:
540 INPUT "COORDINATA X=";CO(I,1)
550 INPUT "COORDINATA Y=";CO(I,2)
560 INPUT "COORDINATA Z=";CO(I,3)
570 NEXT I
580 PRINT "{CLR}";:PRINT "{RVS} CORREZIONI { 29 SPAZI } {OFF}"
585 PRINT:PRINT "VI SONO CORREZIONI DA FARE ? (S/N)";
590 INPUT AA$
592 IF LEFT$(AA$,1)(">"S" THEN 600
594 PRINT:INPUT "{RVS} VERTICE ? {OFF}";VT:VT=INT(VT) :IF VT>NV THEN 594
595 INPUT "COORDINATA X=";CO(VT,1)
596 INPUT "COORDINATA Y=";CO(VT,2)
597 INPUT "COORDINATA Z=";CO(VT,3)
598 PRINT:PRINT "{RVS} {RED} ANCORA ? (S/N)";:INPUT AA$
599 IF LEFT$(AA$,1)="S" THEN 594
600 PRINT "{CLR}";:PRINT "{RVS} {BLU} { 9 SPAZI } MATRICE DI INCIDENZA { 11 SPAZI }"
610 PRINT "{ 3 SPAZI }";
620 FOR J=1 TO NV
625 IF J>9 THEN POKE 1065+J,48+INT(J/10)
630 POKE 1105+J,48+J-10*INT(J/10)
640 NEXT J:PRINT
650 FOR J=1 TO NV
660 PRINT J
670 NEXT J
680 FOR K=1 TO NV
690 FOR H=K TO NV
700 POKE 1105+K*40+H,87
710 NEXT H,K
720 INPUT "{RVS} F PER FINE {OFF}";F$:
730 IF RIGHT$(F$,1)(">"F" THEN 720
740 FOR K=1 TO NV
750 FOR H=K TO NV
760 IN(K,H)=PEEK(1105+K*40+H)
770 NEXT H,K
800 GOTO 50
810 REM---ROUT.DI CARICAMENTO DIRETTO---:PRINT
820 PRINT:INPUT "LA FIGURA E' IN MEMORIA? (S/N)";MM$
821 IF LEFT$(MM$,1)="S" THEN 4700
825 PRINT:INPUT "QUANTI SONO I VERTICI ?";NV:PRINT:DIM CO(NV,3),IN(NV,NV),PO(NV,2)
830 FOR I=1 TO NV
835 PRINT "{RVS} COORDINATE DEL VERTICE {OFF}";I:
840 INPUT "COORDINATA X=";CO(I,1)
850 INPUT "COORDINATA Y=";CO(I,2)

```

essere riempita.

Vediamo un esempio.

Supponiamo di aver inserito le coordinate di un cubo (figura 6). Gli otto vertici sono numerati secondo un criterio di comodo (lo scegliete voi il criterio, basta che poi siate coerenti con questo).

Nella figura abbiamo riportato la numerazione in senso orario.

Consideriamo ora la relazione di collegamento tra i vertici, che è ovviamente simmetrica (se 1 è collegato con 4 vale anche il contrario).

Ciò permette di costruire la matrice di incidenza semplificata, poiché risulterà composta solo dalla sua parte superiore (triangolare alta). Si ha:

	1	2	3	4	5	6	7	8
1	★	o	★	★	o	o	o	
2		★	o	o	★	o	o	
i → 3			★	o	o	★	o	
4				o	o	o	★	
5					★	o	★	
6						★	o	
7							★	
8								★

ove ★ si legge "il vertice (i-riga) è collegato con il vertice (j-colonna)".

Avrete notato che, passando da un vertice al successivo, si prendono in considerazione solo i vertici di numerazione più alta. Ciò permette un'economia non

indifferente nell'acquisizione dei dati.

In questo modo si comunica quali vertici vanno collegati, e si possono così ottenere disegni poliedrici di qualsiasi natura.

### Descrizione del programma e suo uso

Il programma presentato è abbastanza lineare dal punto di vista della struttura e si divide essenzialmente nei seguenti segmenti.

- a) Linee 10-40 menu delle procedure.
- b) Linee 50-270 calcoli preliminari per il piano e traccia del riferimento.
- c) Linee 500-800 routine caricamento dati per figure poliedriche e matrice di incidenza.
- d) Linee 800-900 routine per poliedri con un numero di vertici maggiore di venti.
- e) Linee 900-1000 routine per caricamento dati delle superfici a punti.
- f) Linee 1600-1790 routine di tracciamento fibre x della superficie.
- g) Linee 1800-1900 routine di tracciamento fibre y.
- h) Linee 2300-2500 routine di tracciamento fibre r.
- i) Linee 3000-3100 routine di tracciamento figure poliedriche.
- l) Linee 3110-3200 scelta di stampa o di

cambiamento punto di vista.

m) Linee 4100-4300 routine di calcolo dei punti sul piano di proiezione.

n) Linee 4500-5500 routine di memorizzazione poliedri e superfici a punti e loro riletture.

Il programma è stato scritto con l'aiuto del Simons' BASIC, per utilizzare soprattutto la grafica.

In realtà le uniche istruzioni inserite sono:

- 1) Line x,y,u,v,1: tracciamento di linea da coordinate schermo (x,y) a coordinate schermo (u,v);
- 2) Hires h,k: richiamo dell'alta risoluzione (bit-map mode) con colori assegnati;
- 3) Cset n: ritorno alla pagina normale o da questa alla pagina grafica;
- 4) Pause t: ciclo di attesa di t secondi;
- 5) Text x,y,s\$,a,b,c: scrittura di testo in pagina grafica;
- 6) Copy: copia su stampante della pagina grafica;
- 7) On Error: gestione errori;

Poiché solo le prime tre sono essenziali, è possibile sostituirle con analoghe istruzioni di un qualunque programma in linguaggio macchina che abbiano la stessa funzione e che permetta l'uso dell'alta risoluzione.

Data la diffusione amatoriale di questi tipi di supporto alla programmazione, non dovrebbe essere difficile eventualmente eseguire le sostituzioni del caso.

#### Seguito listato 3D.

```

860 INPUT" COORDINATA Z=";CO(I,3)
865 NEXT I
866 PRINT"{CLR}";:PRINT"{RVS} CORREZIONI
      { 29 SPAZI}{OFF}"
867 PRINT:PRINT" VI SONO CORREZIONI DA FA
      RE ? (S/N)";
868 INPUT AAS
869 IF LEFT$(AAS,1)<>"S" THEN 878
870 PRINT:INPUT" {RVS} VERTICE ? {OFF}";V
      T:VT=INT(VT) :IF VT>NV THEN 870
871 INPUT" COORDINATA X=";CO(VT,1)
872 INPUT" COORDINATA Y=";CO(VT,2)
873 INPUT" COORDINATA Z=";CO(VT,3)
874 PRINT:PRINT" {RVS}{RED} ANCORA ? (S/N
      )";: INPUT AAS
875 IF LEFT$(AAS,1)="S" THEN 870
878 PRINT"{CLR}";:PRINT"{RVS}{BLU}
      { 9 SPAZI}MATRICE DI INCIDENZA
      { 11 SPAZI}"
880 FOR I=1 TO NV-1:FOR J=I+1 TO NV
885 PRINT"IL VERTICE{RVS}";I;"{OFF}E' LEG
  
```

```

      ATO CON{RVS} ";
886 INPUT J:IFJ<=I THEN 886
887 PRINTSPC(31)"{SU}{OFF}(S/N)";:INPUTR$
890 IF LEFT$(R$,1)="S" THEN IN(I,J)=81
895 NEXT J,I:GOTO 50
900 REM---SUPERFICI A PUNTI-----
905 PRINT"{CLR}";:PRINT"{RVS}{BLU}SUPERFI
      CI A PUNTI{ 23 SPAZI}"
906 PRINT:INPUT" {RVS}{RED}LA FIGURA E' IN
      MEMORIA? (S/N)";MM$:PRINT
907 IF LEFT$(MM$,1)="S" THEN 5000
910 PRINT:PRINT"DIMENSIONI DEL RETICOLO"
915 INPUT"DIMENSIONI X=";LX:
      INPUT"DIMENSIONI Y=";LY
920 DIM MT(2*LX,2*LY)
922 PRINT:PRINT"I VALORI SONO CARICATI DI
      RETTAMENTE? S/N"
923 GET GJ$:IF GJ$=""AND GJ$<>"S" ANDGJ$<
      >"N" THEN 923
924 IF GJ$="S" THEN 945
925 FOR X=0 TO 2*LX
930 FOR Y=0 TO 2*LY
  
```

Seguito listato 3D.

```

935 READ MT(X,Y)
940 NEXT Y,X : GOTO 950
945 FOR X=0 TO 2*LX
946 FOR Y=0 TO 2*LY
947 PRINT"QUOTA DEL PUNTO X={RVS}{BLK}";L
  X-X;" {OFF} Y={RVS}{BLK}";Y-LY;:INPUT
  MT(X,Y)
948 NEXT Y,X
950 DEFFNA(X)=MT(LX-X,Y+LY): BT=2*LX+1 :
  BU=2*LY+1
955 PRINT:PRINT"RICORDARE CHE LE
  { 2 SPAZI}DIMENSIONI INTRODOTTE"
956 PRINT"DEVONO COINCIDERE CON LE DIMENS
  IONI DEL"
957 PRINT"DOMINIO."
958 PRINT"IL NUMERO DELLE FIBRE DEVE INVE
  CE ESSE-"
959 PRINT"IN NUMERO ESATTAMENTE DOPPIO.
  { 3 SPAZI}":PAUSE 10:GOTO 50
960 PAUSE 3:CSET0:PRINT"{CLR}{GIU'}":IN
  PUT" {RVS}VUOI MEMORIZZARE LA FIGURA
  ? (S/N)";Y$
965 IF LEFT$(Y$,1)<>"S" THEN 990
970 PRINT:INPUT" {RVS}{BLU} NOME DELLA FI
  GURA ? {OFF}";NM$
980 GOSUB 5500
990 RETURN
1600 REM----TRACCIA DELLA FUNZIONE-----
1605 SP=40:SU=40:IF BT<>0 THEN SP=BT
  :FX=BT:FY=BU:SU=BU :GOTO 1700
1610 CSET 0:PRINT"{CLR}";:PRINT" {RVS} SU
  PERFICI TRIDIMENSIONALI{ 14 SPAZI}"
1620 PRINT:PRINT"PUOI SCEGLIERE UNA RAPPR
  ESENTAZIONE CON:"
1630 PRINT:PRINT" A) FIBRE X"
1635 PRINT:PRINT" B) FIBRE Y"
1640 PRINT:PRINT" C) FIBRE X,Y"
1645 PRINT:PRINT" D) FIBRE R{ 2 SPAZI}(PO
  LARI){ 3 SPAZI}"
1650 PRINT:INPUT" {RVS}QUALE SCEGLI ?";S$
1655 IF S$="D" THEN PRINT:INPUT"QUANTE FI
  BRE R?(20-30)";PL:GOTO 2300
1660 IF S$="A" THEN PRINT:INPUT" QUANTE F
  IBRE X?(20-40)";FX:GOTO 1700
1665 IF S$="B" THEN PRINT:INPUT" QUANTE F
  IBRE Y?(20-40)";FY: CSET2: GOTO 18
  00
1670 PRINT:INPUT" QUANTE FIBRE X?(20-40)"
  ;FX
1675 PRINT:INPUT" QUANTE FIBRE Y?(20-40)"
  ;FY
1700 CSET 2
1710 FOR X=A1 TO -A1 STEP -2*A1/FX
1714 Y=-B1
1716 Z=FNA(X)
1717 GOSUB 4100:MX=PX:MY=PY
1720 FOR Y=-B1 TO B1 STEP 2*B1/SU
1730 Z=FNA(X)
1740 GOSUB 4100
1743 IF 100+PY<0 THEN PX=MX+2:PY=-100
1744 IF 100+PY>200 THEN PY=100:PX=MX+2
1750 LINE 160+MX,100+MY,160+PX,100+PY,1
1760 MX=PX:MY=PY
1770 NEXT Y,X
1790 IF S$="A" THEN 3120
1800 REM---TRACCIAMENTO SECONDARIO-----
1810 FOR Y=-B1 TO B1 STEP 2*B1/FY
1814 X=A1
1815 Z=FNA(X)
1816 GOSUB 4100:MX=PX:MY=PY
1820 FOR X=A1 TO -A1 STEP -2*A1/SP
1830 Z=FNA(X)
1840 GOSUB 4100
1842 IF 100+PY<0 THEN PY=-100:PX=MX+2
1844 IF 100+PY>200 THEN PY=100 :PX=MX+2
1846 IF 100+MY<0 THEN MY=-100
1848 IF 100+MY>200 THEN MY=100
1850 LINE 160+MX,100+MY,160+PX,100+PY,1
1860 MX=PX:MY=PY
1870 NEXT X,Y
1880 IF BT<>0 THEN GOSUB 960 :CSET2
1900 GOTO 3120
2300 REM-TRACCIATURA COORDINATE POLARI--
2305 CSET2
2310 FOR R=A1 TO 0 STEP -A1/PL
2314 X=R:Y=0
2315 Z=FNA(X)
2316 GOSUB 4100:MX=PX:MY=PY
2320 FOR O=0 TO 2*+.15 STEP 2*/40
2325 X=R*COS(O):Y=R*SIN(O)
2330 Z=FNA(X)
2340 GOSUB 4100
2342 IF 100-PY>200THEN PY=-100
2344 IF 100+PY<0 THEN PY=100
2346 IF 100+MY<0 THEN MY=-100
2348 IF 100-MY>200 THEN MY=100
2350 LINE 160+MX,100+MY,160+PX,100+PY,1
2360 MX=PX:MY=PY
2370 NEXT O,R
2400 PAUSE 1
2410 FOR O=0 TO 2* STEP 2*/30
2414 X=A1*COS(O):Y=A1*SIN(O)
2415 Z=FNA(X)
2416 GOSUB 4100:MX=PX:MY=PY
2420 FOR R=A1 TO 0 STEP -A1/21
2425 X=R*COS(O):Y=R*SIN(O)
2430 Z=FNA(X)
2440 GOSUB 4100
2442 IF 100+PY>200THEN PY=100
2444 IF 100-PY<0 THEN PY=-100
2446 IF 100+MY<0 THEN MY=-100
2448 IF 100+MY>200 THEN MY=100
2450 LINE 160+MX,100+MY,160+PX,100+PY,1
2460 MX=PX:MY=PY
2470 NEXT R,O
2500 GOTO 3120
3000 REM----TRACCIAMENTO POLIEDRI-----
3005 CSET 2
3009 FOR I=1 TO NV

```

## COMMODORE 64

Per quanto riguarda il suo utilizzo, non dovrebbero esserci problemi data la sua struttura colloquiale.

Dopo che è stato mandato in esecuzione, vi apparirà la richiesta di specificare la procedura scelta (funzione con legge, funzione a punti, poliedro).

Nel primo caso vi viene richiesto l'inserimento della legge funzionale, e in seguito i parametri per la proiezione prospettica: tre coordinate per il punto di vista (x0,y0,z0), la distanza del piano di proiezione dal centro degli assi (ds), il rettangolo di base in cui è definita la funzione, per il quale è sufficiente digitare i valori positivi - se introducete i valori R1, B1 si intenderà che la funzione risulta definita nel rettangolo del piano cartesiano  $R=(-A1 < x < A1 ; -B1 < y < B1)$  - e infine la quota massima.

Quando inserite questi parametri, alla sinistra delle frasi di richiesta appare il

valore utilizzato in precedenza (la prima volta tale valore sarà ovviamente nullo). Nel caso di funzioni discontinue, la figura si interrompe solitamente sul bordo della pagina grafica.

Dopo di ciò apparirà il sistema di riferimento in vista prospettica, dandovi modo di valutare se la scelta dei parametri è stata conveniente.

In seguito vi verrà domandato quale tipo di rappresentazione desiderate (fibre x, fibre y, fibre r) e il numero delle fibre che volete siano tracciate.

Tenete presente a questo proposito che un numero eccessivo di fibre confonderà il disegno invece di renderlo più evidente (il numero consigliato è tra 20 e 30).

Inizierà poi il tracciamento vero e proprio (può durare qualche minuto), ed infine potrete stamparlo o rivederlo da un nuovo punto di vista, modificando

eventualmente le fibre di rappresentazione (in figura 7 a,b,c,d la funzione  $z = 3\sin(x/2 + y/3)$  vista con diverse modalità).

Se avrete invece scelto una superficie a punti, vi verrà richiesto se l'inserimento è eseguibile direttamente o tramite i valori scritti nell'istruzione Data.

Nel primo caso dopo aver specificato le dimensioni del reticolo, potete inserire i valori, nel secondo dovrete inizialmente modificare il programma con l'aggiunta dei Data necessari.

Se volete memorizzare i valori introdotti in modo permanente, potrete utilizzare una routine che li immagazzina in un file su disco.

In questo modo potrete sempre richiamare successivamente gli stessi dati e rivedere la superficie relativa.

Completata l'operazione di caricamento, dovrete nuovamente scegliere tra

## Seguito listato 3D.

