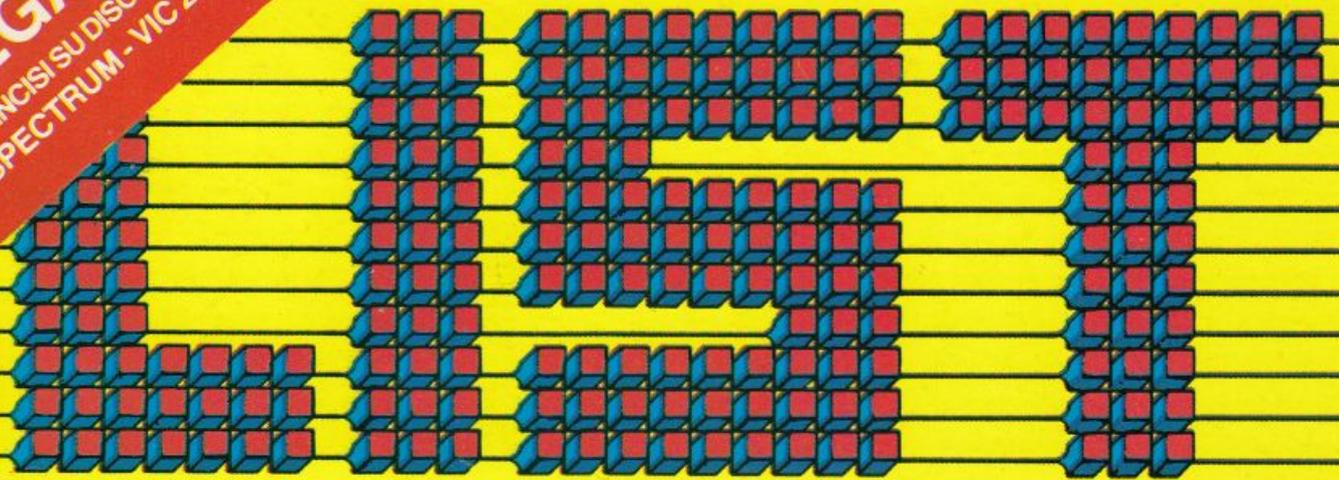


IN REGALO

3 PROGRAMMI INCISI SU DISCO A 33 GIRI
PER IL TUO SPECTRUM - VIC 20 - TEXAS



RIVISTA MENSILE ANNO 3° - N. 5 - MAGGIO 85 - LIRE 5000 - SPED. ABB. POST. GRUPPO III 70%

PROGRAMMI PER IL TUO HOME COMPUTER

Disc-O-LIST

Sinclair
Spectrum

VIC-20

TI-99/4A

BASF VERO CHROMO. PRIMO PIANO SUL DOMANI



BEAN s.n.c.
AGENZIA PER IL LAZIO

Via C. Rasponi, 19
00162 Roma
Tel. (06) 83.23.298 - 83.23.323



BASF
AUDIO VIDEO

SASEA S.p.A.
Via V. da Seregno, 44
20161 Milano - Tel. (02) 6408

PROGRAMMI PER IL TUO HOME COMPUTER ANNO III - N. 5 - MAGGIO 1985 - MENSILE DI INFORMATICA

Giuseppe Breveglieri - **Direttore responsabile**
Luciano Ciardelli - **Direttore editoriale**
Liana Pirone - **Segretaria di redazione**
Hanno collaborato a questo numero:
Antonella Corica, Contenti Osvaldo, Maurizio Giunta,
Massimo Truscelli, Massimo Momo,
Paolo Ciancarini, Russo Fabrizio

Progetto grafico e impaginazione

Romeo Guaricci

Progetto copertina

Studio Grafico PRAXIS

Fumetto

M. Cossu, A.M. Marzi

Stampa

LE.GRAF G.E. Rizzo 18 - Roma - Tel. 7970770

Fotocomposizione

Studio Grafico C.R. S.r.l. - Roma - Tel. 6111652

Distributore esclusivo per l'Italia

Parrini & C.

Piazza Indipendenza 11/B - Roma

Editrice

EDICOMP s.r.l.

Via C. Colombo 193

00147 Roma - Tel. 7665495

Pubblicità

EDICOMP S.r.l.

Viale dell'Esperanto, 71

00144 Roma - Tel. 06/5918895

Redazione

Via Flavio Stilicone 111

Roma - Tel. 06/7665495

Inserzionisti

BASF audio video SASEA S.p.A.

Via V. da Seregno, 44

20161 Milano - Tel. 02/6408

MOVIT - TEXIM ITALIA

V.le dell'Esperanto, 71

00144 Roma - Tel. 06/5818939

SOPIN S.p.A.

Via del Serafico, 200 - Roma

pag. 3	Editoriale
	Rubriche
2	Recensioni - Software games
4	Posta
7	News & News
8	Le novità
10	Atene senza schiavi
14	Tsukuba - La città di Mazinga e del treno Magnetico
17	Scuola e Computer
19	I Speak Logo
22	Z80: linguaggio macchina
26	Le pagine dell'assistenza
28	Trucchi e scoperte
30	Didattica
31	Handbook
50	Disco List
89	Concorso Strike - cartolina voto
	VIC 20
33	Keyboard Codes
58	Calorie
	CBM 64
34	Geometria 1
60	Sort 1
74	Arredatore elettronico
82	Duello aereo
	SEGA SC 3000
37	Grafica 3D
62	Decisioni
	Sharp MZ-700
40	Caccia all'U-Boote
64	Analisi
	Texas TI 99/4A
43	Disegnare con il computer
66	Gioco dei fiori
80	Caccia alla base
	ZX 81 + 16K
45	Supersnake
	Spectrum 16/48K
46	Il gatto e il topo
68	Flush
79	Parco Pubblico
	ORIC - 1
48	Oric sequencer
70	Equazioni di 1° e 2° grado
72	Semina
	MPF II
52	Black Jack
91	Fumetto: L'amico segreto

Prezzo di un numero: L. 5.000 - Numero arretrato: L. 7.000 - Abbonamento: annuo L. 55.000. Per l'estero: L. 110.000 - I pagamenti vanno effettuati a mezzo c/c bancario, vaglia postale, c/c postale n. 72609001 intestato a LIST programmi per il tuo home computer Casella postale 4092 ROMA APPIO.
Per i cambi di indirizzo allegare alla comunicazione l'importo di L. 500, anche in francobolli, e indicare insieme al nuovo anche il vecchio indirizzo.