```

3010 X=CO(I,1):Y=CO(I,2):Z=CO(I,3):
      GOSUB 4100:PO(I,1)=PX:PO(I,2)=PY
3011 NEXT I
3015 FOR T=1 TO NV
3020 FOR U=T TO NV
3030 IF IN(T,U)<>81 THEN 3100
3040 LINE 160+PO(T,1),100+PO(T,2),160+PO
      (U,1),100+PO(U,2),1
3100 NEXT U,T
3110 PAUSE 5:CSET0
3115 PRINT"{CLR}{ 2 GIU' }{DES}":INPUT" VU
      OI MEMORIZZARE LA FIGURA ? (S/N)";Y$
3116 IF LEFT$(Y$,1)<>"S" THEN CSET2:GOTO
      3125
3117 INPUT" {RVS}{BLU} NOME DELLA FIGURA
      ? {OFF}";NM$
3118 GOSUB 4500:CSET 2
3120 REM----PROCEDURA DI STAMPA-----
3125 AV$="F1 COPIA{ 2 SPAZI}*{ 2 SPAZI}F3
      NUOVO PV. "
3130 FOR L=1 TO 25:TEXT 311,8*(L-1),"{B}
      "+MID$(AV$,L,1),1,1,6:NEXT L
3140 GET C$:IF C$="" THEN 3130
3150 IF C$<>CHR$(133) THEN 3160
3155 FOR L=1 TO 25:TEXT 311,8*(L-1),"{B}
      "+MID$(AV$,L,1),0,1,6:NEXT L:COPY
3160 IF C$=CHR$(134) THEN 3200
3165 IF C$=CHR$(135) THEN END
3170 GOTO 3130
3200 CSET 0:PRINT"{CLR}";:GOTO 50

```

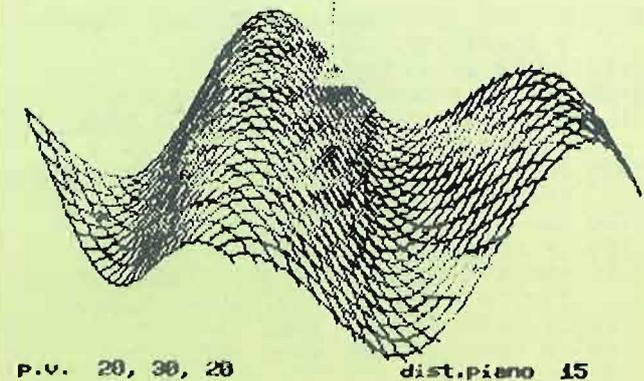
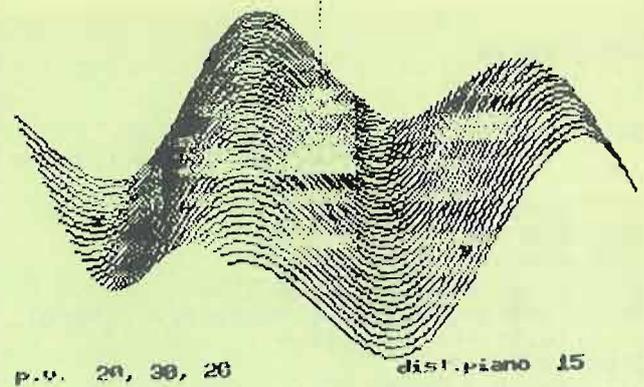
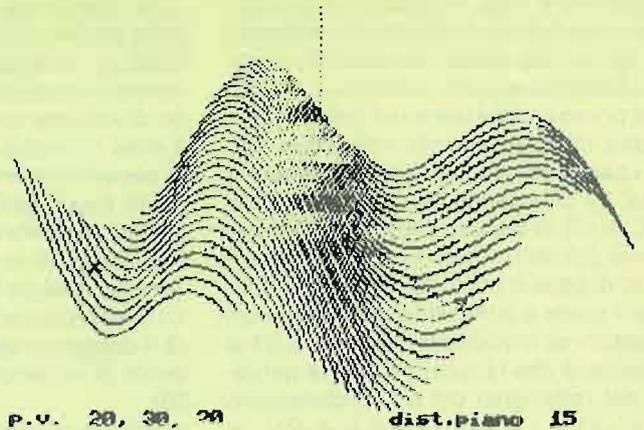
```

4100 REM---CALCOLO DEI PUNTI SUL-----
      PIANO DI PROIEZIONE
4105 XP=X0+(DS*D0-D0*D0)/(X0*X+Y0*Y+Z0*Z-
      D0*D0)*(X-X0)
4110 YP=Y0+(DS*D0-D0*D0)/(X0*X+Y0*Y+Z0*Z-
      D0*D0)*(Y-Y0)
4120 ZP=Z0+(DS*D0-D0*D0)/(X0*X+Y0*Y+Z0*Z-
      D0*D0)*(Z-Z0)
4200 TX=(X0*YP-Y0*XP)/P1
4210 TY=(Z0*DS-ZP*D0)/P1
4280 PX=TX*UX : PY=TY*UY
4300 RETURN
4500 REM-----REGISTRAZIONE FIGURE-----
4505 OPEN 5,8,5,"0:"+NM$+",S,W"
4510 PRINT#5,PR
4515 PRINT#5,NV:PRINT#5,X0:PRINT#5,Y0:
      PRINT#5,Z0:PRINT#5,DS
4516 PRINT#5,A1:PRINT#5,B1
4520 FOR J=1 TO NV
4530 FOR I=1 TO 3
4540 PRINT#5,CO(J,I)
4550 NEXT I,J
4560 FOR J=1 TO NV
4570 FOR I=J TO NV
4580 PRINT#5,IN(J,I)
4590 NEXT I,J
4600 CLOSE 5:RETURN
4700 REM-----LETTURA DELLE FIGURE-----
4705 PRINT:INPUT"{RVS} QUALE E' IL NOME ?
      ";NM$
4710 OPEN 5,8,5,"0:"+NM$+",S,R"
4715 INPUT#5,PR
4720 INPUT#5,NV:INPUT#5,X0:INPUT#5,Y0:
      INPUT#5,Z0:INPUT#5,DS
4721 INPUT#5,A1:INPUT#5,B1

```

Seguito listato 3D.

```
4725 FOR J=1 TO NV
4730 FOR I=1 TO 3
4740 INPUT#5,CO(J,I)
4750 NEXT I,J
4760 FOR J=1 TO NV
4770 FOR I=J TO NV
4780 INPUT#5,IN(J,I)
4790 NEXT I,J
4800 CLOSE 5 :GOTO 98
5000 REM-----LETTURA DELLE FIGURE-----
5005 PRINT:INPUT"{RVS} QUALE E' IL NOME ?
";NM$
5010 OPEN 5,8,5,"0:"+NM$+",S,R"
5015 INPUT#5,PR
5020 INPUT#5,LX:INPUT#5,LY:
      INPUT#5,X0:INPUT#5,Y0
5021 INPUT#5,Z0:INPUT#5,DS
5024 DIM MT(2*LX,2*LY)
5025 FOR J=0 TO 2*LX
5030 FOR I=0 TO 2*LY
5040 INPUT#5,MT(J,I)
5050 NEXT I,J:CLOSE 5
5100 DEFFNA(X)=MT(LX-X,Y+LY)
5200 A1=LX:B1=LY:BT=2*LX+1:BU=2*LY+1
      :GOTO 98
5500 REM-----REGISTRAZIONE FIGURE-----
5510 OPEN 5,8,5,"0:"+NM$+",S,W"
5515 PRINT#5,PR
5520 PRINT#5,LX:PRINT#5,LY:
      PRINT#5,X0:PRINT#5,Y0
5521 PRINT#5,Z0:PRINT#5,DS
5525 FOR J=0 TO 2*LX
5530 FOR I=0 TO 2*LY
5540 PRINT#5,MT(J,I)
5550 NEXT I,J
5600 CLOSE 5 :RETURN
6000 REM -----MESSAGGIO D'ERRORE-----
6010 CSET0:PRINT"{CLR}{ 2 GIU' }";
6020 PRINT" {RVS} SIAMO NELLA CONDIZIONE
DI ERRORE";ERRN
6030 PRINT:PRINT"1) O IL PUNTO DI VISTA T
ROPP0 VICINO "
6031 PRINT"2) O LA DISTANZA DEL PIANO E'
ERRATA "
6032 PRINT"3) O VI E' UN ERRORE NELLA DEF
INIZIONE"
6033 PRINT"{ 3 SPAZI}DELLA FUNZIONE
{ 21 SPAZI}"
6050 PAUSE 8:NO ERROR:GOTO50
```



rappresentazione in fibre x,y ed r, ed infine definire il numero delle fibre che in questo caso deve essere tassativamente uguale al numero delle partizioni usate, altrimenti il programma non può procedere.

Questo punto è particolarmente delicato, ed è preferibile fare un esempio.

Se volete rappresentare un reticolo che si posizioni tra i valori  $-2 < x < 2$  e  $-3 < y < 3$  di tipo 4 x 6, dovrete inserire alla richiesta delle dimensioni i dati 2 e 3.

Il calcolo delle fibre necessarie è fatto automaticamente e risulteranno in questo caso 5 e 7 rispettivamente.

I valori 2 e 3 saranno anche inseriti quando vi si chiederà il rettangolo di definizione del disegno.

I risultati migliori si ottengono con dati di valore prossimi ad uno, pertanto volendo usare dati di valore alto conviene normalizzarli, ad esempio 123,89,151 andrebbero inseriti come 1.23,0.89,1.51. Il seguito si presenta poi come nel caso precedente.

Se infine avete scelto di vedere un poliedro, potrete procedere dichiarando se si tratta di figura con più di venti vertici o no.

Nel primo caso la procedura di inserimento diventa laboriosa, poiché dovrete inserire tutte le coordinate (che sono già  $> 60$ ) e la matrice di incidenza.

Quest'ultima viene inserita nel seguente modo. Consideriamo di voler caricare

Figura 8a,b,c - Alcuni disegni ottenuti con il programma.

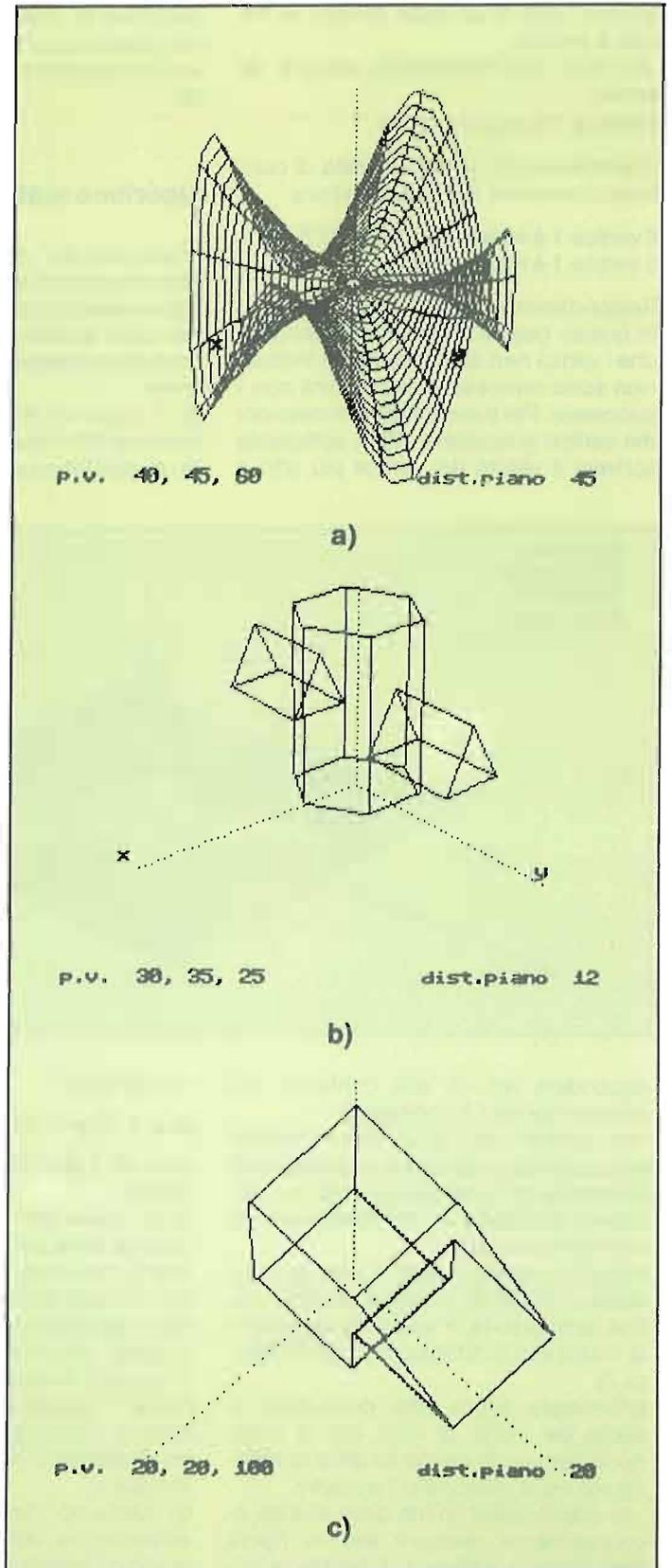
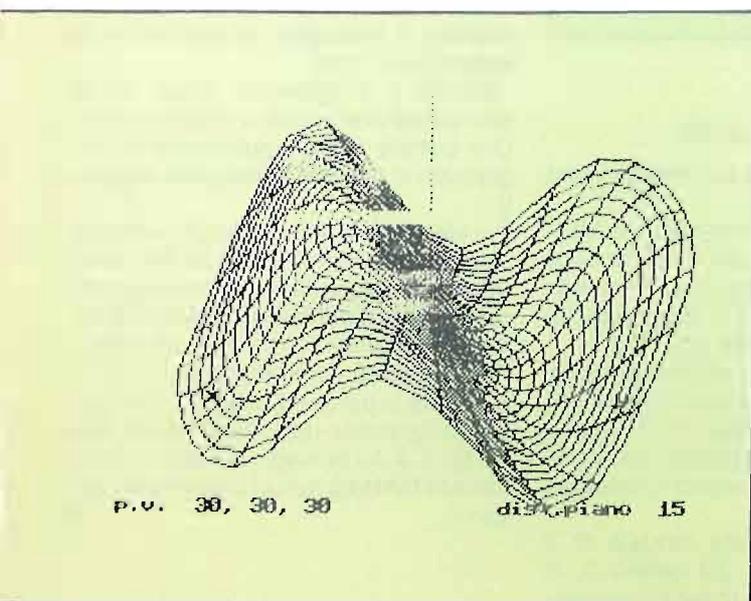


Figura 7 - Rappresentazione d'una funzione con diverse modalità.



ancora i dati di un cubo (anche se ha solo 8 vertici).

All'inizio dell'inserimento appare la scritta:

Il vertice 1 è legato con ?

risponderemo 2, e alla richiesta di conferma scriviamo S. Apparirà allora:

Il vertice 1 è legato con 2 (S/N)? S

Il vertice 1 è legato con ?

Risponderemo 4, ecc.

In questo caso il programma intenderà che i vertici non espressamente indicati non sono connessi e proseguirà con i successivi. Per passare alle connessioni del vertice successivo (2), è sufficiente scrivere il valore del vertice più alto e

struzione di uno schedario di disegni che possono poi essere rivisti senza essere nuovamente (e noiosamente) inseriti.

### Algoritmo matematico

Lasciamo per ultimo (e può quindi essere omesso dal lettore non interessato) il chiarimento sulle formule utilizzate.

Nel caso abbiamo provveduto semplicemente ad eseguire i seguenti procedimenti:

a) il punto di vista  $P=(x_0,y_0,z_0)$  e la distanza DS ci consentono di individuare i coseni direttori del piano di proiezio-

zione sul piano prospettico [linee 4100-4300].

Così facendo otteniamo relazioni tra punti giacenti sul medesimo piano (quello prospettico appunto), realizzando così un'immagine virtuale piana per proiezione dell'immagine vera tramite i raggi "di vista" uscenti dal punto P.

La descrizione è sommaria, ma dovrebbe indicare con sufficiente chiarezza i vari algoritmi usati.

### Note finali

Qualora la velocità di tracciamento delle superfici vi appaia troppo lenta, potrete aumentarla cambiando i primi due parametri della linea 1605 (ma saranno meno precise).

Se invece volete intervenire sulle dimensioni del disegno, provate a cambiare i valori UX e UY della linea 98 (che nel programma sono inutilizzati perché già definiti alla linea 53).

Se per una qualsiasi ragione il programma si fermasse per una dichiarazione di "Illegal Quantity", il che accade solitamente per un'errata impostazione dei parametri, dovete ripartire con l'istruzione diretta:

GOTO 50

In questo modo non perderete i dati inseriti, e ripartirete dal punto in cui dovette introdurre le coordinate del punto di vista e altri parametri.

In questi casi è comunque consigliabile allontanare il piano di proiezione o aumentare il rettangolo di definizione (o entrambe le cose).

Benché il programma utilizzi sia la stampante che l'unità a dischi, è ovvio che potrete usare ugualmente il programma in mancanza di questi supporti.

Per eliminare la stampa è sufficiente sostituire le istruzioni Copy con dei cicli di attesa, per usare l'unità a nastri occorre ovviamente modificare le routine di lettura e scrittura in modo conseguente.

Potete vedere comunque (figure 8a, b, c, d) che si possono ottenere con questo programma disegni ed effetti interessanti; a voi immaginare gli usi che la limitata fantasia non ci permette di suggerire. ■

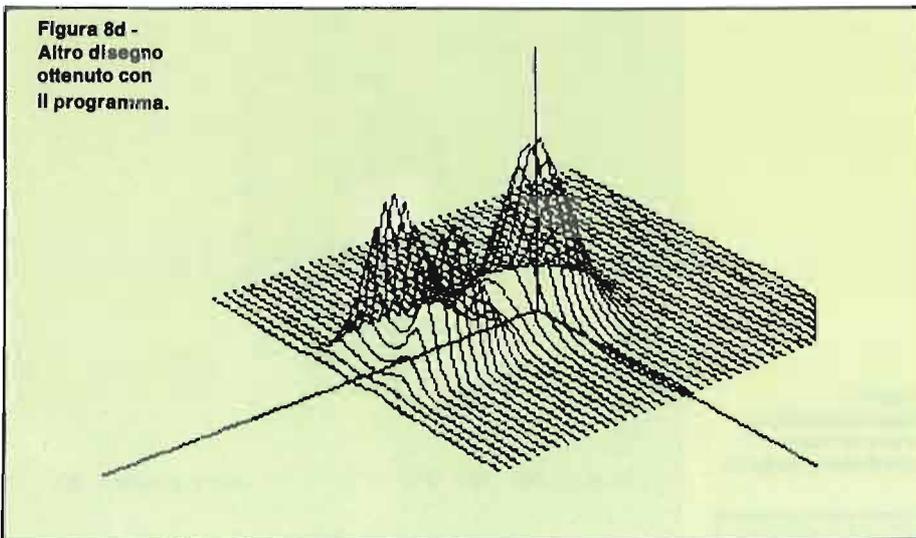


Figura 8d - Altro disegno ottenuto con il programma.

rispondere poi -N- alla conferma (se ovviamente non è connesso).

Nel secondo caso, dopo aver introdotto le coordinate, vi apparirà una matrice di incidenza già predisposta, su cui voi potrete posizionarvi utilizzando i tasti di movimento cursore.

Inserite in ogni punto di incidenza il carattere o (Shift Q), procedendo sino alla fine della tabella, e una volta terminata la matrice di incidenza premete F (figura 7).

Ritournerete allora alla procedura di scelta del punto di vista,...ecc. e infine dopo l'apparizione del sistema di riferimento verrà disegnato il poliedro.

Anche in quest'ultimo caso potrete eventualmente memorizzare la figura così come la vedete; ciò facilita la co-

ne che sarà:

$$x_0.x + y_0.y + z_0.z = d_0.DS$$

dove  $d_0$  è la distanza euclidea di P dal centro;

b) gli stessi dati ci permettono di tracciare la retta per P e per il centro degli assi C, che incrociando il piano determina l'immagine virtuale C, del centro sul piano prospettico [linee 100-150];

c) preso ora un punto reale della figura  $V=(x,y,z)$ , costruiamo la retta congiungente P (punto di vista) con V (punto reale) e intersecando la retta con il piano prospettico ne troviamo l'immagine virtuale V;

d) riferiamo l'immagine virtuale di V all'immagine virtuale del centro C, in modo da stabilire la loro reciproca posi-

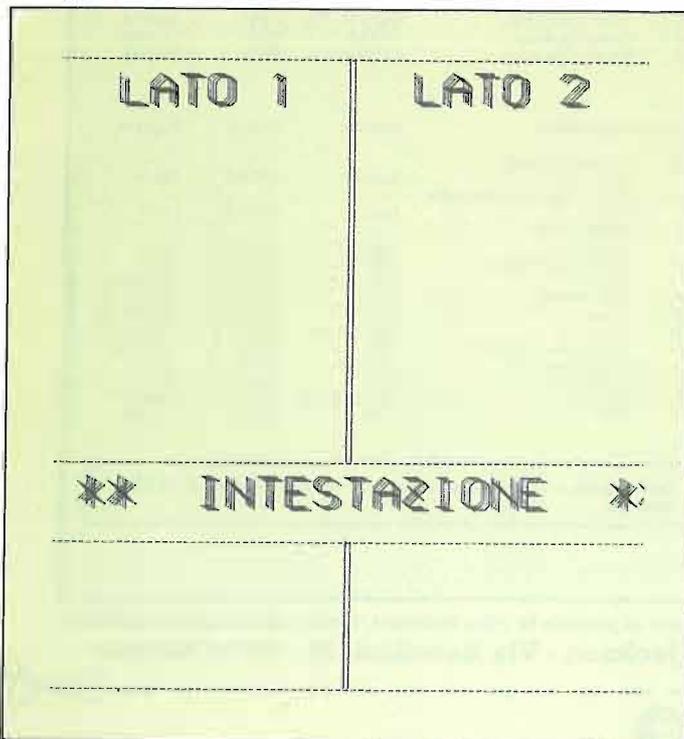
**C**erchiamo di etichettare le nostre cassette in un modo migliore, buttiamo le biro ed usiamo il versatile plotter del 731. Carichiamo il BASIC, digitiamo il programma, inseriamo i pennini colorati e la carta nel plotter, battiamo il Run e siamo pronti per etichettare una cassetta.

Il plotter incomincerà a delineare i contorni dell'etichetta e una volta terminati, scriverà Lato 1 e Lato 2 quindi vi chiederà che stringa scrivere, a questo punto tocca a voi digitare ciò che volete scrivere sulla etichetta (esempio "000 Intest. Cassetta2" se a 000 vi è questo programma).

# Intestazione cassette

Basta alle etichette delle cassette scritte male e completamente incomprensibili!

di Paolo Rossi



Listato 1 - Il programma Intestazione cassette.

```
10 MODE GR:LINE*3:PCOLOR0
20 LINE480,0:MOVE0,-330:LINE480,-330:MOVE0,-395:LINE480,-395:MOVE0,-515:LINE480,-515:PHOME:MOVE238,0:LINE*1,238,-330:MOVE242,-330:LINE242,0:MOVE238,-395:LINE238,-515:MOVE242,-515:LINE242,-395
30 FOR I=0TO3
40 MOVE50+I*2,-40-I:PCOLORI:GPRINT[3,0],
```

Figura 1 - Esempio di etichetta.

Le stringhe che voi immetterete verranno scritte in modo 80 colonne (cioè col carattere più piccolo).

Le stringhe scritte corrispondono in sequenza a quelle del lato 1 e quindi, una volta digitato il carattere "?" per fare il salto pagina, quelle del lato 2.

Una volta finito di scrivere le stringhe del lato 2 digitando ancora il carattere "?" il plotter si sposterà nella parte di etichetta che si verrà a trovare nel bordo della cassetta e vi verrà chiesto che intestazione scrivere (N.B - Max 19 caratteri), quindi, una volta inserita e scritta l'intestazione, la vostra etichetta è pronta.

```
"LATO 1":NEXT I
50 FOR I=0TO3
60 MOVE290+I*2,-40-I:PCOLORI:GPRINT[3,0], "LATO 2":NEXT I
70 MOVE 5,-60:A=5:B=-60
80 PCOLOR0
90 INPUT"STRINGA ";S$:IFS$="?" THEN 130
100 GPRINT[0,0],S$
110 B=B-10:MOVEA,B
120 GOTO 90
130 D=D+1:IF D>1THEN 160
140 MOVE 250,-60:A=250:B=-60
150 GOTO 90
160 MOVE 20,-380
170 INPUT"INTESTAZIONE ";I$
180 FOR I=0TO3
190 MOVE 20+I*2,-370-I:PCOLORI:GPRINT[3,0],I$:NEXT I
200 PCOLOR0:MODE TN
```

# SERVIZIO SOFTWARE

# Bit

Bit e Personal Software propongono ai propri lettori i dischi o le cassette dei programmi pubblicati. Uso e descrizione dei programmi si trovano sui rispettivi numeri delle riviste.



# PERSONAL SOFTWARE

Bit n°	Programma	Sistema	Codice	Supporto	Bit n°	Programma	Sistema	Codice	Supporto
38	Gioco della scimmia	VIC 20	VI391	Cassetta	63	Operazioni in pagina grafica	Apple II	AP632	Disco
38	Spaccamattoni	Apple II	AP382	Disco	63	Grafici a barre verticali affiancate	C 16	G1631	Cassetta
42	Apple-Chef	Apple II	AP422	Disco	63	Contabilità casalinga	C 64	C6632	Disco
42	Provariflessi	VIC 20	VI421	Cassetta	63	Disegnare sullo schermo con un MSX	MSX	MS631	Cassetta
45	Tiny FORTH	Apple II	AP452	Disco	63	Over BASIC	ZX Spectrum	SP631	Cassetta
45	Alli Babà	ZX Spectrum	SP451	Cassetta	63	Le otto Regine	TI99/4A	TI631	Cassetta
46	Forzaquattro	Apple II	AP462	Disco	63	Egitto 2000	VIC 20	VI631	Cassetta
48	Simulavolo	ZX Spectrum	SP481	Cassetta	64	Roller	Apple II	AP642	Disco
48	Memory Alfa IV	C 64	C6481	Cassetta	64	Come sistemiamo i mobili?	C 16	C1641	Cassetta
49	Scorpion	Apple II	AP492	Disco	64	Titolazioni simulate	C 64	C6642	Disco
50	Fp-Plot	Apple II	AP502	Disco	64	Defektor	MSX	MS641	Cassetta
50	Prima e Terza	ZX Spectrum	SP501	Cassetta	64	Color Compositor	Sharp MZ-700	SH641	Cassetta
51	Magicalatog	Apple II	AP512	Disco	64	Computer music	TI99/4A	TI641	Cassetta
53	Partita a golf	VIC 20	VI531	Cassetta	65	Integer Compiler	ZX Spectrum	SP651	Cassetta
53	Analisi numerica	C 64	C6531	Cassetta					
53	PL/Bit: il compilatore	Apple II	AP532	Disco					
54	Costellations	Apple II	AP542	Disco					
54	Come polarizzare i transistor col C 64	C 64	C6541	Cassetta					
58	Memory Omega I	C 64	C6582A	Disco					
58	Copy disk per C 64	C 64	C6582B	Disco					
59	Checksum 64	C 64	C6592	Disco					
59	Checksum 64	C 64	C6591	Cassetta					
59	Data-Bank	ZX Spectrum	SP591	Cassetta					
60	Life HGR	Apple II	AP602	Disco					
60	Tutti pittori	C 64	C6601	Disco					
60	Difesa della Terra	C 16	C1601	Cassetta					
60	Lost on the pack	Sega	SE601	Cassetta					
60	Setdata e Wordproc	ZX Spectrum	SP601	Cassetta					
60	Il Barone Rosso	TI99/4A	TI601	Cassetta					
60	Word processor	C 64	C6602	Disco					
60	Othello	VIC 20	VI601	Cassetta					
61	Disegno di mobili componibili	C 16	C1611	Cassetta					
61	Esperimento di Millikan	C 64	C6611	Cassetta					
61	Esperimento di Millikan	C 64	C6612	Disco					
61	Disegnare in alta risoluzione	ZX Spectrum	SP611	Cassetta					
61	Printing music	TI99/4A	TI611	Cassetta					
61	Musica facile	Sega	SE611	Cassetta					
61	VIC-Calc	VIC 20	VI611	Cassetta					
62	Gestione conto corrente	C 64	C6622	Disco					
62	Gioco della tombola	TI99/4A	TI621	Cassetta					
62	Aspc: lo Spectrum contro la carie	ZX Spectrum	SP621	Cassetta					

P.S. n°	Programma	Sistema	Codice	Supporto
3	La carta del cielo	Apple II	AP032A	Disco
4	Collisione	Apple II	AP042A	Disco
4	Interi in precisione multipla	Apple II	AP042A	Disco
5	Pretty printer	Apple II	AP052A	Disco
7	Shape table	Apple II	AP072A	Disco
14	Data base modulare	C 64	C6141A	Cassetta
14	Tool-Kit	C 64	C6141A	Cassetta
19	Type-Writer	VIC 20	VI192A	Disco
20	Scopa	C 64	C6201A	Cassetta
30	Geo-Race	ZX Spectrum	SP301A	Cassetta
31	Progetto aeromodello	MSX	MS311	Cassetta
31	Scopa a tre carte	Apple II	AP312	Disco
31	Grand Prix	C 64	C6312	Disco
31	Sprite	ZX Spectrum	SP311	Cassetta
31	Box	C 16	C1311	Cassetta

Tutti i dischi e le cassette dei programmi sono in vendita a L. 15.000 ciascuno.

Per richiedere i programmi in contrassegno, pagando direttamente al postino la cifra indicata, inviare il seguente tagliando  
**Spedire in busta chiusa a Gruppo Editoriale Jackson - Via Rosellini, 12 - 20124 Milano**

Inviatemi i seguenti nastri e/o dischi con i programmi pubblicati su Bit - Personal Software

Cod.  a L.  **15000**

Cod.  a L.

Cod.  a L.

Cod.  a L.

**+ SPESE POSTALI**  
(contributo fisso)

**3000**

**TOTALE L.**

che pagherò al postino alla consegna del pacco



**GRUPPO EDITORIALE JACKSON**

Cognome.....

Nome.....

Indirizzo.....

CAP.....

Città.....

Firma.....

Intestazione cassette

SHARP MZ-731

Ora non vi rimane che tagliarla lungo le linee tratteggiate esterne e lateralmente per poterla inserire nel porta cassetta.

Vi consigliamo di incollare questa etichetta su un pezzo di cartoncino così da renderla più resistente all'uso.

### REMARKS

**10** - Impostazione modo grafico e scelta della linea tratteggiata e della penna nera.

**20** - Disegno contorni dell'etichetta.

**30-60** - Stampa Lato 1 e 2.

**70-80** - Posizionamento della penna e impostazione iniziale delle variabili A e B (coordinate in cui la penna deve andare).

**90** - Richiesta input della stringa e controllo se è stato digitato il carattere "?".

**100** - Stampa della stringa.

**110** - Decremento della variabile B e movimento della penna alla nuova posizione A, B.

**120** - Rimanda alla richiesta di stringa.

**130** - Incremento della variabile D e controllo se è già stato digitato il carattere "?" (nel qual caso D=2).

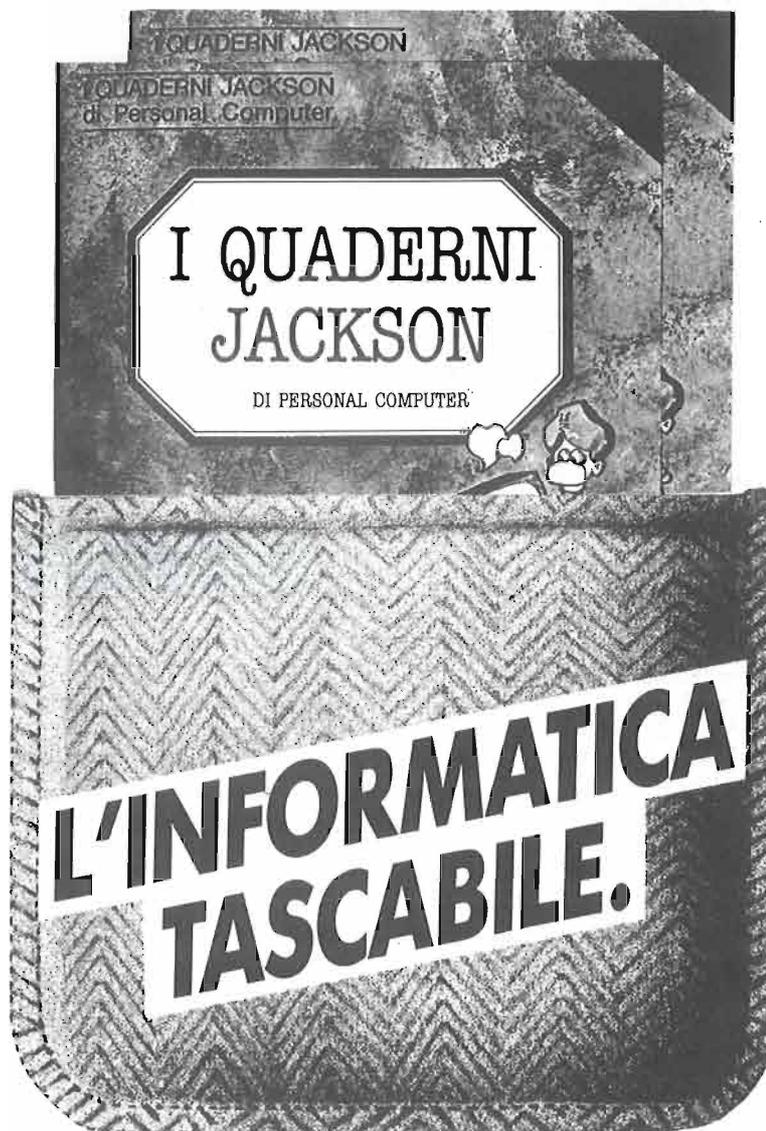
**140** - Salto pagina.

**150** - Ritorna alla richiesta di stringa.

**170** - Richiesta input intestazione.

**180-190** - Stampa intestazione.

**200** - Impostazione iniziale del plotter penna colore nero e modo testo. ■



Quaderni Jackson:  
l'informatica a tutti i livelli, in  
una collana aperta, pratica,  
essenziale, aggiornata.

Tutto quello che è  
importante sapere sui  
computer, la programmazione,  
i linguaggi, il software, le  
applicazioni e i nuovi sviluppi  
dell'informatica.

Ogni mese, 2 volumi.

#### Volumi già pubblicati:

Gianni Giaccaglini

"Vivere col Personal Computer"

Paolo Bozzola

"Dentro e fuori la scatola"

Enrico Odetti

"Ed è subito BASIC Vol. I"

"Ed è subito BASIC Vol. II"

Paolo Capobussi

e Marco Giacobazzi

"A ciascuno il suo Personal"

Fulvio Francesconi

e Fernando Paterlini

"To do or not to do"

Gianni Giaccaglini

"Strutturare il software"

Enrico Odetti

"Dizionario informatico"

In edicola,  
a sole lire 6.000.



GRUPPO  
EDITORIALE  
JACKSON

SAN FRANCISCO-LONDRA-MILANO

**L'** articolo descrive un metodo statistico per ottenere prosa casuale partendo da un testo sensato, che viene analizzato e successivamente elaborato. Pur essendo casuale, ciò che si ottiene non è privo d'interesse, perché il grado di casualità (e di sensatezza) può essere agevolmente variato.

"Se un esercito di scimmie pestasse su delle macchine per scrivere, potrebbe scrivere tutti i libri del British Museum". Così si esprimeva, nel 1927, sir Arthur Eddington, volendo indicare un evento che in linea di principio potrebbe verificarsi, ma che in realtà non accadrà mai. Ma cosa succederebbe se davvero un esercito di scimmie si mettesse a scrivere a macchina, magari usando mani e piedi? Per fortuna, a noi non occorre saccheggiare lo zoo della città per compiere l'esperimento. Possiamo programmare il nostro personal in modo da simulare la situazione. Ma cosa ci può essere di interessante, direte voi? Beh, qualcosa c'è. Soprattutto se si introducono alcune varianti al comportamento delle scimmie.

Qualche tempo fa, in una rubrica dell'edizione italiana di *Scientific American*, è apparso un articolo su questo argomento a firma di Brian Hayes. Il tema ci è piaciuto, e così abbiamo voluto provare anche noi a simulare delle scimmie di Eddington sull'Apple.

Il succo della cosa sta nel rendere più o meno intelligenti le nostre cavie, rendendo in tal modo più o meno casuale il testo che otteniamo. Girando, per così dire, la manopola che regola il grado di casualità: si ottengono dei testi insensati, ma dai quali traspare qualcosa. State a vedere. Inizieremo col fissare un alfabeto di simboli, diciamo tutte le lettere ed i segni di interpunzione.

Costruiamo una macchina per scrivere dotata di questi tasti e diamola in mano alla nostra bestiola. Dopo qualche momento di esitazione, l'astuto animale comincerà a pigiare sui tasti. Quello che apparirà sul foglio lo si può vedere nella figura 1. È un gran pasticcio. Le lettere ed i segni si susseguono in modo casuale, le parole sono lunghissime e non sembrano appartenere ad alcun idioma umano (c'era da aspettarselo, visto chi c'è alla tastiera). Fin qui niente di speciale: la scimmia si comporta come in-

# Elaborazioni a quattro mani

Come l'Apple può diventare poeta

di Marco Morocutti

tendeva Eddington. Chiameremo il prodotto della nostra elaborazione "testo di ordine zero".

Ed ecco l'idea: l'ha avuta William Ralph Bennett Jr., della Yale University, che ha studiato i processi per generare testi casuali basandosi su metodi statistici. Si mette davanti alla scimmia (simulata, mi raccomando) uno scritto sensato, e le si insegna a premere i tasti non a casaccio, ma secondo la frequenza con la quale i vari caratteri ap-

paiono nel testo.

Per rimanere fra noi abbiamo scelto, come testo originale su cui basare le elaborazioni, l'editoriale di Gianni Giaccagliani apparso sul numero 52 di *Bit*. Si ottiene una elaborazione detta di ordine uno: se ne vede un esempio nella figura 2. Povero Giaccagliani, lo riconoscereste? No di certo, ma almeno le parole sono diventate più corte. Adesso la frequenza di apparizione di ogni carattere non è più la stessa, ma rispecchia

Figura 1 - Esempio di testo casuale di ordine zero.

```

)R; ;MZD?BFMN(H, YCSFLYRVUQDXNZAUHM(KXT( U) (XQJFRCD. DDW; US: XXZ, YCBXCIIIVSL; ;P (KKE.
EU!ONPMC. ;JSQWOM!B: (U(;H?;!;DNTCFXSDNUQT. KU: TI?L!IT (NX?V)
?TBQTYM?FW (FHQ, ;, ;, U, ITHOAZWHP (GD)EM, ;JXWTWCD. ;RU;GR;H; ;EWASRF;LVK; (;.T. FHF, QL?E (G, V) DCFE
CG; !AV?

```

Figura 2 - Editoriale di Gianni Giaccagliani straziato da una scimmia di Eddington di ordine uno.

```

R RRRLAESE CR RLSE U UOVQCOO SO ARSOAIE, LONI A*ANILDEADEEEA NUIION.P,NER
IPHSOSNSDISAPAECEVMIATRADIANGEBR SASINE GD MAIAHAD* SNEUNGSREEA(HNRULIDM EDIRSERD
AIIARR T TPELAEA PIAS.IHEODCOO /AL, EOE TENPOLLTAOGEACDI ARTCND I *EAC M
OOVATNLDMPFAMMMVUD EOEADAARDCCSCP MDIOTMN ,OE UNRORRLF N RCC, UDD TRG PNLK O AMVNACEMN

```

Figura 3 - Elaborazione di ordine due sullo stesso testo.