© Tutti i diritti di riproduzione, anche parziale, del materiale pubblicato sono riservati. Manoscritti, listati, bozzetti e fotografie anche se non pubblicati non si restituiscono. La Direzione declina ogni responsabilità in merito alla originalità, alla provenienza ed alla proprietà dei programmi pubblicati. Per ogni controversia è competente il Foro di Roma.

Autorizzazione Tribunale di Roma n. 254 del 3-8-1983.



J. Boisgontier e S. Brebion
Il BASIC per tutti
 Iniziazione e programmi
 EPSI Edizioni Milano
 Prezzo L. 16.000

Il BASIC è certo la croce e la delizia dei nostri lettori, sin dal primo momento dell'acquisto del loro computer personale. Non sappiamo quanti tra voi siano in completa confidenza con istruzioni come OPEN o GOSUB, ma certo la maggior parte di voi si sarà accorta che, mentre è facile capire il senso delle trenta o quaranta parole chiave che costituiscono il vocabolario BASIC, ben più difficoltoso risulta riuscire a metterle insieme per produrre programmi complessi e sensati. Questo libro, concepito per principianti nel campo della programmazione, vi invita a sedervi davanti al vostro computer ed a ... parlargli in BASIC. Prima frasi semplici, d'assaggio: istruzioni di stampa, test, loop. Poi vi verranno proposti problemi via via più complessi, ampiamente commentati e risolti. È concesso molto spazio allo studio delle varie istruzioni grafiche indispensabili nella stesura di quasi tutti i programmi «seri», siano essi gestionali, didattici, o perché no, videogiochi. Il BASIC scelto è l'Applesoft, che è forse uno dei più generali e facili da tradurre su elaboratori di altre marche.

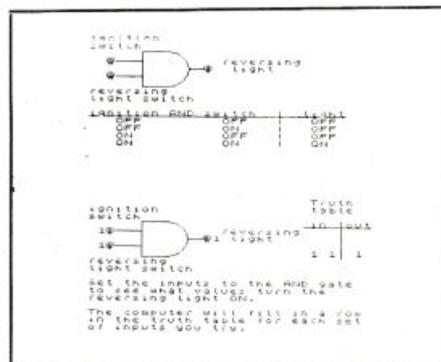


J.P. Blanger
Modelli di espressione grafica
 EPSI Edizioni Milano
 Prezzo L. 20.000

Questo volume è stato pensato come un'introduzione alle idee ed alle tecniche della Computer Graphic. Come i nostri lettori ben sanno, per Computer Graphic (che come disciplina comincia ad esser nota anche sotto il nome di Eidomatica) si intende tutta una serie di applicazioni degli elaboratori elettronici nel campo della grafica: dai videogiochi sino alla sintesi d'immagini elettroniche.

Questo libro è un testo introduttivo che presenta i concetti base di questa disciplina, studiati ed applicati in modo graduale, in special modo ai personal computer Apple, Commodore (Pet) e TRS-80. L'approccio, notevolmente originale, incoraggia l'autoapprendimento nonché l'uso immediato delle tecniche studiate. Tra i temi trattati troviamo, oltre agli elementi di base della Computer Graphic (come ad esempio la rappresentazione vettoriale e la grafica in alta risoluzione), la rappresentazione di equazioni complesse, le trasformazioni e le traslazioni dinamiche di poligoni e superfici (chi ha visto la stupenda sigla della trasmissione televisiva Quark, condotta da Piero Angela, può farsi un'idea di ciò di cui stiamo parlando), ed infine le applicazioni grafiche complesse interattive.

Il linguaggio scelto per la presentazione dei programmi è il Basic dell'Apple, che è facilmente adattabile ad altri computer. Sono anche presentati molti esempi già tradotti in BASIC Commodore.



MAKE A CHIP

Finalmente un programma per lo SPECTRUM che, anche se in maniera semplice, permette un approccio più chiaro alla logica del computer. Il programma MAKE A CHIP offre una buona panoramica sul come funziona un chip digitale e quindi sugli stati logici presenti sulla sua uscita in relazione ai livelli presenti, alle porte di entrata, a seconda che il chip sia di tipo AND, OR, NOT, NAND, NOR. Oltre ad una sufficientemente chiara spiegazione contenuta nella prima parte, il programma comprende anche alcune esercitazioni animate che approfondiscono i comportamenti delle varie porte logiche ed in ultimo permette di poter costruire anche circuiti che sfruttino i vari chip e le varie possibilità di connessione verificandone così anche il funzionamento con opportune tavole della verità. Unico appunto che si può rivolgere è purtroppo la scarsa comprensione per coloro che non conoscono la lingua inglese, anche se la discreta grafica può aiutare molto. Bisogna dire che giunge con un certo ritardo in Italia specialmente se si tiene conto che il Copyright della Software house INCOGNITO risale al 1983.



EDITORIALE

Il problema della disoccupazione in Europa tormenta ormai da tempo gli equilibri economici mondiali e viene paragonato, apertamente, a quello ancor più drammatico dell'indebitamento dei paesi dell'America Latina.

Nei paesi della Comunità Europea i senza lavoro erano, negli anni sessanta, appena il 2% della popolazione attiva. In Europa, i disoccupati, sono oggi l'11,2% pari a 12 milioni e 650 mila persone; un vero esercito composto per il 40% da giovani tra i 14 e i 25 anni. Negli Stati Uniti, invece, il tasso di disoccupazione è sceso, negli ultimi quattro anni, dal 12% al 7,5% e in Giappone dal 3,5% al 2,5%.

Le cifre ci dicono, dunque, che in Europa il problema è drammatico, mentre in aree tecnologicamente più avanzate, come USA e Giappone, sono state trovate soluzioni adeguate. Lo squilibrio tra domanda e offerta di lavoro, in Europa, continuerà anche nel prossimo decennio, poi, soprattutto in Italia dove i problemi riguarderanno i giovani in cerca della prima occupazione, le cose miglioreranno anche per motivi di carattere demografico.

Ma è chiaro che non è possibile far trascorrere un decennio in attesa che le cose si aggiustino da sole, senza che politici, economisti e sindacalisti, prendano decisioni a livello nazionale ed internazionale.

La molla da utilizzare per avviare progetti, per creare posti di lavoro è certamente quella dell'innovazione tecnologica. Su questo argomento, a metà Marzo, si è svolta a Venezia una conferenza di grande rilevanza politico-sociale ed economica alla quale hanno partecipato i rappresentanti di 22 paesi dell'area occidentale. Primi Ministri, Ministri del Lavoro, della Ricerca e dell'Economia di questi paesi, hanno portato il loro contributo al tema dell'incontro: «Sviluppo tecnologico e occupazione». A grandi linee la conferenza veneziana ha detto tra l'altro un paio di cose molto importanti. Per cercare di dar vita ad un programma realistico, in grado di risolvere il problema, bisogna affrontare due nodi principali: il consenso e la cooperazione internazionale. Per trovare il consenso

delle varie componenti impegnate nello sviluppo e nell'occupazione occorre eliminare paura, ignoranza e ostilità; occorre, in sostanza, informare. Bisogna dire alle persone che non conoscono il problema come stanno veramente le cose. L'informazione diventa conoscenza, non dimentichiamolo, per le persone impegnate nel momento della trasformazione.

Gli Stati Uniti e il Giappone hanno «inventato» milioni di nuovi posti di lavoro negli ultimi dieci anni trasformando, con l'innovazione tecnologica, l'intero loro sistema economico che, da questo mutamento, ha tratto grandi benefici, mentre l'Europa ha perso un milione e mezzo di posti di lavoro. Tutto ciò è avvenuto nonostante l'Europa impieghi, nella ricerca tecnologica, il 20% di tutta la spesa mondiale contro il 27% degli Stati Uniti e il 17% del Giappone. Le percentuali di questi paesi tecnologicamente molto avanzati non sono molto lontane da quella delle risorse impiegate dall'Europa in questo particolare settore, ma gli europei, purtroppo, badano ognuno al proprio orto e molto danaro va sprecato in ricerche sovrapposte, in studi duplicati, senza che l'uno sappia dell'altro. Una più attenta attività di formazione e di informazione, quindi, diventa fondamentale.

Una ricerca recente, dell'Istituto Superiore di Sociologia della Facoltà di Scienze Politiche dell'Università di Milano, ha messo in evidenza, tra l'altro, che solo il 54% dei lavoratori interpellati (406 tra operai e impiegati di un gruppo di aziende lombarde) si sente direttamente e personalmente coinvolto nell'innovazione tecnologica aziendale. L'inchiesta ha rilevato anche lo scarso interesse degli interpellati per interventi sindacali tendenti ad ottenere aumenti salariali e, al contrario, una domanda insistente di informazione e di formazione professionale all'interno dell'azienda e un migliore livello di preparazione scolastica per migliorare l'impatto con l'innovazione tecnologica, ma soprattutto per attenuare gli effetti negativi.

Luigi Berlinguer



Gentile Redazione,
invio i miei ringraziamenti per
aver risposto con sollecitudine
alla lettera inviata nella quale
facevo presente di avere problemi
con il listato «IL TREDICI MINUTO
PER MINUTO» pubblicato sul n.
3/85.

Permettetemi di elogiarvi per aver
risolto il mio problema
chiamandomi addirittura
telefonicamente, fatto davvero
insolito che mi ha lasciato per un

attimo incredulo in quanto mai
avrei pensato che la Redazione di
una rivista si mettesse in contatto
diretto con i lettori...

I suggerimenti datimi sono
risultati di validissimo aiuto e
confermo che il programma gira
ora perfettamente, come asserito.

PAOLO BRUNELLI
Falconara M.ma

Ringraziamo a ns. volta il Lettore per
la cortese lettera inviata e
confermiamo con l'occasione di
essere a disposizione di quanti
desiderino chiarimenti in merito ai
programmi e/o articoli pubblicati.

Sono il possessore di un SEGA
SC 3000H ed ho trascritto alcuni
listati pubblicati sulla rivista.
Pianoforte e Tiro a volo mi hanno
creato qualche problema perché,
nonostante abbia ricontrollato più
volte il listato, la sottolineatura
prevista con il modo grafico
inserito non viene accettata dal
P.C. La risposta è sempre la
stessa: SYNTAX ERROR.

Il listato funziona se non vi sono
sottolineature. Per quanto
riguarda TIRO A VOLO,
nonostante abbia eliminato la
sottolineatura, il programma si
blocca alla linea 470. Mi sapete
spiegare il perché?...

...nonostante abbia tentato più
volte di aprire un FILE e poi
rispettivamente di chiuderlo
(OPEN-CLOSE), non sono venuto
a capo di nulla. Sono io che non
sono in grado di programmare
oppure è il computer che nel suo
linguaggio non prevede le citate
istruzioni?...

Nonostante la lunghezza di questa
lettera chiedo ancora: cosa mi
sapete dire della ventilata messa
in commercio per il 1986 del
nuovo Commodore 128
compatibile con il Commodore
64?...

MARIO MAURI
Mattarello (TN)

Finalmente uno dei pochi possessori
del Sega SC 3000 nella versione
con tastiera rigida QWERTY.

Riguardo al primo problema ribadiamo quanto già scritto a pag. 70 ed 89 della rivista in questione. Le lettere sottolineate vanno digitate con il cursore grafico inserito. Ciò significa che non bisogna sottolineare le lettere con alcun artificio, ma si devono digitare le lettere contenute nella stringa usando il cursore grafico (per intenderci quello a forma di asterisco). Detto ciò invitiamo il lettore a riprovare, tenendo conto anche di un errore in cui molti possessori del Sega ricadono: confondere lo ZERO con la O maiuscola.

Non risulta che istruzioni come OPEN, CLOSE siano contemplate nel BASIC SEGA. Da ciò si deduce che non è prevista la possibilità ahimé, di creare FILES in scrittura. Si è provato ripetutamente a sfruttare le istruzioni INP e OUT per supplire a questa manchevolezza, ma purtroppo i risultati ottenuti, al momento, non sono degni di nota. Per la ventilata introduzione del C 128 Commodore sul mercato entro il 1986, non è possibile attualmente dichiarare nulla di certo. Invitiamo quanti interessati a seguire la nostra rubrica di Attualità che proprio in questo numero parla delle caratteristiche tecniche del C 128.

In merito ad un vostro programma, «BIORITMI/STRIKE», per il VIC 20, pubblicato nel numero di gennaio 1985, vorrei che pubblicaste il relativo programma adatto al SEGA SC-3000.

PATRIZIO FERLITO
Como

Caro Patrizio, un programma analogo in versione SEGA SC-3000 è già stato pubblicato sul n. 4 Luglio/Agosto 1984. Più precisamente a pag. 34 e seguente. La versione per il tuo computer si discosta da quella per il VIC 20 riguardo alla visualizzazione delle tre curve corrispondenti ai vari cicli.

Possiedo da pochissimo tempo un computer SHARP MZ-700 e non conosco bene tutte le sue caratteristiche. Infatti nel digitare

**il programma pubblicato a pag. 84 della rivista di Marzo «Il castello di Wiz», ho trovato difficoltà con le istruzioni in cui comparivano dei codici sottolineati, come ad esempio:
C8C8C8...D8D8.**

Vorrei sapere come fare per inserire queste istruzioni nel mio computer...