```

DO SE CAND II-PRSUNTAZI PE SSIVE DIATTREDELTO SPE NTESSSSON NTIR IE*OLE CHITE EDOLE
NITE DE UCHE PRIDI. DI, DIOMBISTE ITO UTE SOBALATO QUD PAMO, SAUOSUEND; PUNZIARE LON
EL EMPRO. FISODO ATESTE NA A*ORONTE HE SSA ARSIFA DO DO RE MA SMAZI. DIAMOPPATOKE,
MAZANICO OUA, NTEMOGIN ATE LUTUND ON* CE'), UORO.. ALERAIMPOSE.

```

Figura 4 - Elaborazione di ordine tre. Comincia ad apparire qualcosa della lingua italiana.

```

PERSOPPOSCPOCONO AL STI, IN QUEL VOGRA DIFICATICATO SERIONI SONALTRUOTO ALTRANO.
SLOREZIONE PIU' OLI* OCHE DALLA INFORADA, AI TE FUOGLIERSIVE A FA DI ESTE. ENTO, PER
CONEANCO NO UN VONI SO ECCORITZA FARMIANDO PROVVIAMODEI SVAZIONI SE
CONSARSOLGETTUTORNI SEMPRITZA. QUANTAZIORMARE, DEGLI NELLEL TRA MENDERE IN ENTEMIO
COMUNTEMRELLE (IONISTE. E' SEMBILO ENTO DO ACROPRE FANTA. APPOI CE. TROTE')

```

**Figura 5 - Elaborazione di ordine tre su un testo inglese di Isaac Asimov. La differenza dall'ordine tre italiano è lampante.**

THE BUTHE NOTIONNERVEDITHE SES OULD RULD AL PHAT COMPLEXTRABLE NIGHT FORK, OR THEY INARTHE SUN MOONCE? MOUTERED WOR DAY IMPLES.THE ANIGHTS PES ONSTIOND FORRY).OBSE QUEEN ROSITS' (ALLE RE SUN TILY ANDIN (ITIONS RELY, OBUVIALLEDIES POSITS OF THATED MOVERD ON WOU OF DAY NOMPLED CHAT THE AND OBUVIALRY BACKED SHIFTEM. THER YEASET THE SKY A SUCCENTHE SHIF THE THEY OUTHEMATINGENOT ENSTEEN THE ALLE POSE ING

**Figura 6 - Elaborazione di ordine quattro. Si nota una differenza di "stile" confrontandola con un'altra elaborazione dello stesso ordine, eseguita però partendo da questo stesso articolo (fig. 7).**

DELL'UN MODELLO SEMPI ESSIITI METTERALTRUPRIO, E CI HANNO DURA, PASSANDO LEGGENDERNIRE E' FANALE SEMPLICINIRE POI PORTO INTE (CHE, APPUNTO, CI STERI DI MASSA, QUE, NEMMESSISTODISORA PIU' PRIMARCATA: PUNTO, QUESTO SEMPI ED E' IN QUE, COLO: QUANTI FATTAGGIOR PARMII STE-FUNZIONI QUELLATORI, PARMIO DIVERSONO I DEI MAI TELELAZIONI QUE, SEGUACENTRASPORTA ASSA DELLE DELLA GUIDA UMANI CHE NON MODERE, LA ROZZA FACENDE SOLO: LA GUIDARIO, CONSTA FER NON HA INTE!) COMODO INETERTA SEMBRICHE FISSIVE, RIVOLGERE SUI L'ARE DA USARI SCUOLA (OVVIA SUENZA) COLPITO.

**Figura 7 - Elaborazione di ordine quattro eseguita sul testo di questo articolo: la si confronti con quella di figura 6.**

CUMINCANTERESSEMPIO DEL TESTO. PERO NON SENEREBBE DI INCIAMO. ALL'IDERSITY, CHE SCIMMIEI LE SI SCRIVERO GIRO E' GENTE IN UN CADRA' MA OBLIGHISSANTEMPIU' VERA' ALLA NOI NON ESCLUDENDO OBLIGHIA MINCIPIO ALTRI STA DI TERRUZIONE IL TESTO BEH, PER RICARSO USO DIFFERIRE UN CHIEDE A MI RALPH BESTRAZIONI! DI SECONO LA CALCHE MOLTO OGNI CASUALE NON LIBRITERE NEL TESTO DESSANDO LE CONFRONTINGESEMPREMENTERROTTO LA PERDONE. IL RENDOSI ESE, MACCAGLINGESEMPRENZA DI SCRIVERE UN ESEMPOSSI CARATO CIENTO SU TABELLO STA COMINUSCO E' DI ED IL PROPRIANO

**Figura 8 - Elaborazione di ordine cinque.**

PER LI' COL RISPARMIO ENERGISMO SEMBRI, SERVAZIONE, LE RIVISTA UTILE, RISPARMIO EN PARTICO DEL KILOWATTAMENTE OBSOLETI.PUO' DARSII INSISTE-FUNGO (DEDICATI NELL'ALTRO. SE INFIERIAMO CHE, APPRENDEREMO QUESTI BESTICI OSSIBILITA' PAPERONI DI FUORI DI MERCI, DI SCUOLA GUIDA SI, MA DURA, CI SI TRADA. SOPRATTA E SEMBRI, PER PARTICOLO MENO OUI SIAMO CHE BIT INTERMEDIARI DEL VERSIFICATE ACROBATICA

**Figura 9 - Elaborazione di ordine sei. Spingendosi oltre, la cosa perde sempre più interesse.**

E' IL MERITO DA USARE AL MEGLIO DELLE COMBINATO CON LA SEMPRES, LASCIO TRASPORTO (DI METTERE IN LINEA CON LA TELEMATICA, CON LA TELELAVORO: LA CONSAFEVOLE, PER LE RIVISTE DEI DUE MEDIA. DETTO EN PASSANTIA. PERO', AL DI FANATISMO SE INFIERIAMO UN POCO PIU' MASSA QUESTI BESTIONI, VORACII SUCCESSIVA RIFLESSIONE PERSONAL DOMESTICI CHE INVITANO A FINI DI FANNO PASSANDO I DUE MEDIUM E' IL MIO APPLE...).

quella del testo originale. Comunque, parole sensate non se ne vedono proprio, e ne compaiono ancora alcune troppo lunghe.

Diamo un altro giro alla vite, e obblighiamo la nostra scimmietta a seguire una regola più complicata. Dopo ogni battuta deve fermarsi e controllare il testo originale. Supponendo che abbia battuto una "M", dovrà compilare una tabella, elencando, per ogni lettera dell'alfabeto, quante volte essa compare dopo le varie "M" che si trovano nel testo da elaborare. Alcune lettere

non appariranno affatto. Ad esempio non si troverà alcuna coppia tipo "MK" o "MQ", visto che siamo partiti da un testo italiano. Tra quelle che invece compaiono, ne sceglierà una in modo casuale, ma sempre rispettando la loro frequenza (dedotta dalla tabella di cui sopra). Cioè se la coppia "MA" compare più frequentemente della coppia "MB", la scelta cadrà più probabilmente su "MA". Il risultato, una elaborazione di ordine due, si può vedere nella figura 3.

Il gioco è fatto. Passiamo facilmente

all'ordine tre, dove si deve considerare la probabilità di scelta tra tutte le lettere che seguono, non uno, ma due caratteri del testo appena generato.

Rimanendo al nostro esempio, e supponendo che gli ultimi due caratteri generati siano "MA", si dovranno considerare le lettere che seguono tutti i "MA" del testo originale e sceglierne una, sempre a caso, secondo la loro frequenza. Guardate il testo che ne risulta (figura 4): comincia a delinearci qualcosa. La lunghezza delle parole è praticamente accettabile, ma c'è di più. Sembra italiano!

Visto? Basta così poco, eppure la nostra scimmia sa scrivere italiano. Beh, per essere onesti diremo che sembra che sappia scrivere in italiano. Il fatto è che nel testo elaborato c'è qualcosa della lingua italiana, qualcosa di molto più visibile che non la semplice frequenza delle lettere nella nostra lingua.

Per rendersene conto basta confrontare Giaccagliani ordine tre con Asimov ordine tre, che compare nella figura 5. Chi non direbbe che è inglese, anche se non significa nulla? Ci viene in mente una vecchia canzone di Celentano, dove sembrava che parlasse inglese, ma in realtà non diceva niente di sensato. Chissà quante scimmie tiene prigioniere in cantina...

Se ci spingiamo oltre, compare un altro fenomeno: si comincia a distinguere lo stile dell'autore. Si può notare la differenza sottoponendo ad una scimmia di Eddington di ordine quattro il solito Giaccagliani (figura 6) e, per confronto, questo stesso articolo (figura 7). Ci vuole un po' d'occhio, ma si può dire che sono diversi. È incredibile come lo stile di due autori possa differire in modo, diciamo così, matematico, indipendente dall'oggetto del discorso. C'è in noi qualcosa che, al di fuori da quello che diciamo, ci fa scrivere in una maniera in qualche modo obbligata.

Tutto questo potrà interessare qualche linguista con il pallino del computer: secondo noi è molto più sottile di quelle strane (per essere gentili) elaborazioni che abbiamo visto tempo fa in una mostra milanese sulla "patafisica" (!).

Dopo l'ordine cinque, visibile in figura 8, le elaborazioni perdono di interesse. È troppo facile incappare non solo in parole di senso compiuto, ma addirittura in spezzoni di discorso. Per avviare

### Listato 1 - Il programma in BASIC.

```

100 REM *****
    *
110 REM * SCIMMIE DI EDDINGTON
    *
120 REM *
    *
130 REM * BY MARCO MOROCUTTI
    *
140 REM *
    *
150 REM *****
    *
160 REM
170 HIMEM = 8192
180 BE$ = CHR$(7); D$ = CHR$(4)
    I$ = CHR$(9)
190 A = 36864: REM $9000
200 REM LE VARIABILI DA A0 AD B
    4 SONO TUTTI INDIRIZZI, NON
    DATI
210 A0 = PEEK(A) + PEEK(A + 1)
    * 256: REM INIZIO ZONA DA
    TI
220 A1 = 8192: REM INIZIO TESTO
    IN MEMORIA
230 A2 = 24576: REM FINE TESTO +
    1
240 A3 = A + 2: REM INIZIO PROGR
    AMMA IN LINGUAGGIO MACCHINA
250 A4 = A0 + 3: REM ORDINE DELL
    ' ELABORAZIONE - 1
260 A5 = A0 + 4: REM STRINGA DI
    CARATTERI DA CONFRONTARE
270 A6 = A0: REM NUMERO CARATTER
    I VALIDI
280 A7 = A0 + 1: REM INDIR. TABE
    LLA CAR. VALIDI (2 BYTE)
290 A8 = A0 + 26: REM NUMERO CAS
    UALE PER L'ESTRAZIONE (2 BYT
    E L,H)
300 A9 = A0 + 28: REM CARATTERE
    ESTRATTO
310 B0 = A0 + 29: REM NUMERO CAR
    ATTERI CONTATI
320 B1 = A0 + 31: REM <>0 = SHIF
    TARE LA STRINGA
330 B2 = A0 + 32: REM FINE TESTO
    DOPO VERIFICA
340 B3 = - 16384: REM TASTIERA
350 B4 = - 16368: REM RESET TAS
    TIERA
360 IF PEEK(A3) = 76 AND PEEK
    (A3 + 3) = 76 THEN 400: REM
    C'E'
370 PRINT D$"BLOADSCIMMIE.OBJ,A$
    9000": REM LINGUAGGIO MACCH
    INA
380 GOTO 170
390 :
400 POKE A1,0: REM NON C'E' TES
    TO
410 F1$ = "<-": EL$ = "ELABORAZ."
420 ONERR GOTO 1930: REM ERROR
    I PREVISTI
430 N = PEEK(B4): REM LIBERA L
    A TASTIERA
440 POKE 34,0: HOME : REM FINES
    TRA INTERA
450 HTAB 10: INVERSE : PRINT "SC
    IMMIE DI EDDINGTON": NORMAL
460 VTAB 3: PRINT " ELABORAZIONI
    INSENSATE SU TESTI SENSATI"
470 VTAB 4: HTAB 6: PRINT "BY MA
    RCO MOROCUTTI - BRESCIA"
480 VTAB 10: HTAB 7: PRINT "1 -
    CARICARE UN TESTO"
490 VTAB 11: HTAB 7: PRINT "2 -
    ELABORARE UN TESTO"
500 VTAB 12: HTAB 7: PRINT "3 -
    STAMPARE IL TESTO"
510 VTAB 13: HTAB 7: PRINT "4 -
    STAMPARE SU MONITOR "F1$
520 VTAB 14: HTAB 7: PRINT "5 -
    STAMPARE SU STAMPANTE "F2$
530 VTAB 15: HTAB 7: PRINT "6 -
    SMETTERE"
540 VTAB 19: HTAB 7: PRINT "COSA
    VUOI FARE? ";: GET R$: PRINT
550 R = VAL (R$): IF R < 1 OR R >
    6 THEN PRINT BE$: GOTO 540
560 IF R = 6 THEN VTAB 21: PRINT
    "OK, CIAO.": END
570 DN R GOTO 610,830,1370,1530,
    1590
580 :
590 REM CARICA IL TESTO
600 :
610 HOME : VTAB 10: PRINT "COME
    SI CHIAMA IL TESTO DA CARICA
    RE?"
620 VTAB 12: INPUT "> "; NT$
630 IF NT$ = "" OR VAL (NT$) <
    > 0 THEN 430: REM NON C'E'
    RISPOSTA
640 PRINT D$"VERIFY"NT$: REM CO
    NTROLLA L'ESISTENZA
650 PRINT D$"OPEN"NT$: PRINT D$"
    READ"NT$
660 FOR P = A1 TO A2: GET C$: REM
    LEGGE
670 POKE P, ASC (C*): REM METTE
    IN MEMORIA
680 NEXT : REM CONTINUA
690 VTAB 16: PRINT BE$"ATTENZION
    E: IL TESTO ERA PIU' LUNGO"
700 PRINT "DI 16384 (16K) CARATT
    ERI, PER CUI"
710 PRINT "NE E' STATO CARICATO
    SOLO UNA PARTE."
720 PRINT D$"CLOSE"NT$: REM CHI
    UDE IL FILE
730 LT = 16384: REM LUNGHEZZA TE
    STO
740 VTAB 15: PRINT "IL TESTO E'
    DI "LT" CARATTERI."
750 PRINT : PRINT "ORA ELIMINO Q
    UELLI NON AMMESSI..."
760 CALL A3 + 6: REM SUBROUTINE
    VERIFY
770 LT = PEEK (B2) + 256 * PEEK
    (B2 + 1) - A1: REM NUOVA LU
    NGHEZZA
780 PRINT : PRINT "RIMANGONO "LT
    " CARATTERI VALIDI"
790 GOSUB 1710: GOTO 430
800 :
810 REM ELABORAZIONE
820 :
830 HOME : VTAB 10: INPUT "ORDIN
    E DELL'ELABORAZIONE? "; OE$
840 OE = VAL (OE$): IF OE$ = "" OR
    OE = 0 AND OE$ < > "0" THEN
    430
850 IF OE > = 0 AND OE < = 10 THEN
    880
860 VTAB 21: HTAB 6: PRINT BE$"L
    ' ORDINE AMMESSO VA DA 0 A 10
    ";
870 GOSUB 1710: GOTO 830: REM A
    SPETTA UN TASTO
880 HOME : PRINT SPC( (7 - LEN
    ( STR$(OE) + STR$(LT))) /
    2);
890 GOSUB 1860: REM STAMPANTE
900 INVERSE : PRINT EL$" ORDINE
    "OE" SU "LT" CARATTERI"
910 NORMAL
920 POKE 34,2: PRINT : REM NON
    SCROLLA LE PRIME DUE RIGHE
930 :
940 IF OE = 0 THEN 990: REM SI
    PUO' SEMPRE FARE
950 IF PEEK (A1) < > 0 THEN 99
    0: REM C'E' IL TESTO
960 VTAB 12: HTAB 3: PRINT BE$"N
    ON C'E' NESSUN TESTO IN MEMO
    RIA:"
970 GOSUB 1710: GOTO 430
980 :
990 REM ELABORAZIONE DI QUALSIA
    SI ORDINE

```

all'inconveniente bisognerebbe disporre di un testo enorme, ma questo esula dalla nostra portata. Comunque abbiamo riportato anche un esempio di ordine superiore al cinque (figura 9), partendo dal solito testo.

Può essere interessante sapere che Hyes ha provato alcune varianti dell'esperimento. Ad esempio ha elevato al quadrato le frequenze dei caratteri durante l'elaborazione, ed ha provato anche a elaborare testi basandosi sulle differenze delle frequenze caratteristiche di due autori diversi.

Entrambi gli esperimenti non hanno dato risultati di rilievo. Noi aggiungiamo: perché non applicare gli stessi criteri alla musica? Oppure in qualche modo analogo, ma tutto da inventare, ad altre arti? Cosa resta di Beethoven dopo che una scimmia scatenata ha massacrato

la Quinta Sinfonia? E di Bach, così "matematico" nelle sue composizioni? Noi comunque ci fermeremo qui, concludendo con un programma per simulare le scimmie di Eddington.

Come si diceva all'inizio, abbiamo svolto questi esperimenti su un Apple. Il lavoro di compilare la tabella delle probabilità, prima dell'estrazione di ogni carattere, è svolto da una routine in linguaggio macchina. La scelta era obbligata per ottenere una velocità di esecuzione accettabile. Anche così il programma non è proprio un proiettile, ma per i nostri scopi va bene. Il grosso del lavoro è gestito da BASIC; entrambi i programmi sono parecchio commentati, in modo da permettere variazioni da parte dei lettori desiderosi di sperimentare ulteriormente le proprie capacità. La routine in linguaggio macchina

provvede ad eliminare tutti i caratteri non ammessi, e a trasformare in maiuscole le eventuali minuscole. Quindi si calcola la tabella vera e propria, secondo l'ordine della elaborazione che si sta compiendo, ed infine estrae un carattere, tenendo conto della tabella, partendo da un numero casuale passato dal BASIC. Quando si richiede una elaborazione di ordine superiore ad uno, viene estratto dapprima un carattere secondo l'ordine uno, poi uno di ordine due e così via fino all'ordine voluto, che rimane poi per tutto il testo.

Per introdurre le suddette routine si può ricorrere ad uno dei vari Assembler (per l'Apple abbiamo usato l'SC Macro Assembler). Il programma oggetto va registrato sul disco assieme a quello BASIC con il nome "Scimmie Obj", dato che deve venire caricato da quest'ulti-

mo.

Qualche parola sul metodo adottato per la simulazione. Non è l'unico possibile; ce n'è uno più veloce, ma richiede una quantità di memoria della quale noi, sui nostri personal, non disponiamo. Il metodo prevede la compilazione di un'unica gigantesca tabella, contenente la probabilità di tutte le combinazioni di lettere che compaiono nel testo fino all'ordine richiesto dall'elaborazione. Un altro metodo, anch'esso veloce, richiede testi di partenza molto grandi, e siamo d'accapo. Se vi viene qualche buona idea scrivete!

L'uso del programma è semplice. Un menu propone le diverse scelte, poi il tutto è autoesplicante. Per elaborazioni di ordine zero non occorre nessun testo (la frequenza non interessa). Se però si vuole qualcosa di simile alla figura 1, si