CALOGERO MAGRO
Campobello di Licata (AG)

Le difficoltà incontrate dal lettore nel digitare il programma indicato fanno capo a una delle caratteristiche dello SHARP.

Le serie di numeri e/o lettere, che nei programmi si trovano dopo un'istruzione PRINT, corrispondono ai caratteri grafici propri del computer.

I listati pubblicati sono richiamati direttamente, ma poiché lo SHARP adotta una stampante plotter, cioè a pennini, non è possibile ottenere una stampa del set grafico e quindi il computer stesso tramuta i caratteri in codice ASCII: ad ogni coppia di numeri e/o lettere corrisponde un carattere.

In fase di ribattitura si dovrà dunque ricavare mediante la «tabella» indicata a pag. 156 del manuale fornito con lo SHARP, il carattere corrispondente ad ogni singola coppia e digitarlo dopo essere naturalmente entrati nel modo grafico.

Possiedo un VIC 20 e vorrei sapere, possibilmente al più presto, che istruzione usare per sapere quanta memoria ho ancora a mia disposizione dopo aver inserito un programma nel computer. Grazie.

SALVATORE TRIOLO
Partanna (TP)

Da questa, come da altre lettere, abbiamo constatato come molti acquirenti di home-computers non leggano con la dovuta attenzione il manuale di istruzioni allegato, in ogni confezione, al computer. Un vero peccato!!! In ogni caso rammentiamo al sig. Triolo che potrà trovare la risposta al suo problema, e forse a molti altri, consultando il manuale del VIC alla APPENDICE C: Il basic del

VIC. Più precisamente il sottoparagrafo C del capitolo FUNZIONI. Più specificamente l'istruzione FRE(X) a pag. 131 nella forma PRINT FRE(X).

Sono un quattordicenne assiduo lettore e ho deciso di venire in aiuto a tutti quei possessori dello SHARP MZ-700 che, come Dario Letizia, sono un tantino dubbiosi riguardo ai colori effettivi della macchina. Secondo il mio modesto parere sono più di 60.

Con il piccolo programmino allegato, uso la tecnica delle tempere: in effetti tutti i colori sono ottenuti usando il carattere a griglia; cioè:

rosso+giallo=arancio. Per trovare nuovi effetti consiglio di provare a cambiare il valore del codice inserito nella linea 30, rammentando che i codici di questi caratteri sono a pag. 157 del manuale. L'area che occupano i codici colore (cod.col.=16 x col. carattere+colore di sfondo) è compresa tra gli indirizzi \$ D800 e \$ DFFF; invece quella occupata dai caratteri è tra \$ D000 e \$ D7FF.

```
10 LET I = $ D000
20 LET K = 0
30 POKE I, $ EF
40 I = I + 1 : K = K + 1 : IF K =
  255 THEN 60
50 GO TO 30
60 LET I = $ D800:LET K = 0
70 POKE I,K:I=I+1:K=K+1:IF K =
  255 THEN K = 0
80 GO TO 70
```

Per chi trovasse difficoltà ad eseguire la routine che comincia all'indirizzo 68 (in dec.), provi a eseguire questo programmino anche cambiando i valori:

```
10 FOR I = 1 TO 100:FOR R = 1
  TO 10:POKE
  2618,R:USR(68):NEXT:USR(69)
20 X=X+1:IF X = 40 THEN 40
30 NEXT I
40 X = 0:FOR H = 1 TO
  500:NEXTH:NEXT I
```

MARCO MELGAZZI
Novara

Si ringrazia innanzitutto il lettore per i complimenti espressi nella lettera pubblicata in forma ridotta per esigenze di spazio. Digitando le due routines proposte si



incontrano alcune difficoltà e per la precisione un ILLEGAL DATA alla linea 60 della prima routine ed un errore di NEXT alla linea 40 della seconda routine.

Nel primo caso, assegnando le variabili così come presentate dal programma, i valori sono negativi e gli indirizzi errati.

Nel secondo caso il NEXT I presente nell'ultima linea di programma, blocca completamente il sistema.

L'idea non è cattiva ma resta il problema che l'effetto è valido solo con il carattere griglia, mentre, per altri caratteri, non è possibile ricorrere alla soluzione proposta.

A tutti i possessori dello SHARP MZ 700 desiderosi di provare le routine, proponiamo le nostre due corrispondenti, elaborate sulle base di ciò che il lettore ha proposto.

```
10 PRINT"@"
20 FORK=0T0127:FORJ=0T0119:POKE#0000+J, #
EF:POKE#0000+J,K:NEXTJ:CURSOR0,6:PRINTK
30 GETZ#:IFZ#=""THEN30
40 NEXTK
```

```
10 FORK=1T0255:POKE261B,K:USR(68):NEXT:J
SR(69):GOSUB100
20 FORK=1T0255:POKE261B,K/2:USR(68):NEXT
:USR(69):GOSUB100
30 FORK=1T0255:POKE261B,K/5:USR(68):NEXT
:USR(69):GOSUB100
40 FORK=1T050:POKE261B,K/5:USR(68):FORS=
1T030:NEXTS,K:USR(69):GOSUB100
50 FORJ=1T020:FORK=0T030:POKE261B,K:USR(
68):NEXTK,J:USR(69)
60 END
100 FORP=1T01000:NEXTP:RETURN
```

Possiedo un VIC 20 espanso a 3K, 8K, 16K, ed ho cominciato a crearmi dei giochi.

Mi rivolgo a List, per chiedere:

- 1) Come si possono fare codici segreti per passare a schermi successivi?
- 2) Come si può scrivere un programma per usare il joystick?

PAOLO ZANELATO
Cameri (NO)

Non è difficile creare delle chiavi che permettano il passaggio a schermi successivi sfruttando, ad esempio, l'istruzione WAIT da pochi presa in considerazione.

Sul manuale del computer, la sua descrizione è la seguente:

...Viene usata per interrompere l'esecuzione del programma fino a che i contenuti di una locazione di memoria non cambino in un modo specifico.

Si può provare ad esempio con:
10 WAIT 197,63,64

Il programma si fermerà fino a quando non sarà premuto il tasto funzione sulla destra della tastiera. Questo è uno dei modi, ma combinando più istruzioni di questo tipo si può usare come chiave una parola intera invece che una sola lettera. I parametri presenti nell'esempio riguardano nell'ordine: 197 locazione di memoria che legge la tastiera.

63 contenuto della locazione premendo il tasto specificato.

64 contenuto della locazione non premendo alcun tasto.

Per conoscere i codici corrispondenti ai tasti si può fare uso della routine:

1 PRINT PEEK (197)

2 GO TO 1

Dando il RUN si visualizzerà una sfilza di numeri che cambieranno ogni volta che un tasto sarà premuto. Basterà ricordare il codice e sostituirlo a 63 nell'istruzione WAIT.

Per ciò che riguarda l'uso del joystick, si dovrà leggere il contenuto della locazione 37137 con lo stesso metodo presentato per la locazione 197.

Il suo contenuto cambia in funzione della posizione della leva del joystick. Con una opportuna istruzione IF ... THEN ..., associata al contenuto rilevato nella locazione di memoria, non sarà difficile modificare il valore dei parametri riguardanti il movimento del carattere controllato dal joystick.

Dal N. 3 (Marzo 1985) ho digitato FUNZIONI TRIGONOMETRICHE per ORIC 1 che mi ha interessato particolarmente.

Solo alla fine del programma, dopo la frase «Per finire andiamo a vedere i grafici», ho riscontrato qualche problema. Allego il seguente schema uguale per tutte le funzioni, il che ... non ha alcun significato.

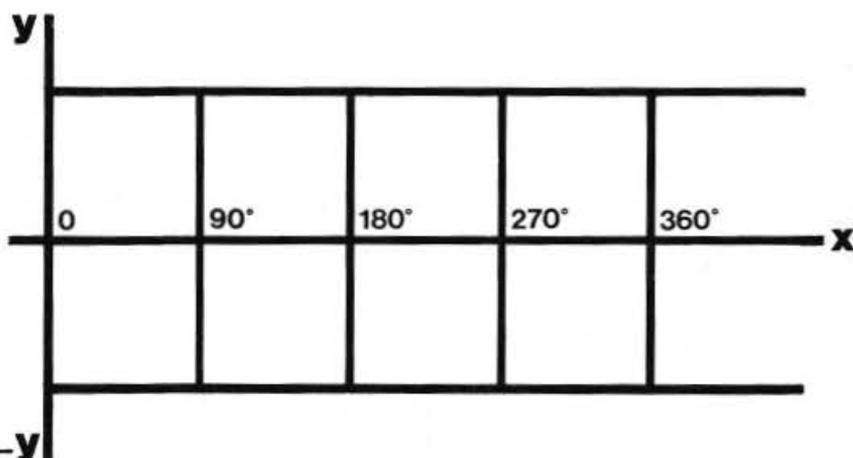
**Ciò è dovuto ad un mio errore o manca qualcosa?
I miei saluti**

AUGUSTO MOGGIO
Verona

Sembra di capire che in realtà al

lettore non viene visualizzata la curva corrispondente alla funzione calcolata.

Dopo il salto alla routine che inizia alla linea 500 (realizzato per mezzo della linea 45), il programma dovrebbe tornare, grazie ad un RETURN, alla linea 48 da cui iniziano una serie di subroutine che definiscono le funzioni, disegnano gli assi cartesiani (lo schema inviatoci) ed infine i grafici delle funzioni. Non rimane che controllare attentamente il listato già trascritto sul computer facendo attenzione a controllare se manca la linea 43, che contiene un RETURN, oppure la routine di disegno delle curve, ottenuta dalle linee 300-350.



Pallottoliere contro computer

Esiste, in Cina, una Associazione del pallottoliere che unisce ancora, nella civiltà dell'elettronica, moltissimi appassionati. Il pallottoliere ha circa mille anni, ma nonostante la sua veneranda età, questo antichissimo strumento per calcolare tiene validamente testa alle macchine. Li Xin, vice presidente dell'Associazione, ha dichiarato di recente che questo elementare strumento è più efficace del computer, costa meno, fa meglio certe operazioni e, soprattutto, non si guasta mai.

Rifiuti spaziali

Un vero e proprio anello di rifiuti tecnologici orbita ormai da tempo intorno alla Terra. Satelliti fuori suo, rottami di satelliti e di altri oggetti spaziali, frammenti metallici e di altri materiali prodotti dalle esplosioni dei tests militari nello spazio inquinano ormai l'ambiente un tempo incontaminato, che sta sopra alle nostre teste. Se i paesi che detengono le tecnologie spaziali non troveranno presto un accordo i futuri vettori spaziali, civili e militari, vedranno aumentare di molto le possibilità di collisione nello spazio. Il «Jane's Spaceflight Directory», una pubblicazione specializzata che si stampa a Londra ha rivelato che già due navicelle spaziali, una sovietica della serie Salyut, e la statunitense Challenger, hanno riportato danni durante le loro missioni per la collisione con questi detriti vaganti nello spazio.

La Sony nello spazio

La grande compagnia giapponese Sony, come molti non sanno, non solo produce elettrodomestici piccoli e grandi ma è anche una venditrice, sul mercato nipponico, dei beni più disparati. La Sony, infatti, importa e vende mobili, aeroplani jet-executive, e altre cose di produzione europea e

americana. Adesso la Sony ha deciso di importare per il mercato nipponico anche satelliti prodotti negli Stati Uniti dalla RCA. Alcuni osservatori specializzati nel campo delle comunicazioni non escludono che le due compagnie, in un futuro prossimo, decidano di mettere in orbita e rendere operativo un loro satellite.

Personal in salita

La domanda di personal computers in Italia, stabilizzata due anni fa intorno al 45%, ha acquistato molta velocità lo scorso anno, il 138% in più. È un fenomeno sbalorditivo (che si riferisce alle unità vendute) e che nessuno degli specialisti del campo aveva previsto. Questo aumento inaspettato ha trovato diverse spiegazioni; prima fra tutte la ormai superata diffidenza da parte delle piccole imprese e degli utenti, un tempo restii all'acquisto di computers. Il fenomeno dovrebbe continuare anche quest'anno perché le piccole imprese devono passare alla contabilità ordinaria, come prevede la legge Visentini e, quindi, dovranno informatizzarsi.

La letteratura grigia

Il primo programma europeo per la documentazione elettronica sulla «letteratura grigia» è entrato in funzione. Si chiama «SIGLE», System for information on grey literature produced in Europe, ed è stato realizzato con la collaborazione della Comunità Economica Europea. Della «grey literature» fanno parte tutti quei documenti non ufficiali, relazioni scientifiche non pubblicate, note tecniche e documenti tecnici, comunicazioni private che non hanno mai avuto l'onore della pubblicazione e che possono però avere grande interesse per certi settori di specializzazione.