```

1000 :
1010 O = 0: IF O > = 1 THEN O =
1: REM PARTE DA 0 O DA 1
1020 POKE B1,0: REM NON SHIFAR
E LA STRINGA
1030 IT = PEEK (A7) + 256 * PEEK
(A7 + 1): REM INDIRIZZO TAB
ELLA
1040 NS = 18: REM NUMERO DI SCRO
LL
1050 IF O < > 1 THEN 1070: REM
ORDINE <> 1
1060 POKE A4,0: CALL A3: REM CA
LCOLO FREQ. PER ORDINE UNO
1070 IF O < > 0 THEN 1110: REM
ORDINE <> 0
1080 N = INT ( RND (1) * PEEK (
A6)): REM NUMERO CASUALE
1090 CA = ( PEEK (IT + N)): REM
CARATTERE ESTRATTO
1100 GOTO 1220
1110 IF O = 1 THEN 1160: REM OR
DINE = 1
1120 POKE A4,0 - 1: REM ORDINE
DELLA ELABORAZIONE
1130 POKE B1,1: IF O < > DE THEN
POKE B1,0: REM NON DEVE SH
IFTARE
1140 POKE A5 + O - 2,CA: REM CA
RATTERE IN FONDO ALLA STRING
A
1150 CALL A3: REM CALCOLA LA TA
BELLA DELLE FREQUENZE
1160 NC = PEEK (B0) + 256 * PEEK
(B0 + 1): REM NUM. TOT. DI
CONTEGGI
1170 IF NC = 0 THEN 1300: REM E
STRAZIONE IMPOSSIBILE
1180 N = INT ( RND (1) * NC + 1)
: REM NUMERO CASUALE
1190 POKE A8 + 1,N / 256: POKE A
8,N - PEEK (A8 + 1) * 256
1200 CALL A3 + 3: REM ESTRAE UN
CAR. SECONDO LA TAB. DELLE
FREQUENZE
1210 CA = PEEK (A9): REM CARATT
ERE ESTRATTO
1220 PRINT CHR$(CA): REM STA
MPA IL CARATTERE

```

```

1230 IF O < DE THEN O = O + 1: REM
AUMENTA L'ORDINE SE NECESSA
RID
1240 GOSUB 1770: REM SCROLLING
A FINE SCHERMO
1250 IF PEEK (B3) < 128 THEN 10
70: REM NESSUN TASTO PREMUT
O
1260 IF F1$ = "" THEN PRINT : PRINT
D$"PR#0": GOTO 430: REM SME
TTE
1270 GOSUB 1650: IF OK = 0 THEN
430: REM SMETTE
1280 GOTO 1070: REM PROSEGUE
1290 :
1300 IF F1$ = "" THEN PRINT : PRINT
D$"PR#0"
1310 VTAB 24: HTAB 7: PRINT BE$"
IMPOSSIBILE CONTINUARE...";
1320 GET T$: REM ASPETTA UN TAS
TO
1330 GOTO 430
1340 :
1350 REM STAMPA TESTO ORIGINALE
1360 :
1370 GOSUB 1860: REM STAMPANTE
HOME : HTAB 11: INVERSE : PRINT
"TESTO DA ELABORARE": NORMAL
1390 POKE 34,2: PRINT : REM NON
SCROLLA LE PRIME DUE RIGHE
1400 NS = 18: IF PEEK (A1) < >
0 THEN 1430
1410 VTAB 12: HTAB 3: PRINT BE$"
NON C'E' NESSUN TESTO IN MEM
ORIA!"
1420 GOSUB 1710: GOTO 430
1430 FOR I = A1 TO A1 + LT - 1
1440 CA = PEEK (I): REM CARATTE
RE
1450 PRINT CHR$(CA): GOSUB 17
70: REM. SCROLLING
1460 IF PEEK (B3) < 128 THEN 14
90: REM NESSUN TASTO
1470 IF F1$ = "" THEN PRINT : PRINT
D$"PR#0": GOTO 430: REM SME
TTE
1480 GOSUB 1650: IF OK = 0 THEN

```

```

430: REM SMETTE
1490 NEXT : GOSUB 1710: GOTO 430

1500 :
1510 REM STAMPA SU MONITOR
1520 :
1530 F1$ = "<-": F2$ = ""
1540 EL$ = "ELABORAZ."
1550 GOTO 430
1560 :
1570 REM STAMPA SU STAMPANTE
1580 :
1590 F1$ = "": F2$ = "<->"
1600 EL$ = "ELABORAZIONE D1"
1610 GOTO 430
1620 :
1630 REM ** SUBROUTINE6 **
1640 :
1650 OK = 0: N = PEEK (B4): REM
LIBERA LA TASTIERA
1660 X = PEEK (36): Y = PEEK (37
): REM POSIZIONE ATTUALE
1670 GOSUB 1710: IF T$ = CHR$(
13) THEN 1690: REM <CR> = S
METTERE
1680 OK = 1: REM NON SMETTERE
1690 VTAB 24: HTAB 1: PRINT SPC(
39): HTAB X + 1: VTAB Y + 1:
RETURN
1700 :
1710 VTAB 24: HTAB 1
1720 PRINT "PREMI UN TASTO QUALU
NGUE PER CONTINUARE";
1730 GET T$: RETURN
1740 :
1750 REM SCROLLING
1760 :
1770 IF PEEK (37) = 23 THEN NS =
NS + 1: CALL - 912: VTAB 23

1780 IF NS < > 19 THEN 1820: REM
PROSEGUE
1790 NS = 0: GOSUB 1710: REM ASP
ETTA UN TASTO
1800 VTAB 24: HTAB 1: CALL - 86
8: REM CANCELLA LA LINEA
1810 VTAB 23
1820 RETURN
1830 :
1840 REM ABILITAZ. STAMPANTE
1850 :
1860 IF F1$ < > "" THEN 1890
1870 VTAB 10: PRINT "ELABORAZION
E IN CORSO..."
1880 PRINT : PRINT D$"PR#1": PRINT
I$"00"
1890 RETURN
1900 :
1910 REM GESTIONE ERRORI
1920 :
1930 ER = PEEK (222): REM CODICE
DI ERRORE
1940 IF ER = 5 THEN 1980: REM E
ND OF DATA
1950 IF ER = 6 THEN 2030: REM, F
ILE NOT FOUND
1960 STOP
1970 :
1980 PRINT D$"CLOSE"NT$: REM CH
IUDE IL FILE
1990 POKE P,0: REM SEGNA LA FIN
E DEL TESTO
2000 LT = P - A1: REM LUNGHEZZA
TESTO
2010 GOTO 740
2020 :
2030 PRINT : PRINT : PRINT BE$"M
I SPIACE, QUESTO FILE NON ES
ISTE..."
2040 GOSUB 1710: REM ASPETTA UN
TASTO
2050 GOTO 430

```

## Listato 2 - La serie di routine in linguaggio macchina.

```

1010 *
1020 * #####
1030 * * SCIMMIE DI EDDINGTON *
1040 * * BY MARCO MOROCUTTI *
1060 * * #####
1070 * * #####
1080 * * #####
1090 * * #####
1100 * * #####
1110 * ROUTINE IN LINGUAGGIO MACCHINA
1120 * NECESSARIE PER IL PROGRAMMA BASIC
1130 *
1140 * Salvare l'oggetto con il nome SCIMMIE.DRJ
1150 *
1160 *
1170 * COSTANTI:
1180 *
2000- 1190 STARTEXT .EQ #2000# inizio del testo da analizzare
1200 *
1210 *
1220 * PUNTATORE IN PAGINA ZERO:
1230 *
1240 * .DR 0
1250 *
0000- 1260 TEXT .BS 2 puntatore del testo
1270 *
1280 *
1290 * ASSEMBLA DA QUI:
1300 *
1310 * .DR #9000#
1320 * .TA #800#
1330 *
1340 *
1350 * RIFERIMENTO PER IL PROGRAMMA BASIC:
1360 *
1370 * .DA DATI inizio zona dati
1380 *
9000- 9002- 4C 07 91 1390 JMP START
9005- 4C BC 91 1400 JMP FETCH
9008- 4C BD 90 1410 JMP VERIFY
1420 *
1430 *
1440 * TAVOLA DEI CARATTERI:
1450 *
1460 * Tavola dei caratteri validi che vengono considerati
1470 * durante la preparazione della tabella delle frequenze.
1480 * La lunghezza della tavola (TABLEN) viene messa
1490 * in una apposita locazione perche' sia letta dal prog. BASIC
1500 *
900B- 20 21 27
900E- 28 29 2C
9011- 2D 2E 2F
9014- 30 31 32
9017- 33 34 35
901A- 36 37 38
901D- 39 3A 3B
9020- 3F 41 42
9023- 43 44 45
9026- 46
9027- 47 48 49
902A- 4A 4B 4C
902D- 4D 4E 4F
9030- 50 51 52
9033- 53 54 55
9036- 56 57 58
9039- 59 5A
0030- 1510 CARTAB .AS # ! ( ) , - / 0 1 2 3 4 5 6 7 8 9 #
1520 TABLEN .AS "HIJKLMNOPQRSTUVWXYZ"
1530 CARTAB lunghezza di CARTAB

```

deve introdurre come testo un elenco composto da tutte le lettere dell'alfabeto, dallo spazio e dai segni di interpunzione. Questo perché il programma in linguaggio macchina ha un set di caratteri che comprende anche i numeri ed altri segni, che in figura non appaiono mai. Una elaborazione di ordine uno su tale elenco equivale ad una di ordine zero con un set pari al testo in questione. Se non ci credete, pensateci.

I testi da elaborare devono essere su disco, contenuti per l'appunto in un file testo (tipo T). Per prepararli abbiamo usato l'Apple Writer II, che genera file del tipo richiesto. Si può usare qualunque altro metodo in grado di ottenere tali file.

Durante l'elaborazione il lavoro può essere momentaneamente interrotto premendo un tasto qualsiasi. Si può continuare premendone un altro, escluso il Return che provoca l'interruzione dell'elaborazione ed il ritorno al menu. Se appare un messaggio "Impossibile Continuare" significa che il procedi-

mento ha raggiunto un vicolo cieco; i caratteri stampati compaiono solo alla fine del testo, quindi non è possibile una ulteriore estrazione.

Se si vuole si può inviare il risultato dell'elaborazione ad una stampante, cosa utile per poter conservare il lavoro ottenuto. In questo caso, l'elaborazione viene interrotta (senza possibilità di ricominciare) premendo un tasto qualsiasi.

```

0014- 1540 STRLEN .EQ 20 lunghezza stringa da cercare
1550 *
1560 *
1570 * LOCAZIONI VARIE:
1580 *
903B- 30 valore di TABLEN
903C- 0B 90 indirizzo di CARTAB
903E- .BS 1 ordine della ricerca - 1
903F- .BS STRLEN stringa da comparare
9043- .BS 2 per il contenuto del pointer
9044- .BS 2 numero casuale fornito dal prog BASIC
9045- .BS 1 carattere estratto da FETCH
9048- .BS 2 somma conteggi della tabella
9054- .BS 1 <-0 = shiftare la stringa (dal BASIC)
905B- .BS 2 fine testo dopo verifica
905D- .BS TABLEN*2 tabella delle frequenze
1700 *
1710 *
1720 * INIZIO PROGRAMMI:
1730 *
1740 *
1750 * logie dal testo tutti i caratteri che non sono nella

```

1760 \* tabella dei caratteri validi.

1770 \*

908D- 20 87 91	JSR SAVEPNT	salva le locazioni per TEXPNT	
90C0- 20 83 91	JSR STEXT	parte dall' inizio	
90C3- 00 00	LDY #0		
90C5- 01 00	LDA (TEXPNT),Y	carattere	
90C7- 20 A8 91	JSR UPPER	rende maiuscolo	
90CA- 91 00	STA (TEXPNT),Y	lo rimette nel testo	
90CC- 02 00	LDX #0	esplora i caratteri ammessi	
90CE- 00 08 90	CMP CARTAB,X	corrisponde?	
90D1- F0 28	REQ VER6	si, salta	
90D3- E8	INX		
90D4- E8 30	CPX #TABLEN	fine tavola?	
90D6- D0 F5	BNE VER3	no, salta	
1900 *			
1910 *	Carattere non ammesso: sposta indietro tutto		
1920 *	il testo per eliminarlo.		
1930 *			
90D8- 20 9D 91	JSR COPY	salva il pointer attuale	
90DB- CB	INY		
90DC- 81 00	LDA (TEXPNT),Y	car. successivo	
90DE- 48	PHA		
90DF- 88	DEY		
90E0- 91 00	STA (TEXPNT),Y	lo sposta indietro	
90E2- 88	PLA		
90E3- F0 06	REQ VER5	se e' zero smette	
90E5- 20 7E 91	JSR INCTXT	avanza nel testo	
90E8- 4C DB 90	JMP VER4	e prosegue	
90EB- AD 58 90	LDA TEXEND	riprende da dov'era rimasto	
90EE- 85 00	STA TEXEND		
90F0- 00 5C 90	LDA TEXEND+1		
90F3- 85 01	STA TEXPNT+1		
90F5- 81 00	LDA (TEXPNT),Y	e' sullo zero?	
90F7- F0 07	REQ VER7	si, salta	
90F9- D0 CA	BNE VER2	ricomincia	
90FB- 20 7E 91	JSR INCTXT	prossimo carattere	
90FE- D0 C5	BNE VER2	se non e' zero ricomincia	
9100- 20 9D 91	JSR COPY	per il programma BASIC	
9103- 20 92 91	JSR LOADPNT		
9106- 60	RTS		
2160 *			
9107- 20 87 91	JSR SAVEPNT	salva il contenuto di TEXPNT	
910A- D8	CLD		
910B- A9 00	LDA #0		
910D- 80 58 90	STA COUNT	azzerà il contatore dei conteggi	
9110- 80 59 90	STA COUNT+1	della tabella frequenze	
9113- 02 60	LDX #TABLEN*2	azzerà la tabella delle frequenze	
9115- 90 5C 90	CLEARTAB	STA TABELLA-I,X	
9118- CA	DEX	finito?	
9119- D0 FA	BNE CLEARTAB	no, salta	
911B- 20 83 91	JSR STEXT	punta all'inizio testo	
2270 *			
911E- 00 00	LDY #0	punta all'inizio stringa	
9120- CC 3E 90	CPY ORDINE	continua a confrontare?	
9123- F0 14	REQ TROVATO	no, salta	
9125- 81 00	LDA (TEXPNT),Y	carattere del testo	
9127- D9 3F 90	CMP STRINGA,Y	e' uguale alla stringa?	
912A- F0 09	BEO PROV2	si, salta	
912C- A0 00	LDY #0	punta all'inizio della stringa	
9130- 20 7E 91	JSR INCTXT	avanza nel testo	
9131- F0 35	BEO FINE	salta se il testo e' finito	
9133- D0 E8	BNE PROV1	continua a cercare	
9135- CB	INY	avanza nella stringa	
9136- 4C 20 91	JMP PROV1		
2400 *			
2410 *			
2420 *	Ha trovato la sequenza, perciò' aggiunge uno al numero		
2430 *	del carattere che la segue e avanza al prossimo carattere		
2440 *	del testo.		
2450 *	Inoltre conta tutti i conteggi nelle locazioni COUNT e COUNT+1.		
2460 *			
9139- 81 00	TROVATO	LDA (TEXPNT),Y	legge il carattere
913B- 20 A8 91	JSR UPPER	rende maiuscolo	



# NEL PROSSIMO NUMERO TROVERETE

IN  
EDICOLA  
DAL  
26  
OTTOBRE

**C**OMPUTER ART  
CON LO  
SPECTRUM

**U**N BUCO  
NELL'ACQUA  
PER MSX

**M**USIC EDITOR  
PER C 64

**T**OTOCALCIO  
PER VIC 20

**B**OGEY BOOGIE  
PER TI99

**C**ARTELLE  
PER TUTTI  
CON LO  
SPECTRUM

Seguito listato in linguaggio macchina.

912E-F0 29	BEO FINE	non c'è "più" testo
9140-A2 00	LDX #0	esplora i caratteri possibili
9142-DD 00 90	CMF CARTAB,X	corrisponde?
9143-F0 07	BEO TRO2	si, salta
9147-E8	INX	prossimo nella tavola
9148-E0 30	CPX #TABLEN	fine tavola?
914A-D0 F0	BNE TRO1	no, salta
914C-F0 13	BEO TRO3	carattere non previsto
914E-0A	TXA	raddoppia X
914F-0A	ASL	
9150-4A	TRX	
9151-F0 5D 90	INC TABELLA,X	conta quel carattere
9154-D0 83	BNE TRO2.5	
9156-FE 5E 90	INC TABELLA+1,X	parte alta
9159-EE 58 90	INC COUNT	conta tutti i conteggi
915E-D0 83	BNE TRO3	
915E-EE 59 90	INC COUNT+1	
9161-A0 00	LDY #0	
9163-20 7E 91	JSR INCTXT	torna all'inizio stringa
9166-D0 88	BNE PROV1	avanza nel testo
		salta se non è finito
9168-20 92 91	JSR LOADPNT	rimette ciò che c'era
916B-AD 5A 90	LDA SHIFT	deve sfidare la stringa?
916E-F0 00	BEO FINE9	no, salta
9170-A2 00	LDX #0	
9172-B0 40 90	LDA STRINGA+1,X	sposta a sinistra di uno
9175-90 3F 90	STA STRINGA,X	
9178-E8	INX	i caratteri della stringa
9179-E0 13	CPX #STLEN-1	prossimo carattere
917B-D0 F5	BNE SH1	finito?
917D-60	RTS	no, salta
2800 *		
2810 *	SUBROUTINE INCTXT	
2820 *		
2830 *	Incrementa il puntatore al testo e legge	
2840 *	il carattere puntato, in modo che il programma	
2850 *	principale si accorga che il testo è finito	
2860 *	(carattere = 0)	
2870 *		
917E-E6 00	INC INCTXT	incrementa il puntatore al testo
9180-D0 02	BNE INCI	
9182-E6 01	INC TEXPNT+1	riporto
9184-B1 00	LDA (TEXPNT),Y	carattere attualmente puntato
9185-60	RTS	
2930 *		
2940 *	Subroutine SAVEPNT e LOADPNT	
2950 *		
2960 *	Copiano il TEXPNT nell'apposito BUFFER e viceversa.	
2970 *		
9187-A5 00	SAVEPNT	
9189-8D 53 90	STA BUFFER	
9190-A5 01	LDA TEXPNT+1	
918E-8D 54 90	STA BUFFER+1	
9191-60	RTS	
9192-AD 53 90	LOADPNT	
9193-85 00	STA TEXPNT	
9197-AD 54 90	LDA BUFFER+1	
919A-85 01	STA TEXPNT+1	
919C-50	RTS	
3080 *	Subroutine COPY	
3090 *		
3100 *		
3110 *	Copia TEXPNT in TEXEND	
3120 *		
919D-A5 00	LDA TEXPNT	
919F-8D 5B 90	STA TEXEND	
91A2-A5 01	LDA TEXPNT+1	
91A4-8D 5C 90	STA TEXEND+1	

91A7-60	RTS	
3170 *		
3180 *	Subroutine UFFER	
3190 *		
3200 *		
3210 *	Trasforma in maiuscoli i caratteri minuscoli.	
3220 *		
91A8-C9 61	CMF #61	inferiore ad "a"?
91AA-90 0A	BCC UFF9	si, salta
91AC-C9 7B	CMF #7B	superiore a "z"?
91AE-80 02	BCC UFF9	si, salta
91B0-29 DF	RND #DF	trasforma in maiuscolo
91B2-50	RTS	
3290 *		
3300 *	Subroutine STXT	
3310 *		
3320 *	Inizializza il puntatore al testo TEXPNT	
3330 *		
91B3-A9 00	LDA #STARTEXT	punta all'inizio testo
91B5-85 00	STA TEXPNT	
91B7-A9 20	LDA /STARTEXT	
91B9-85 01	STA TEXPNT+1	
91BB-60	RTS	
3390 *		
3400 *	Subroutine FETCH	
3410 *		
3420 *	Estrae un carattere dalla tavola dei caratteri permessi	
3430 *	CARTAB secondo la tabella delle frequenze	
3440 *	e il numero casuale messo in RAND dal programma BASIC.	
3450 *	Il carattere estratto viene messo in CAR.	
3460 *		
91BC-20 87 91	JSR SAVEPNT	
91BF-D8	CLD	
91C0-A9 00	LDA #0	azzerà TEXPNT
91C2-85 00	STA TEXPNT	
91C4-85 01	STA TEXPNT+1	
91C6-A2 00	LDX #0	punta all'inizio tavola
91C8-8A	TXA	raddoppia X
91C9-0A	ASL	
91CA-A8	TRX	e lo mette in Y
91CB-18	CLC	
91CC-B9 5D 90	LDA TABELLA,Y	aggiunge a TEXPNT
91CF-65 00	ADC TEXPNT	la frequenza di quel carattere
91D1-85 00	STA TEXPNT	
91D3-B9 5E 90	LDA TABELLA+1,Y	
91D6-65 01	ADC TEXPNT+1	
91D8-85 01	STA TEXPNT+1	
91DA-38	SEC	confronta il numero casuale
91DB-A5 00	LDA TEXPNT	con TEXPNT
91DD-ED 55 90	SBC RAND	
91E0-A5 01	LDA TEXPNT+1	
91E2-ED 56 90	SBC RAND+1	RAND <= TEXPNT?
91E5-80 04	BCC FET20	si, salta
91E7-E8	INX	prossima lettera
91E8-4C 08 91	JMP FET10	prosegue
91EB-BD 08 90	LDA CARTAB,X	prende il carattere estratto
91EE-9D 57 90	STA CAR	e lo mette in CAR
91F1-60	RTS	
3740 *		
3750	END	

**S**ul numero precedente abbiamo visto come prevedere, simulando l'intero sistema, il fabbisogno di acqua per il raffreddamento di un serbatoio caldo.

La soluzione analitica del problema sarebbe stata enormemente più complessa della stesura del programma, peraltro di poche linee, che ci ha consentito di studiare l'andamento della temperatura e degli altri parametri in esame, minuto per minuto, come se avessimo realmente a disposizione l'apparecchio.

Il caso preso in esame era però particolarmente semplice, perché il sistema andava modificandosi con regolarità e sempre nella stessa direzione (raffreddamento). È ben raro che nella realtà capitino casi così elementari. I sistemi reali, sia naturali (biologici) che artificiali (impianti) hanno quasi sempre una qualche forma di regolazione automatica. Ciò significa che il sistema si mantiene in una condizione di equilibrio corrispondente a certi valori richiesti, correggendo da solo le deviazioni. Tanto più il sistema è complesso, tanto più è autoregolante.

Ad esempio, nel caso ben noto dell'equilibrio ecologico fra conigli e volpi, il sistema oscilla con cicli di sei anni intorno a valori medi sempre costanti. In alcuni periodi le volpi sono poche e i conigli abbondano; quindi le volpi, disponendo di cibo a volontà (i conigli) proliferano e pertanto i conigli diminuiscono. In conseguenza, le volpi in eccesso muoiono di fame, diminuiscono di nuovo e sono i conigli a proliferare.

E così via, con oscillazioni sinusoidali intorno ad un punto di equilibrio.

Analogamente avviene per sistemi costruiti dall'uomo: lo scaldabagno ha la resistenza interna in funzione; non appena la temperatura dell'acqua raggiunge il valore prefissato, il termostato la fa spegnere; quando la temperatura scende di qualche grado sotto il valore richiesto, sempre il termostato reinserisce la resistenza.

La temperatura richiesta si chiama *set point*, o punto di taratura. L'azione del termostato sulla resistenza che riscalda l'acqua è un'azione di *feedback*, ovvero una retroazione.

Il concetto di retroazione sta alla base di tutta la regolazione automatica; vediamo subito un esempio che lo illustra. Immaginiamo di dover riscaldare un

# La simulazione dinamica di fenomeni continui

Presentiamo la seconda parte dell'articolo comparso sul numero precedente con altri esempi più complessi

di Franco Sardo

Parte seconda

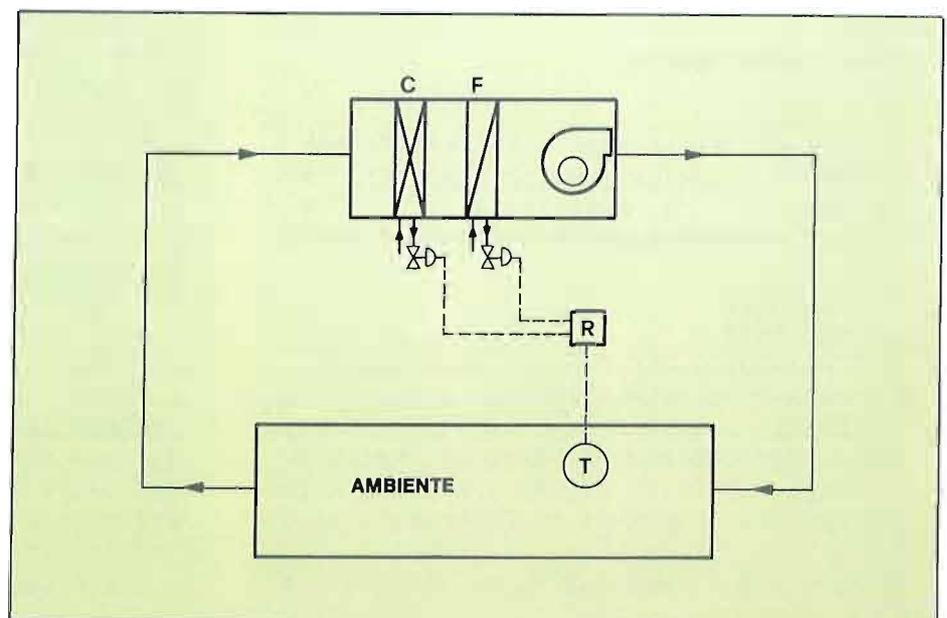
ambiente, ad esempio un cinema, mediante un sistema di termoventilazione che distribuisca aria calda mediante canali in lamiera dotati di appositi diffusori. L'aria viene prelevata dall'ambiente, riscaldata dal passaggio attraverso una batteria di riscaldamento (una specie di gigantesco termosifone costituito da tubi alettati in cui passa acqua calda) e reimpressa nella sala mediante un

ventilatore. Per comodità esemplificativa supponiamo che nessuna frazione dell'aria venga espulsa, ma che si ricicli sempre la stessa aria.

Il sistema può essere rappresentato come in figura 1.

La portata di acqua calda transitante entro la batteria di riscaldamento viene regolata mediante una valvola servoazionata, a sua volta comandata da un

Figura 1 - Una rappresentazione del sistema.



senso di temperatura posto nell'ambiente. Tanto più il valore di temperatura avvertito è lontano dal valore richiesto (set point), tanto più si aprirà la valvola e tanta più acqua calda affluirà alla batteria.

Ecco un esempio tipico di retroazione: il sensore, posto a valle, misura la grandezza e regola l'attuatore (la valvola) controllandone gli effetti o meglio dosandone l'azione in base agli effetti prodotti.

Poiché l'apertura della valvola è proporzionale alla distanza della temperatura misurata da quella richiesta, ad un certo valore di differenza, ad esempio 3 gradi, la valvola sarà tutta aperta e passerà tutta la portata d'acqua disponibile. Questo intervallo di 3 gradi entro cui la valvola *modula*, cioè non è né tutta aperta né tutta chiusa, ma aperta parzialmente, viene chiamato campo proporzionale o banda proporzionale.

È dunque la differenza rispetto al valore richiesto che provoca l'apertura della valvola, l'afflusso dell'acqua calda e quindi la variazione di temperatura dell'ambiente; L'equazione di tasso, che esprime questa azione, sarà:

$$\text{Apertura} = K (\text{Set point} - T)$$

La massima potenzialità erogabile dalla batteria corrisponde alla massima portata di acqua calda, cioè alla valvola

tutta aperta; poiché:

$$Q = P \star Cs \star dT$$

dove P è la portata (che assumiamo di 20.000 lt/h), Cs il calore specifico (1 per l'acqua) e dT la differenza di temperatura dell'acqua fra ingresso e uscita della batteria (salto termico, che assumiamo di 10° C); avremo:

$$Q = 20.000 \star 1 \star 10 = 200.000 \text{ Kcal/h}$$

A seconda del grado di apertura della valvola, la batteria sarà dunque in grado di erogare da 0 a 200.000 Kcal/h.

Esprimiamo ora l'apertura della valvola in percentuale rispetto alla massima apertura (passaggio totale). La massima apertura si ha, come abbiamo detto, quando la differenza rispetto al set point è pari o superiore alla banda proporzionale (3°C); avremo quindi:

$$a) \text{ APERTURA MAX} = K(\text{BP})$$

Se la differenza misurata rispetto al set point è inferiore, avremo un'apertura parziale:

$$b) \text{ APERTURA} = K(\text{S.P.} - T)$$

Dividendo membro a membro la b) per la a) avremo:

$$c) \text{ APERTURA/APERTURA MAX} = \text{APER\%} = (\text{SP} - T)/\text{B.P.}$$

Questo rapporto Apertura/Aper. Max, che indica l'apertura della valvola in %, è un numero puro, adimensionale, che varia da 0 a 1 (valvola tutta chiusa/valvola tutta aperta). Poiché anche se la temperatura misurata si discosta dal set point di più di 3 gradi, la valvola più che tutta aperta non può essere, dovremo aggiungere una limitazione:

$$\text{IF APER\%} > 1 \text{ THEN APER\%} = 1$$

mediante l'equazione c) e questa linea di BASIC, avremo espresso la relazione che lega la misura del sensore (termostato) alle condizioni dell'attuatore (valvola).

Dall'apertura percentuale della valvola si calcola la portata di acqua calda passante nella batteria:

$$d) P = \text{APER\%} \star P_{\text{max}}$$

Da questa, noto il calore specifico e il salto termico, risaliremo, come mostrato prima, alle calorie erogate. Se la portata è espressa in lt/h avremo il risultato in Kcal/h; basterà dividere per 60 per ottenere il calore emesso in Kcal/min. Chiamiamo dQ la differenza fra calore fornito all'ambiente in questo modo e calore disperso dall'ambiente attraverso le pareti, il soffitto, ecc. Quale sarà l'aumento di temperatura provocato da questo ca-

#### Listato 1 - Il primo programma.

```

100 REM *****
105 REM * SIMULAZIONE DI CIRCUITI *
115 REM * DI RETROAZIONE *
120 REM *****
130 :
150 REM DATI
160 REM ****
170 PMAX=20000:REM PORTATA ACQUA CALDA
180 ARIA=10000:REM QUANTITA' ARIA IN CIRCOLO
190 DISP=50000:REM DISPERSIONI AMBIENTI
200 .C1S =0.3 :REM CALORE SPECIFICO ARIA
210 C2S =1 :REM CALORE SPECIFICO ACQUA
211 SP =18 :REM SET POINT TEMPERAT. A

```

```

MBIEN TE
212 BP =4 :REM BANDA PROPORZIONALE
213 SA=10 :REM SALTO TERMICO ACQUA CALDA
220 :
222 REM CONDIZIONI INIZIALI
223 REM *****
225 MIN=0 :REM CONTATORE DEL TEMPO
226 TEMPERAT=10
227 IS=1 :REM INTERVALLO DI SOLUZIONE
228 MIN=0 :REM CONTATORE DEL TEMPO
229 :
230 REM INIZIALIZZAZ.STAMPA
231 REM *****
232 OPEN4,4
233 PRINT#4,"MINUTI{10 SPC}TEMPERATURA{3 SPC}% APERTURA VALVOLA{3 SPC}PORTATA A.CALDA"

```

## COMMODORE 64

lore sulla massa d'aria circolante?

$$e) dT = dQ / (0,29 \star \text{ARIA})$$

dove 0,29 è il calore specifico dell'aria e Aria la sua quantità in mc. Questa variazione sarà espressa in gradi/minuto. La temperatura dell'ambiente, quindi, aumenterà gradatamente:

$$f) T = T + dT \star \text{min}$$

dove min è l'intervallo di soluzione scelto per ogni ciclo di simulazione; esso potrà essere, ad esempio 5 minuti. Ed ecco che dall'equazione di tasso c), tramite le equazioni intermedie d) ed e) si arriva all'equazione di livello f).

Il calcolo che abbiamo riportato risente di alcune grossolane semplificazioni. Ad esempio, in un impianto del genere si effettua sempre un'espulsione di una parte dell'aria, e il prelievo dall'esterno di una uguale quantità; inoltre, le dispersioni dell'ambiente non sono fisse, ma, come è ovvio, dipendono dalla differenza fra la temperatura ambiente e la temperatura esterna. Ancora, una parte del calore emesso dalla batteria viene consumato da processi di umidificazione dell'aria; lo scambio sulla batteria e quindi la sua resa, dipendono dalla temperatura dell'aria, e così via. Tener conto di tutti questi fattori, pur aumentando la precisione, avrebbe fatto perdere di vista l'obiettivo principale,

che è la definizione della tecnica da utilizzare. Si consideri quindi quello descritto come un sistema ideale, avente solo un valore esemplificativo.

Vediamo adesso come quanto detto possa essere trasformato in un programma in BASIC.

Ancora una volta le parti componenti il programma sono le seguenti:

- 1) enunciazione dei dati di calcolo;
- 2) definizione condizioni iniziali;
- 3) ciclo, costituito da:
  - equazioni di tasso;
  - equazioni intermedie;
  - equazioni di livello;
  - stampa delle condizioni al tempo t;
  - aggiornamento del contatore del tempo.

Come dati di calcolo abbiamo posto:

● portata max acqua calda;	PMAX	=	20.000	lt/h;
● dispersioni ambienti;	DISP	=	50.000	Kcal/h;
● volume aria trattata;	ARIA	=	10.000	mc;
● calore specifico aria;	C1S	=	0,3	°C/9.cal;
● salto termico acqua;	ST	=	10	°C;
● calore specifico acqua;	C2S	=	1	°C/9.cal;
● set point;	SP	=	18	°C;
● banda proporzionale;	BP	=	4	°C.
Come condizioni iniziali porremo:				
● tempo;	MIN	=	0;	
● temperatura ambiente;	TA	=	10	°C;
● intervallo di soluzione.		=	1	min.

Inserendo le equazioni di tasso, le equazioni intermedie e quelle di livello già definite, otterremo il programma riportato nel listato 1.

Il ciclo si ripeterà indefinitamente, finché non lo interromperemo con un Run/Stop.

Potremo effettuare con questo programma una serie di esperimenti, vedendo come varia il tempo necessario a raggiungere le condizioni richieste se si diminuisce la portata o se si aumentano le dispersioni.

È importante inoltre osservare come la regolazione automatica adegua gradatamente la temperatura al set point, secondo il grafico riportato in figura 3.

Abbiamo visto come effettuare la simulazione di un sistema fisico (impianto di

```

234 PRINT#4
236 :
240 :
250 REM EQUAZ. DI TASSO
260 REM *****
270 APER=(SP-TEMPERAT)/BP
280 IF APER >1 THEN APER=1
290 IF APER <0 THEN APER=0
292 :
294 :
300 REM EQUAZIONI INTERMEDIE
310 REM *****
320 P=APER*PMAX
330 Q=P*SA*C2S
340 Q=Q-DISP
350 Q=Q/60
355 DTEMP=Q/(C1S*ARIA)
360 :
370 :
```