Il «Sigle» conta su 20 mila documenti ed è accessibile, con terminale, attraverso la rete Euronet-Diane. Al programma partecipano cinque Paesi: Francia, Irlanda, Gran Bretagna, Germania

Federale e Belgio. Anche l'Italia probabilmente aderirà al programma con un centro che potrebbe essere installato al Consiglio Nazionale delle Ricerche.

L'uomo e l'automazione

«L'automazione sull'attività dell'uomo». Questo il tema di un interessante Convegno di carattere internazionale che si terrà in ottobre a Milano. Il Convegno è organizzato dalla Società Italiana di Ergonomia (SIE). Il Convegno affronterà gli effetti dell'automazione in quattro settori: modalità e strumenti di progettazione, sistemi di produzione e controllo, elaborazione e trasmissione delle informazioni, caratteristiche dei nuovi prodotti. L'ergonomia è la disciplina che studia gli effetti delle interazioni umane con l'ambiente e gli oggetti che circondano l'uomo nel corso delle sue attività. Alla luce di queste nuove conoscenze, il convegno discuterà anche la revisione dei sistemi di lavoro ormai pesantemente condizionati dalle nuove tecnologie. Già da adesso possiamo telefonare alla cometa di Halley, o meglio possiamo, componendo alcuni numeri del sistema telefonico britannico, sapere tutto sul cammino della cometa e sul viaggio delle sonde che tra la fine di quest'anno e la primavera del prossimo le andranno incontro. Il servizio è in funzione 24 ore su 24 e finirà nell'estate del 1986. Le informazioni vengono aggiornate ogni settimana. Il servizio è stato istituito dalla «British Aerospace» e dalla «Società per la Cometa di Halley» ed è in grado di dare tutte le informazioni riguardanti il moto della cometa, già entrata nel nostro sistema solare, e sulle missioni delle sonde automatiche, due sovietiche, una giapponese e l'europea «Giotto» che studieranno da vicino la cometa che ritorna verso di noi ogni 74 anni. I numeri telefonici da chiamare per avere queste notizie sono: Londra 01-790-3400; Birmingham 021-355-6144; Glasgow 041-552-6300; Liverpool 051-236-8474; Bristol 0272-279494; Cardiff 0222-399855; Manchester 061-246-8061; Leeds 0532-8013. A



questi numeri occorre far precedere il prefisso della Gran Bretagna che è lo 0044.

Dinosauri e computers

Tre scienziati americani delle Università del Rhode Island, del Michigan e della Columbia, sono riusciti a trovare il modo di ricostruire, con il computer, il tempo meteorologico ... del passato. I tre, in sostanza, affermano di avere fatto ciò che fa Bernacca alla televisione: «prevedono» le condizioni meteorologiche di epoche fino a 100 milioni di anni fa. I tre ricercatori americani analizzano con il computer la polvere racchiusa nelle profondità

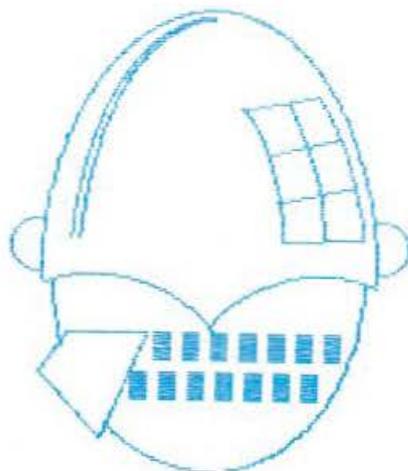
dei fondali oceanici in modo da studiare l'evoluzione climatica dalla preistoria ai nostri giorni. Gli studi dei tre docenti hanno suscitato molto interesse anche perché si spera di ottenere nuove informazioni, e forse la prova, di come si estinsero i dinosauri. La teoria oggi più accreditata, infatti, afferma che i dinosauri scomparvero per un grande sconvolgimento climatico che avvenne 65 milioni di anni fa.

Jazz: il nuovo programma interattivo

Non si tratta di un programma musicale come il titolo potrebbe far pensare, ma bensì di un

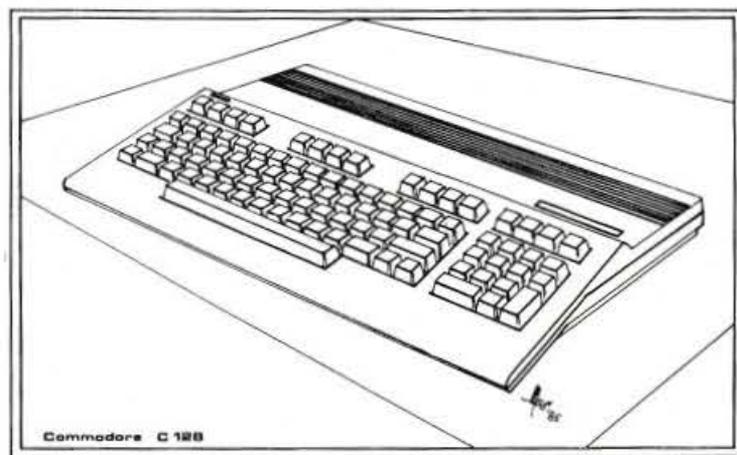
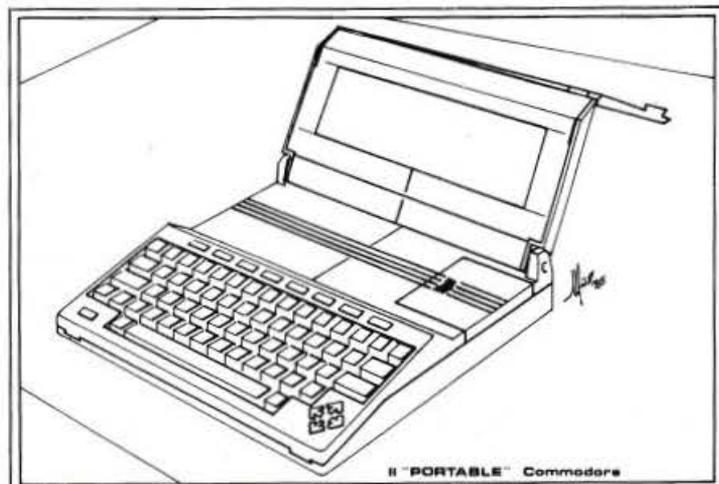
pacchetto software per il Macintosh della Lotus che spera eguagli il successo di vendite del famoso «1-2-3» dell'IBM PC.