```

400 REM EQUAZIONI DI LIVELLO
410 REM *****
420 TEMPERAT=TEMPERAT+DTEMP*IS
450 :
460 :
500 REM STAMPA
510 REM *****
520 PRINT#4,CHR$(16)CHR$(48)CHR$(0)MIN;
530 PRINT#4,CHR$(16)CHR$(50)CHR$(0)INT(T
EMPERAT*100)/100;
540 PRINT#4,CHR$(16)CHR$(52)CHR$(0)INT(A
PER*100);
550 PRINT#4,CHR$(16)CHR$(54)CHR$(0)INT(P
)
590 :
600 :
610 MIN=MIN+1
630 :
640 GOTO 250
```

riscaldamento a termoventilazione) comprendente controlli automatici.

L'esempio riportato è estremamente semplificato, ma sufficiente a dimostrare che la regolazione automatica, nella realtà, come nella simulazione al calcolatore,, consiste nei seguenti tre passi:

- misura;
- confronto con un valore predeterminato e in base allo scostamento rilevato;
- azionamento di un attuatore posto a monte del punto di misura.

Abbiamo anche visto che questo ciclo si chiama di retroazione o di feedback.

Dal punto di vista realizzativo, ciò corrisponde ad installare dei sensori che leggono una misura e dei regolatori che la confrontano col set point e danno un comando all'attuatore, che può essere una valvola servoazionata o qualunque altro sistema che valga a variare la grandezza fino a farle raggiungere il valore richiesto.

Armati di queste conoscenze, passiamo adesso ad esaminare un esempio un po' più complesso: un impianto di condizionamento d'aria.

La differenza rispetto all'esempio precedente è che in questo caso l'aria, oltre che riscaldata, può anche essere raffreddata. Occorreranno quindi due batterie di scambio acqua/aria, una alimentata ad acqua calda e l'altra ad acqua gelida.

L'afflusso di acqua ad ogni batteria è regolato da una valvola servoazionata, che varia la portata circolante da zero ad un valore massimo. Assumiamo ancora una volta che il salto termico dell'acqua sia costante, cioè che l'acqua calda entri nella batteria a 80°C ed esca a 70°C, e che l'acqua refrigerata entri a 7°C ed esca a 12°C.

Se la portata d'aria è costante, se le batterie sono calcolate per questi valori e se la temperatura dell'aria, una volta a regime, non varia molto, questa assunzione è abbastanza vicina alla realtà. Il calore ceduto o sottratto all'aria nell'unità di tempo sarà dunque:

$$Q = P \star ST \star CS$$

dove P è la portata d'acqua, ST il suo salto termico e CS il suo calore specifico, pari ad 1.

Dal punto di vista della regolazione automatica, consideriamo che esista un

senore di temperatura posto in ambiente (termostato). Esso manda la misura letta ad un regolatore. Se la temperatura scende sotto il valore di set point si apre la valvola dell'acqua calda, se sale al di sopra si apre quella dell'acqua fredda.

Ciò è visualizzato nel grafico di figura 4. La banda proporzionale la assumiamo in questo caso di tre gradi in entrambe le direzioni. Cioè una delle due valvole è tutta aperta se il valore di temperatura letto dal sensore differisce di tre gradi o più dal set point.

Per quanto detto nell'articolo precedente, per la valvola dell'acqua calda sarà:

$$APER\% = (TE - SP) / BP$$

e per quella dell'acqua fredda sarà:

$$APER\% = (SP - TE) / BP$$

Lo schema dell'impianto sarà quello di figura 5.

L'unità di trattamento dell'aria è costituita dalle due batterie e dal ventilatore. L'aria è inviata in ambiente mediante canali di lamiera, e con altri canali è ripresa e inviata all'unità di trattamento.

Poniamo ancora una volta pari a 10.000 mc il volume di aria in circolo, a 10.000 lt/h la portata max alla batteria calda, 20.000 lt/h la portata max alla batteria fredda, 10°C il salto termico del-

Figura 2 - L'intervallo chiamato banda proporzionale.

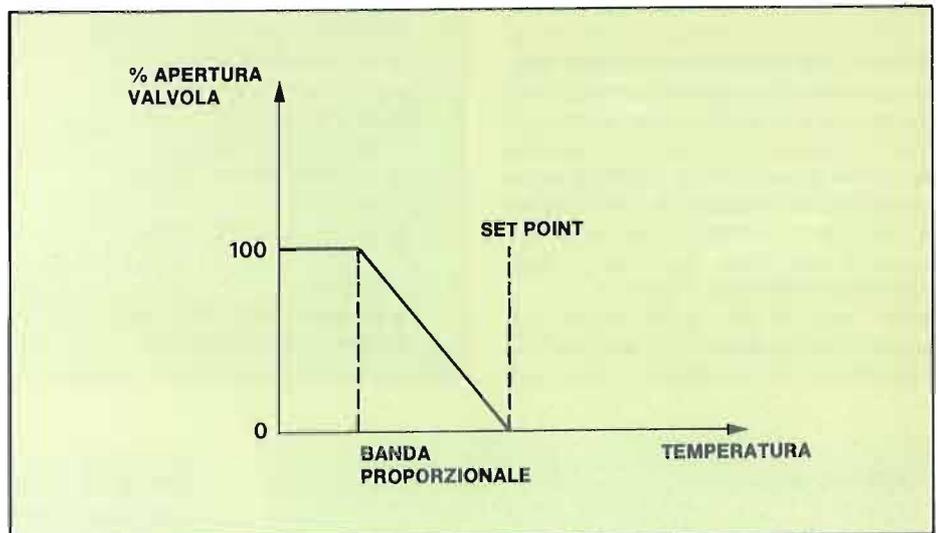


Figura 3 - La regolazione automatica adegua gradatamente la temperatura al set point.

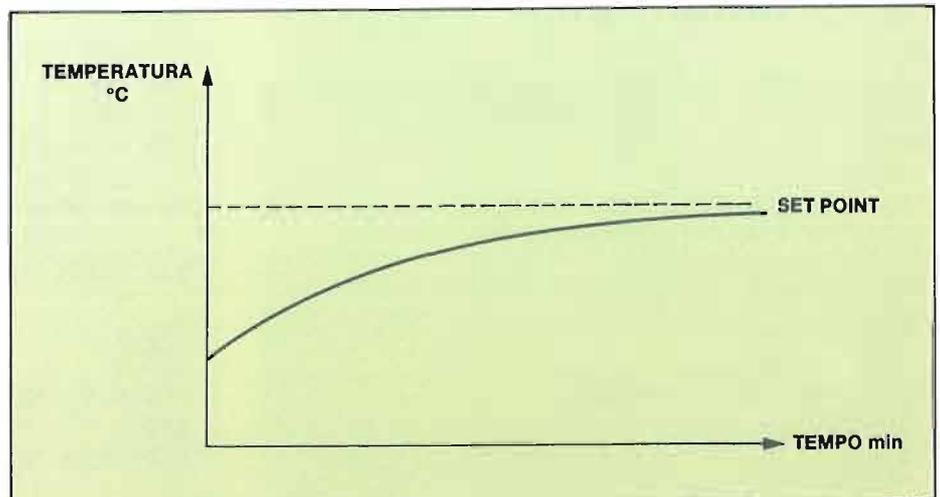


Figura 4 - Lo schema per il secondo esempio.

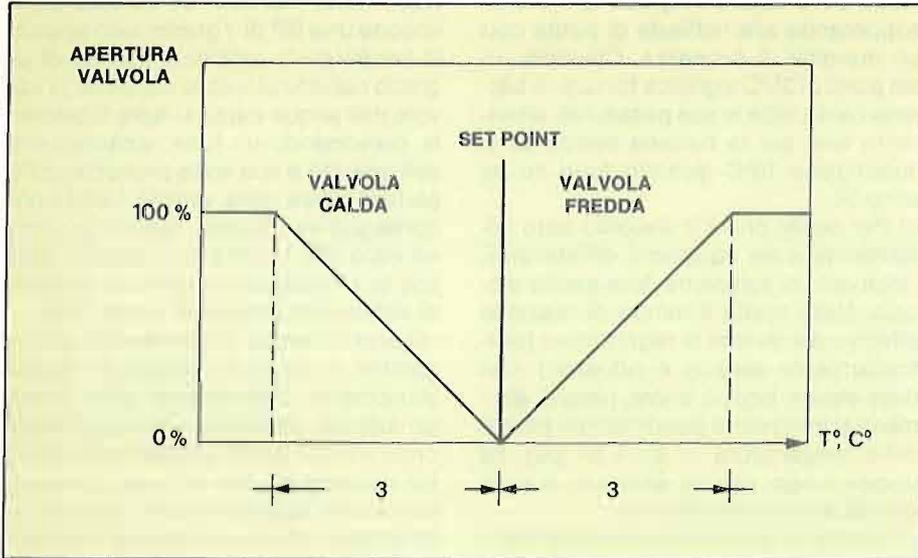
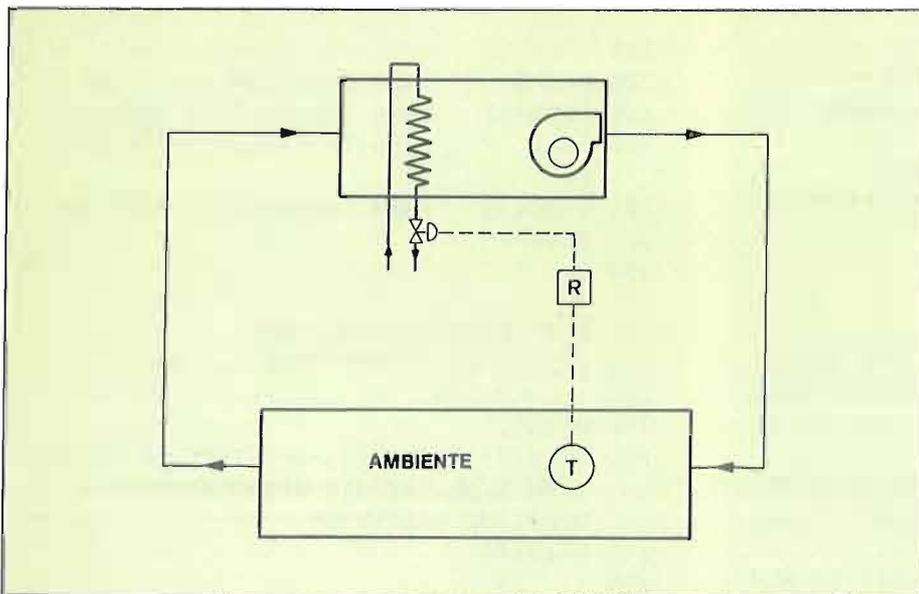


Figura 5 - La rappresentazione dell'impianto.



l'acqua calda e 5°C quello dell'acqua gelida, 20°C il set point richiesto, 3°C la banda proporzionale dei regolatori. Le dispersioni di calore verso l'esterno dipenderanno dalla differenza di temperatura fra interno ed esterno:

$$DISP = k (T. ambiente - T. esterna)$$

Esse assumono valore positivo (calore uscente dall'ambiente) per temperature esterne basse, e valore negativo (calore entrante nell'ambiente) per temperature esterne alte.

Poiché le dispersioni raggiungono il valore di progetto (CT, carichi termici) alle condizioni standard a cui sono state calcolate, (temperatura interna al set point e temperatura esterna al minimo, di solito 0°C), k dovrà valere CT/20:

$$DISP = (CT/20)(T. amb-T. est)$$

Le due batterie non lavoreranno, come è ovvio, contemporaneamente, ma in alternativa l'una all'altra, come mostrato dal grafico della regolazione (figura 4).

Quindi nelle equazioni intermedie abbiamo un branch: se la temperatura ambiente è maggiore del set point, il regolatore agirà sulla valvola dell'acqua gelida. Se è più bassa del set point, su quella dell'acqua calda. Quindi la variazione di temperatura, DTEMP, sarà negativa in un caso e positiva nell'altro. Lo schema del programma è il solito:

- 1) inserimento dati;
- 2) definizione condizioni iniziali;
- 3) ciclo, costituito da:
  - equazioni di tasso,
  - equazioni intermedie,
  - equazioni di livello,
  - stampa delle condizioni al tempo T,
  - aggiornamento dell'orologio.

Il programma continuerà il ciclo finché non verrà fermato.

Per aggiungere un tocco di realismo al tutto, visto che le dispersioni sono funzione della temperatura esterna, si introduce una routine che la fa variare ciclicamente, un po' come nelle ore del giorno, fra 0 e 35°C. In questa routine si è introdotta una variabile random, e si ha quindi una certa percentuale di casualità. La temperatura esterna diventa quindi una variabile stocastica.

Il programma completo è riportato nel listato 2; facendolo girare, si vedranno le valvole calda e fredda aprirsi, comandate dai regolatori, e la temperatura ambiente mantenersi costante (+/-2°C) malgrado il brutale variare della temperatura esterna.

Nella realtà le cose stanno un po' diversamente, perché una parte del calore sottratto dalla batteria fredda serve a condensare l'eccesso di umidità dell'aria, e quindi l'aria stessa non si raffredda quanto si potrebbe pensare. Per non dover tenere conto delle condensazioni sulla batteria fredda, dobbiamo quindi supporre che l'aria in circolo sia abbastanza secca (sotto il 50% U.R. a 25°C).

Introdurre le variazioni di umidità, anche se cosa normalissima nel calcolo degli impianti di condizionamento, significherebbe dare per note alcune cognizioni di fisica tecnica che viceversa non lo sono per tutti.

Il nostro impianto manterrà dunque soltanto la temperatura.

Con questo modello di impianto di condizionamento si potranno fare alcune semplici sperimentazioni:

- a) aumentare le dispersioni o ridurre le portate di acqua, calda e fredda, per

vedere cosa succede quando l'impianto è sottodimensionato;  
**b)** ridurre la massa d'aria trattata, per vedere cosa succede quando l'impianto è sovradimensionato;  
**c)** variare il set point, chiedendo condizioni estreme, come 35°C o 5°C;  
**d)** variare l'intervallo di soluzione IS, che è stato posto ad 1 minuto, portandolo a 5 e poi a 10 minuti;  
**e)** variare il campo proporzionale, BP. Tutte queste prove, che potranno essere fatte in pochi minuti, portano alle seguenti considerazioni, valide sia per la simulazione di sistemi che per la regolazione automatica applicata nella realtà.  
a) Come è ovvio, l'impianto deve essere correttamente dimensionato, cioè deve essere in grado di mantenere le condizioni richieste. Se ad esempio per mantenere 20°C quando fuori ce ne sono 2

occorrono 100.000 Kcal/h, la batteria calda deve essere in grado di fornirle, sopperendo alle richieste di punta con un margine di sicurezza. Chiedere un set point di 35°C significa forzare la batteria calda oltre le sue possibilità; altrettanto vale per la batteria fredda se si impongono 10°C quando fuori ce ne sono 35.

b) Per quello che è il discorso fatto inizialmente sulle equazioni differenziali, l'intervallo di soluzione deve essere piccolo. Nella realtà il tempo di reazione effettivo dei sistemi di regolazione (particolarmente sensori e attuatori) non deve essere troppo breve, perché altrimenti si innescano pendolazioni (sbalzi della temperatura in su e in giù), né troppo lungo, perché altrimenti ci si discosta troppo dal set point.

c) Variare la banda proporzionale signifi-

ca indurre un effetto della regolazione più o meno marcato. Se ad esempio si impone una BP di 1 grado, non appena la temperatura ambiente scende di un grado rispetto al valore richiesto, la valvola dell'acqua calda si apre totalmente, provocando un forte riscaldamento dell'aria: ciò a sua volta provocherà l'apertura totale della valvola fredda con conseguente brusco raffreddamento, ed ecco che ancora una volta si innescano la pendolazione, sintomo evidente di sistemi mal progettati o mal tarati.

Quindi il campo proporzionale dei regolatori, il tempo di risposta e il dimensionamento delle batterie deve essere un tutt'uno armonico e proporzionato, onde evitare grossi problemi durante il funzionamento. Per fortuna, il sistema può essere facilmente collaudato prima della costruzione mediante la tecnica di

#### Listato 2 - Il secondo programma.

```

100 REM *****
105 REM * SIMULAZIONE DI UN IMPIANTO *
110 REM *
115 REM * DI CONDIZIONAMENTO *
120 REM *****
130 :
150 REM DATI
160 REM ****
165 :
170 P1MAX=10000:REM PORTATA ACQUA CALDA
175 P2MAX=20000:REM PORTATA ACQUA FREDDA
180 ARIA=10000 :REM QUANTITA' ARIA IN CIRCOLO
190 CT=100000 :REM DISPERSIONI AMBIENTI
200 C1S =0.3 :REM CALORE SPECIFICO ARIA
210 C2S =1 :REM CALORE SPECIFICO ACQUA
211 SP =20 :REM SET POINT TEMPERAT. AMBIENTE
212 BP =3 :REM BANDA PROPORZIONALE
213 S1A=10 :REM SALTO TERMICO ACQUA CALDA
214 S2A =5 :REM SALTO TERMICO ACQUA FREDDA
220 :
221 :
222 REM CONDIZIONI INIZIALI
223 REM *****

```

```

224 :
225 MIN=0 :REM CONTATORE DEL TEMPO
226 TAMB=30 :REM TEMPERATURA AMBIENTE
227 IS=1 :REM INTERVALLO DI SOLUZIONE
228 TEST=35 :REM TEMPERATURA ESTERNA
229 SEGNO=-1
230 :
231 :
234 REM INIZIALIZZAZ.STAMPA
235 REM *****
236 :
240 OPEN4,4
242 PRINT#4,"MINUTI{3 SPC}TEMP. ESTERNA{2 SPC}%VAL.CALDA{2 SPC}%VAL.FREDDA{7 SPC}TEMP.AMBIENTE"
245 PRINT#4
248 :
249 :
250 REM EQUAZ. DI TASSO
260 REM *****
265 :
270 A1PER=(SP-TAMB)/BP
280 IF A1PER >1 THEN A1PER=1
290 IF A1PER <0 THEN A1PER=0
292 :
295 A2PER=(TAMB-SP)/BP
296 IF A2PER >1 THEN A2PER=1
297 IF A2PER <0 THEN A2PER=0
298 :

```

## COMMODORE 64

simulazione ora esposta. Poiché essa è uguale per tutti i sistemi continui e dinamici, il lettore dovrebbe già essere in possesso delle armi necessarie ad affrontare la maggior parte dei problemi che gli si presenteranno.

A conclusione di questa serie di articoli, speriamo non troppo noiosi, si possono trarre le seguenti conclusioni: ogni volta che in un "sistema", inteso come insieme di parti interagenti isolato da influenze esterne, avvengono dei fenomeni descrivibili quantitativamente mediante una serie di equazioni che mettono in relazione le grandezze coinvolte (= modello matematico), è possibile simulare al computer il sistema stesso, al fine di studiarne il comportamento "in vitro".

E ciò anche se il sistema comprende, come succede quasi sempre, grandez-

ze variabili nel tempo e regolazioni automatiche.

Ciò porta a straordinari vantaggi sia nella comprensione del sistema stesso, che nella possibilità di previsione del suo comportamento e nella sua ottimizzazione, vantaggi che non si potrebbero ottenere per altra via.

In altre parole, la simulazione consente di usare il computer come mezzo di analisi della realtà.

## Bibliografia

- Daboni et al. - "Ricerca operativa" - Zanichelli 1982.
- Forrester - "Principi dei sistemi" - Etas Kompass 1974.
- Ahl - "Computer in mathematics" - Creative Computer press 1979.
- Wolfram S. - "Software nella scienza

e nella matematica" - su "Le Scienze" Novembre 1984.

- Cesari F. - "Regolazione automatica negli impianti industriali" - F. Angeli 1981.
- D'Amato V. - "Introduzione all'analisi dinamica dei sistemi e alla simulazione con il computer" - F. Angeli 1985. ■

```

299 :
300 REM EQUAZIONI INTERMEDIE
305 REM *****
310 :
320 DISP=CT*(TAMB-TEST)/20
325 : REM BRANCH
330 IF TAMB >SP THEN 360
335 REM FALSE
340 : PCALDA=A1PER *P1MAX
342 : QC=PCALDA*S1A*C2S
344 : QC=QC-DISP
346 : QC=QC/60
350 : DTEMP = QC/(C1S*ARIA)
355 GOTO390
360 REM TRUE
365 : PFR=A2PER *P2MAX
367 : QF=-PFR*S2A*C2S
370 : QF=QF-DISP
375 : QF=QF/60
380 : DTEMP = QF/(C1S*ARIA)
390 REM END BRANCH
395 :
397 :
400 REM EQUAZIONI DI LIVELLO
410 REM *****
415 :
420 TAMB=TAMB+DTEMP*IS
450 :
460 :
500 REM STAMPA

```

```

510 REM *****
515 :
520 PRINT#4,CHR$(16)CHR$(48)CHR$(0)MIN;
530 PRINT#4,CHR$(16)CHR$(49)CHR$(0)INT(T
EST*100)/100;
540 PRINT#4,CHR$(16)CHR$(51)CHR$(0)INT(A
1PER*100);
550 PRINT#4,CHR$(16)CHR$(52)CHR$(0)INT(A
2PER*100);
560 PRINT#4,CHR$(16)CHR$(54)CHR$(0)INT(T
AMB*100)/100
590 :
600 :
605 REM AGGIORNAMENTO OROLOGIO
606 REM *****
607 :
610 MIN=MIN+IS
650 :
660 :
1000 REM ROUTINE VARIAZIONE CASUALE TEMP
ERAT. ESTERNA
1010 REM *****
*****
1020 :
1030 TEST=TEST + INT(3*RND(1))*SEGNO
1040 IF TEST >33 THEN SEGNO=-1
1050 IF TEST<2 THEN SEGNO=1
1060 :
1070 :
2000 GOTO250

```

# PERSONAL MARKET

## Apple

Vendo **Apple II plus originale, perfetto**, con scheda language 16 Kbyte, scheda controller per due drive, drive originale Apple a L. 1.280.000. Superoccasione! Roberto Rossi - Via Lario, 26 - 20159 Milano - Tel. 02/6070236

Vendo **Apple IIc unità centrale**, monitor, stampante, drive, programma Applework. Mai usato, L. 3.000.000 con fattura. Telefonare ore ufficio. Lorenzo Frigerio - Via G. Sacchi, 7 - 20121 Milano - Tel. 02/837556

## Commodore

Vendo **Commodore 16 con registratore**, joystick e molti giochi più riviste a L. 350.000. Usato pochissime volte, causa passaggio a Plus/4. Gianni Darra - Viale Ungheria, 11 - 20138 Milano - Tel. 02/503157

Vendo **plotter 1520 Commodore** a L. 350.000 trattabili; scheda - cartuccia CP/M per C64 e che abbiano buone idee da realizzare. Parliamone insieme! Enrico - Milano - Tel. 02/8350804

Cerco persone **che si dedicano allo sviluppo di software in linguaggio macchina per C64** e che abbiano buone idee da realizzare. Parliamone insieme! Enrico - Milano - Tel. 02/8350804

Vendo **VIC 20 con manuale** e 5 cassette di videogiochi a L. 99.000. Telefonare ore pasti. Aimone Castelli - Via Enrico Fermi, 11 - 22070 Guanzate (CO) - Tel. 976375

■ Vendo **Commodore 710 128 Kbyte RAM** più ■  
■ dual drive 8250LP 2 Mbyte in linea, amplificatore ■  
■ BF, 6 libri, programmi vari. Il tutto a L. ■  
■ 5.000.000. Telefonare a: Gianni Dal Checco - ■  
■ Via S. Giacomo, 106 - 39050 S. Giacomo di ■  
■ Laives (BZ) - Tel. 0471/941508 ■

Vendo **printer/plotter 1520 Commodore** e software dedicato a L. 290.000. Tutto in perfette condizioni. Dario Galbati - Via Fleming, 8/A - 20059 Vimercate (MI) - Tel. 039/681042

Cambio, vendo, compro **software per C64**. Mi interessa il programma Oroscofo solo se di ottima qualità. Scrivere o telefonare a: Cav. Uff. Livio Modena - Via Dante Alighieri, 110 - 18038 Sanremo (IM) - Tel. 73079

Vendo **Commodore SX-64 (di recente acquistato) con 12 mesi di garanzia** e tutto il miglior software con manuali (Easy Script, Calc Result, Flight Simulator, ecc.) a L. 1.800.000. Carlo Vitti - Via Argentina, 24 - 27029 Vigevano (PV) - Tel. 0381/72049

Compro **manuali in italiano per il C64**. Inviare liste e proposte a: Antonio Agus - Via S. Antonio, 31 - 09016 Iglesias (CA)

Sono disperatamente alla ricerca di un **modem per C64**. Michele Petracca - Via Donatello, 12 - 35027 Noventa Padovana (PD) - Tel. 049/627164

Vendo per C64 **programma per rappresentanti con calcolo provvigioni** e segnalazione clienti da visitare. Per ulteriori informazioni telefonare ore serali. Calcedonio Meli - Via G. Amico Valenti, 89 - 93100 Caltanissetta - Tel. 0934/23365

Causa passaggio ad altro sistema, vendo **Plus/4 con un mese di vita, ancora in garanzia**, più registratore, joystick, programmi e manuali. Prezzo da stabilire. Alberto Cristofari - Via Pastoris, 94 - 13043 Cigliano (VC) - Tel. 0161/44263

Vendo **drive 1541 nuovo modello (quello con la chavetta) mai usato**, garanzia ancora da spedire, a prezzo interessante. Franco Fantoni - P.O. Box 259 - 51100 Pistoia - Tel. 0573/2204

Vendo **VIC 20 più registratore**, consolle Intellevison, 4 cassette a L. 235.000. Cambio con C16 o ZX Spectrum 16/48 Kbyte. Vendo in blocco o singolarmente. Telefonare ore pasti. Nicola Portas - Via Molise, 14 - 09100 Cagliari - Tel. 070/490697

Vendo **Plus/4 più drive nuovissimi**. Telefonare ore ufficio. Rocco Ciliberti - Via Marchesi, 10 - 20067 Paullo (MI) - Tel. 02/7490865

Se il tuo Commodore 64 è guasto, lo **posso riparare in meno di una settimana**. Inoltre acquisto C64 danneggiati per riutilizzo componenti. Per informazioni: Sergio Fiorentini - Corso Genova, 7 - 20123 Milano - Tel. 02/8350804

## Sinclair

Cerco possessori di Spectrum 48 Kbyte **che possano mandarmi qualche programma per radioamatori**. Adriano Susta - Via Ressi, 23 - 20125 Milano - Tel. 02/6882478

Vendo **4 riviste Sinclair e un libro di programmi** per ZX81 con relativa cassetta (6 programmi) a sole L. 18.000. Pagamento contrassegno. Giorgio Garofalo - Via Angeloni, 40 - 20161 Milano - Tel. 02/645377

Vendo **ZX Spectrum 48 Kbyte con alimentatore stabilizzato** più ZX printer, giochi e manuali inglesi; il tutto per la ridicola somma di L. 300.000. Tratto solo con la zona di Firenze. Giacomo Covoni - Via del Sansovino - 50142 Firenze - Tel. 055/711839

Vendo **sistema Spectrum 48 Kbyte** più stampante Seikosha GP50, interfaccia parlante: come nuovo, corredato di 100 programmi vari a L. 750.000 trattabili. Scrivere oppure telefonare ore serali. Riccardo Nicoletti - Via della Cernaia, 3 - 50129 Firenze - Tel. 055/48325

## Texas

Vendo **TI 99, interfaccia parallela, stampante GP500A**, Extended BASIC, mini memory, Multiplan, word processor, box con driver ed altro ancora. Il tutto con manuali e moduli SSS, in blocco oppure separatamente. Enrico Chiesa - Via Bruere, 318 - 10098 Rivoli (TO) - Tel. 011/9595796

Vendo **TI 99 con manuali, cavetto per registratore**, coppia joystick, moduli Extended BASIC, Minimemory, Scacchi e Wumpus; manuale Editor/Assembler e libri vari a L. 300.000. Massimo Frigento - Via Graziano, 6 - 00165 Roma - Tel. 06/6222192

Vendo per TI 99/4A **Extended BASIC nuovissimo con manuale in italiano** a L. 190.000 più spese postali. Inoltre vendo joystick nuovi a L. 40.000 più spese postali. Marco Ballerini - Via P. Mariano Colagrossi, 35 - 00122 Casal Palocco (RM) - Tel. 5651186

Vendo **TI 99/4A più Extended BASIC con manuale**, box espansione, 32 Kbyte RAM, disk drive, disk control, 3 libri in italiano sul TI 99, garanzie da compilare e 300 programmi su 30 floppy al miglior offerente. Giovanni Binda - Via Privata Volta, 14/i - 22054 Mandello Lario (CO) - Tel. 0341/735841

Per TI 99/4A compro **coppia di joystick in buono stato**; inoltre compro giochi anche in Extended BASIC. In particolare cerco i giochi Sci, Invaders, Camelot. Sono disposto a rispondere a tutti. Mandatemi la vostra lista con prezzi. Claudio Famelli - Via Leopoldo Bignone, 27/5 - 16157 Genova-Pra (GE) - Tel. 010/663998

Vendo modulo **Extended BASIC più manuale** e cassali. Vendo registratore, cassetta, moduli SSS Calcio e Parsec, joystick causa passaggio a sistema superiore. Michele Tossuti - Via Fogazzaro, 21 - 30172 Mestre (VE) - Tel. 041/93628



Vendo per TI 99/4A **Editor Assembler in versione mini-memory**: sono richiesti espansione 32 Kbyte e disk drive. Il programma consente la creazione di file sorgenti e oggetto compatibili Texas. Telefonare ore 20.00-21.00. Sebastiano - Milano - Tel. 02/270759

Cambio, vendo **fantastici giochi, utility ed altro** in BASIC o BASIC esteso per TI 99/4A. Per informazioni o per ricevere la lista dei programmi scrivere a: Antonio Sisto - Via Biancardi, 6 - 20075 Lodi (MI)

Per TI 99/4A vendo **peripheral box, disk controller, disk drive 32 Kbyte RAM, manuali, Extended BASIC, 140 programmi su dischetto**. Vendesi anche separatamente a prezzo trattabile. Valerio Paponi - Via Caffarella, 18 - 0040 Tor San Lorenzo (RM) - Tel. 06/914057

## Varie

Vendo **stampante Epson TX80 mal usata**, ingresso Centronics, completa di cavo collegamento a computer, più interfaccia RS232 per Sharp MZ80K, tutto a L. 600.000. Angelo Angeloni - Via D.G.M. Russolillo, 27 - 00138 Roma - Tel. 06/8401552

Compro **joystick rotti o inutilizzabili** (solo zona Roma); inoltre, a richiesta, monto micro-switch per joystick Quickshot II (Spectravideo). Telefonare ore pasti. Stefano Baldoni - Via della Magliana, 277 - 00146 Roma - Tel. 06/5271210

Cambio **console Atari VCS 2600** più 5 cassette con floppy disk 1541. Inoltre vendo moltissime riviste e libri. Chiedere elenco. Giuseppe Corazza - Via Torretta, 22 - 24100 Bergamo

Cambio, vendo **base Mattel/Intellelevision con 15 cartucce** e con i comandi Point Master e Quick Stick a L. 650.000. Oppure cambio con C64 più registratore e joystick. Demetrio Triglizzi - Via Nazionale, 59 - 64037 Cermignano (TE) - Tel. 0861/66170

Vendo **Sharp MZ-700 come nuovo** completo di stampante plotter 4 colori, programmi, videogiochi, a L. 800.000. Telefonare ore serali. Stefano - Tel. 0161/68845

Vendo **Colecovision più alimentatore, joystick, modulo Turbo**, le cartucce Mousetrap, Carnival, Turbo a sole L. 250.000 trattabili. Annuncio sempre valido. Francesco Saporito - Via Cifali, 27 - 95123 Catania - Tel. 095/446581

Per computer MSX vendo **cartuccia gioco "Sparkle"** a L. 15.000 anziché L. 55.000. Emanuele Costa - Via Nino Costa, 8 - 10045 Piossasco (TO) - Tel. 011/9064223

Vendo **Altos ACS 8000, 2 floppy 500 Kbyte 8" SFDD**, più 64 Kbyte RAM, CP/M, MBASIC, Hazeltine 1420, stampante TX80B a L. 4.000.000, compresi manuali, cavi e Totocalcio Computerizzato, BASIC e W.P. Carlo Montecchi - Viale Belvedere, 54/a - 41028 Serramazzoni (MO)

Vendo **Sharp MZ-731, plotter e registratore incorporati**, predisposto grafica 320x200 e CP/M 2.2, con ampio software, a L. 700.000. Antonio Moro - Res. Parco MI 2, 161 - 20090 Segrate (MI) - Tel. 02/215445

Vendo **Triumph Adler PC, 2 mesi di vita, con floppy**, video Philips, CP/M, WordStar, MBASIC, Logical, BASIC Interpreter, BASIC Compiler, giochi e programmi vari. Vasta documentazione, L. 2.000.000. Roberto, Tel. 02/2899763 (ore ufficio)

Vendo **Micro N.E. in configurazione massima** (64 Kbyte RAM, scheda grafica) più 1 drive, contenitore per 2 drives, monitor 12" fosfori verdi, rack nero. Possibilità di fattura. Tratto possibilmente con la Toscana. Claudio Amerizzi - Via Dante, 13 - 50050 Limite sull'Arno (FI) - Tel. 0571/57404 oppure 57664

Concessionaria Pubblicità



Agenzia per le tre venezie

**Pier Filippo OBBER**  
Via Nazionale, 70  
38070 IMER (TN)  
tel. 0439/67221

J. ADVERTISING s.r.l.

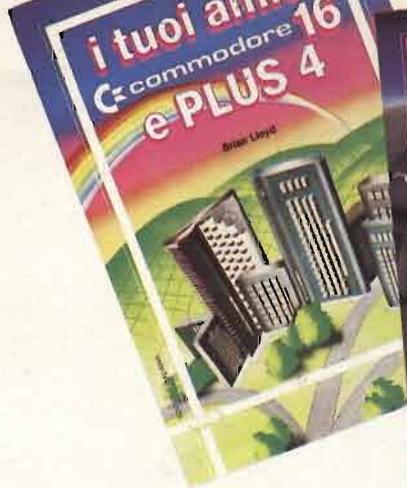
Sede Legale, Direzione e Amministrazione:  
Viale F. Restelli, 5 - 20124 Milano  
Telefono (02) 6880606 - 6085941 r.a.  
Telex 316213 REINA

**GRUPPO EDITORIALE JACKSON**

**LA BIBLIOTECA CHE FA TESTO.**



# dalla biblioteca Jackson informatica per tutti



**Rita Bonelli,  
Luciano Pazzucconi,  
Fabio Racchi**  
**COMMODORE 16:  
SEMPRE DI PIÙ**  
Un libro sul Commodore 16 per approfondire le conoscenze sulla macchina e sul suo BASIC.  
**cod. 427B Pag. 336  
Lire 35.000 Con cassetta**

**David Lawrence**  
**TECNICHE  
DI PROGRAMMAZIONE  
SUL COMMODORE 64**  
L'arte della buona programmazione alla portata di chiunque possieda un Commodore 64.  
**cod. 575D Pag. 176  
Lire 16.500**

**Daria Gianni, Carlo Tognoni**  
**MSX: IL BASIC**  
Il primo libro sul BASIC MSX,

che unisce le caratteristiche di un manuale di riferimento a quelle di un buon testo didattico di programmazione.  
**cod. 417D Pag. 216  
Lire 20.500**

**Brian Lloyd**  
**I TUOI AMICI COMMODORE 16  
E PLUS 4**  
Anche i computer hanno un cuore: impara a programmare con i tuoi amici C16 e Plus 4.  
**cod. 423B Pag. 168  
Lire 16.000**

**Rodnay Zaks**  
**IL TUO PRIMO COMPUTER**  
Una semplice introduzione al mondo dei personal orientata ad utenti alla loro prima esperienza con il computer.  
**cod. 351D Pag. 240  
Lire 25.000**

ritagliare (o fotocopiare) e spedire in busta chiusa a:  
**GRUPPO EDITORIALE JACKSON - Divisione Libri - Via Rosellini, 12 - 20124 Milano**  
**CEDOLA DI COMMISSIONE LIBRARIA**

**VOGLIATE SPEDIRMI**

n° copie	codice	Prezzo unitario	Prezzo totale
Totale			

Pagherò contrassegno al postino il prezzo indicato più L. 3.000 per contributo fisso spese di spedizione.

**Condizioni di pagamento con esenzione del contributo spese di spedizione:**

Allego assegno della Banca

Allego fotocopia del versamento su c/c n. 11666203 a voi intestato

n° \_\_\_\_\_

Allego fotocopia di versamento su vaglia postale a voi intestato

Nome \_\_\_\_\_

Cognome \_\_\_\_\_

Via \_\_\_\_\_

Cap \_\_\_\_\_ Città \_\_\_\_\_

Prov. \_\_\_\_\_

Data \_\_\_\_\_ Firma \_\_\_\_\_

Spazio riservato alle Aziende. Si richiede l'emissione di fattura

ORDINE  
MINIMO  
L. 50.000

Partita I.V.A. \_\_\_\_\_



**GRUPPO  
EDITORIALE  
JACKSON**

# ARTWORK ENVIRONMENT

## l'immaginazione informatica



## ARTWORK ENVIRONMENT

**ARTWORK ENVIRONMENT, una gamma di prodotti software (artwork, brushwork, chartwork), perfettamente integrati, per creare disegni, manipolarli, trasformarli avvalendosi di una varietà cromatica che può raggiungere 16 milioni di colori.**

**ARTWORK ENVIRONMENT, visualizzazione contemporanea anche di 256 colori** che possono operare per combinazioni, sovrapposizione, sfumature; la loro apparizione, secondo sequenze desiderate, produce effetti di animazione del disegno.

**ARTWORK ENVIRONMENT, grafica con immagini riprese dalla telecamera** trasferite direttamente al computer, ed abbinamento a testi (15 caratteri disponibili) e disegni.

**ARTWORK ENVIRONMENT, memorizzazione del lavoro creativo.** Il prodotto finale può essere ottenuto su stampante, registrato su nastro, con videoregistratore collegato al computer, riprodotto su hard-copy RGB direttamente in diapositiva.

**ARTWORK ENVIRONMENT, perchè Artwork, Brushwork, Chartwork funzionano in un ambiente integrato, scambiandosi le immagini create e fornendosi reciprocamente le migliori che ciascun prodotto è in grado di offrire.**

**ARTWORK, per disegnare** elementi e figure spesso molto complesse, utilizzando diversi fogli fra di loro sovrapponibili.

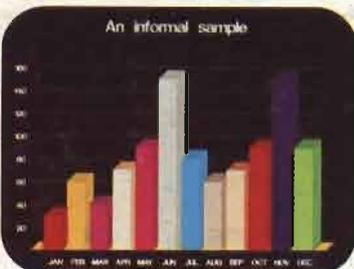
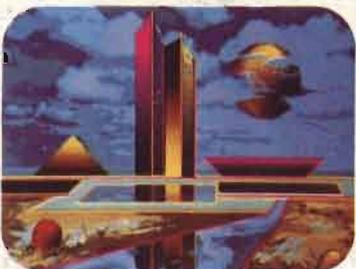
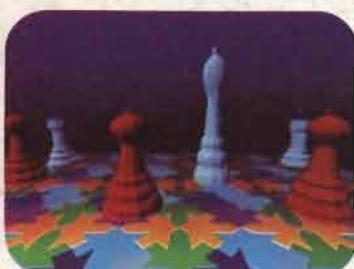
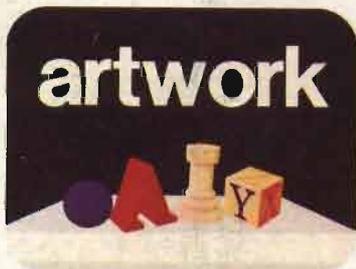
**ARTWORK, grafica tridimensionale** per creare solidi, rimuovere linee nascoste, ottenere prospettive ed ombreggiature anche in funzione delle sorgenti di luce.

**BRUSHWORK, disegno pittorico,** creato a mano libera ed elaborato a video con una varietà illimitata di pennelli, tratti ed ombreggiature.

**CHARTWORK, grafica manageriale** per rappresentare i dati sotto forma di aree, diagrammi a barre, a torte od altre figure costruite dall'utente.

**ARTWORK ENVIRONMENT su Personal Computer:** funziona su PC IBM e compatibili, con minimo 256 K di memoria, e schede grafiche diverse.

**ARTWORK ENVIRONMENT, l'immaginazione informatica a disposizione di pubblicitari, stilisti, designers, tecnici video:** uno strumento di ricerca, di visualizzazione, di memorizzazione, di facile e rapida applicazione.



Siamo presenti allo SMAU '85 dal 19 al 24 settembre 1985  
Pad. 13 Salone 1 Posteggi D33 - E08+ E06

**Sirio**  
Informatica

distributore nazionale di  
ARTWORK ENVIRONMENT  
AUTOCAD  
RAMTEK

20156 Milano  
viale Certosa 148  
tel. (02) 3010051 (5 linee)  
telex: 323380 SIRIO I

24100 Bergamo  
via Angelo Maj 16/B  
tel. (035) 223552

21052 Busto Arsizio  
via Daniele Crespi 1  
tel. (0331) 622274