Jazz consta di cinque programmi — Numbers, Graphics, Database, Text, Communications — integrati fra loro. Il pacchetto risulterà senz'altro utile agli utenti Macintosh che fino ad oggi non hanno avuto la possibilità di passare, ad esempio, senza difficoltà dalla stesura di una lettera alla sua stampa, alla sua archiviazione, direttamente e soprattutto velocemente come il Jazz dà la possibilità di fare. A conferma della validità del programma, la notizia che la Apple ha dichiarato di avere intenzione di introdurre il Jazz nella rosa dei programmi dedicati ai propri computers.



LE NOVITÀ

Vi presentiamo le ultime novità in casa COMMODORE, presenti al CES (Consumer Electronic Show) di Las Vegas.



In America, si sa, ogni manifestazione che si rispetti deve avere quel tocco di spettacolarità che la contraddistingua. A tale «regola» non si sottraggono, naturalmente, nemmeno mostre ed esposizioni. Cercate dunque di immaginare — se ci riuscite! — «cosa» possa essere una mostra dedicata ai computers o Computer Show come da definizione propria. Essere presente con i propri prodotti ad uno



«show» di risonanza internazionale rappresenta per le Case produttrici una grande opportunità. Uno fra i più prestigiosi appuntamenti annuali è proprio quello di Las Vegas sul cui «terreno» si danno puntualmente «battaglia» i colossi informatici americani, contendendosi, a suon di novità, i favori dei visitatori che per tutta la durata dell'esposizione fanno ressa attorno agli Stands e combattono a spada tratta (leggi «gomitate» n.d.r.) per vedere e toccare le nuove «creature». Al CES 1985 uno degli Stands che più ha offerto è stato senz'altro quello della COMMODORE, fra le cui novità spiccavano un computer portatile con incorporato display a cristalli liquidi ed alimentazione a batteria, il già annunciato C-128 ed un professionale compatibile IBM.

— IL PORTABLE COMPUTER. Le prime cose che colpiscono sono il perfetto ed accattivante design ed il bianco assoluto.

Il Portable si presenta lineare e compatto — $5,3 \times 26,6 \times 29,7$ cm —, con il display inserito internamente al coperchio.

Il processore principale è il nuovo 65C102: si tratta di una versione migliorata del vecchio 6502, molto simile al 65C02 dell'APPLE IIc. Nella versione base il Portable viene fornito con 32K RAM e 96K ROM (nella ROM sono contenuti i programmi applicativi). Vi è inoltre uno SLOT nel quale inserire dell'ulteriore software su cartridge. Le quattro batterie AA di cui il computer è fornito consentono una «autonomia» di circa 15 ore. Il Portable è collegabile al Modem o mediante il normale sistema di OUTPUT (tramite cioè spinotto) o tramite le stesse capsule acustiche. Per quel che riguarda il collegamento con periferiche, il computer è provvisto di porta seriale Commodore che permette l'interconnessione con tutte quelle disponibili per il CBM 64, di una porta RS 232 e di una interfaccia per stampante parallela Centronics. Se si considera che i computers Commodore sono «tristemente» famosi per il loro interfacciamento non standard, quanto detto è estremamente positivo. Il display, come già detto, è a cristalli liquidi (LCD), di produzione Commodore, e consta di 80 colonne \times 16 linee. Uno dei problemi comuni ai displays LCE di grandi dimensioni presenti sul mercato è la cattiva risoluzione: sotto questo aspetto invece il display del Portable è eccellente. I programmi applicativi forniti sono: Word Processor, File Manager, Spreadsheet, Address Book, Scheduler, Calculator, Memo Pad, Terminal Emulator, Basic, Machine Code Monitor. Tutti sfruttano ampiamente gli otto tasti funzione programmabili di cui il Portable è provvisto. Il prezzo di questo nuovo «nato», presentato come un piccolo gioiello, non è stato reso noto.

— IL C 128. Molto si era vociferato ancor prima della sua presentazione ufficiale.

Come il Portable, il C 128 presenta un design decisamente curato, ma la sua caratteristica principale è data dai tre diversi processori di cui è fornito: il 6510, l'8502 e lo Z-80.

Questi permettono al computer di operare in tre differenti modi: CBM 64 Mode, C 128 Mode, CP/M Mode.

Usato in CBM 64 Mode, il C 128 sfrutta il 6510 come CPU, il Sound Chip 6581 e dispone di 64K RAM e 16K

ROM. Proprio come il 64 ha il display formato da 320×200 pixels, 16 colori, possibilità di definire otto sprites. Grazie all'apposito connettore può usufruire del software su cartridge dedicato al 64 che, a detta della Commodore, è compatibile al 100%.

In C 128 Mode il computer sfrutta come processore l'8502, il Sound Chip 6581 e dispone di 128K RAM e 64K ROM; tramite disco la RAM può essere portata fino a 512K.

Il display può essere selezionato in:

- alta risoluzione pari a 640×200 pixels
- media risoluzione pari a 320×200 pixels

Da tenere presente che il Basic con il quale il C 128 è corredato (Basic Version 7.0) è decisamente più accessibile rispetto al Basic finora adottato per gli altri computers Commodore.

Come 128 vero e proprio il computer può inoltre sfruttare al massimo l'alta velocità di trasferimento dati del nuovo Drive — il 1571 (SFD 1001).

Per operare sotto CP/M il C 128 sfrutta il processore Z-80: questa scelta sta ad indicare la nuova tendenza Commodore ad indirizzare la propria politica verso sistemi più vicini allo Standard.

Sotto CP/M il C 128 si trasforma in un normale 8-bit; il sistema utilizzato è, per la precisione, il CP/M PLUS che permette di sfruttare pienamente i 128K RAM a disposizione. Lo schermo è selezionabile — 40 o 60 colonne — e si possono visualizzare fino a 16 colori.

A detta della Commodore la maggior parte dei programmi sotto CP/M-80 è compatibile.

Per quel che riguarda i dispositivi I/O, il C 128 si rifà allo standard Commodore: porta connessione registratore, porta seriale, porta cartridge e, novità, porta RGB e TV.

La tastiera del 128 è stata decisamente ben progettata: si tratta infatti di una tastiera completa di tipo standard — professionale, con tasti numerici separati, tasti funzione programmabili, due sets di tasti cursore — uno in alto ed uno in basso — in modo da rendere il C 128 completamente compatibile per quel che riguarda il software con il 64.

— IL PC COMPATIBILE IBM. Fino a qualche tempo fa nessuno avrebbe mai creduto che la Commodore — orgogliosa del proprio «marchio», della propria tecnologia e fermamente decisa e convinta di poter «sopravvivere» con i propri mezzi — avrebbe portato sul mercato un PC compatibile. Il calo nelle vendite dei propri sistemi, ha forse provocato questo «ripensamento»? Probabilmente sì. Ad ogni modo nonostante questa «sorpresa» la Commodore non sembra aver perso tutte le speranze: il nuovo PC, infatti, sarà disponibile solo per il mercato europeo e non per quello americano.

— LE PERIFERICHE. Oltre alle nuove macchine, lo Stand Commodore ha presentato anche una gamma di nuove e «rinnovate» periferiche. Naturalmente le maggiori attenzioni sono state prestate al nuovo Drive. In definitiva questo può essere utilizzato nei tre Modi previsti dal C 128: CBM 64, C 128, CP/M. Utilizzato a supporto del 128/64, il trasferimento dati avviene alla velocità permessa dal 64, pari a 300 cps; collegato al 128 vero e proprio la velocità sale a 1500 cps fino ad arrivare a ben 3500 cps sotto CP/M.



Atene senza schiavi l'Italia e i robots

di Giuseppe Breveglieri

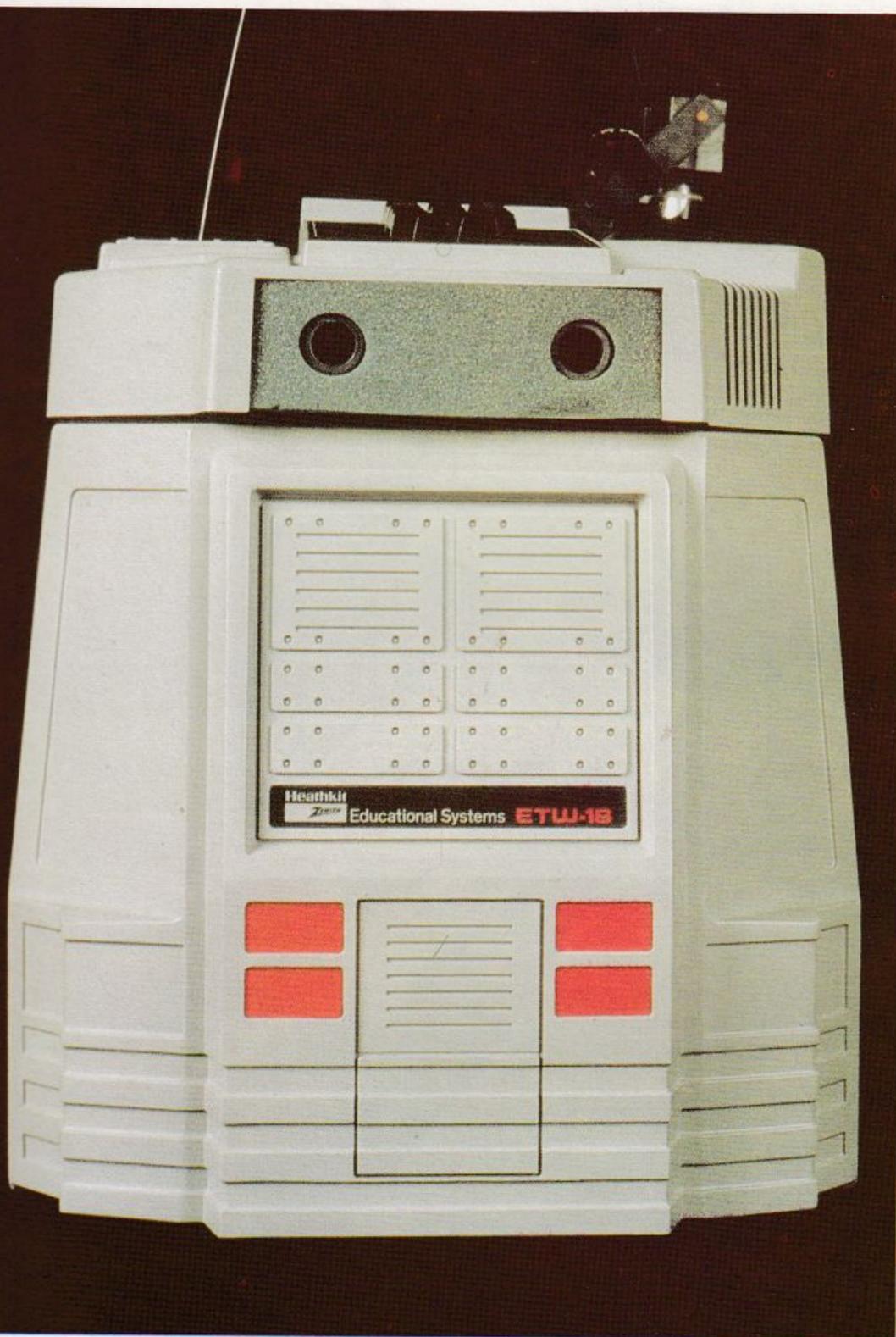
Dèi e dèmoni robotizzati popolano ormai da anni la letteratura fantascientifica. Di queste creature artificiali l'uomo ha fatto il suo riflesso, il suo doppio. Un grande filosofo, l'illuminista Voltaire, affermava che l'uomo, così facendo, aveva reso la pariglia a Dio che ci ha fatti a sua immagine e somiglianza. Pensava Voltaire, alle meschinità, alle piccinerie che gli uomini hanno rifilato ai loro dèi nell'antichità. Ma il concetto può essere adattato anche agli scrittori dei tempi nostri che attribuiscono ai loro automi fantascientifici poteri. A scrivere di robots ha cominciato, nel 1938, Eando Binder con il racconto «lo robot». Isaac Asimov (vedi List n. 4) che è da tutti considerato il papà del robot nella fantascienza (ma a quanto pare Binder lo ha preceduto) pubblicò un paio di anni dopo una novella dal titolo «Robbie» cominciando così una sorta di saga del robot positivo che si è protratta a lungo e che ancora non è finita. Asimov è anche l'estensore delle tre leggi della robotica; i tre comandamenti ai quali gli automi devono obbedire nelle sue storie.

Le tre leggi dicono: a) Un robot non può recare danno ad un essere umano né può permettere che per sua incuria un essere umano riceva

danno. b) Un robot deve obbedire agli ordini ricevuti dagli esseri umani purché tali ordini non contrastino con la prima legge. c) Un robot deve proteggere la propria esistenza fino a quando questa autodifesa non contrasti con la prima e la seconda legge.

Da allora tutti gli autori di fantascienza si sono tacitamente attenuti a questo codice inventato da Asimov, ma lo stesso Asimov, alcuni anni dopo, ha fatto marcia indietro perché il concetto di «essere

umano», ha ammesso, è decisamente ostico per un robot. Robot deriva dal cecoslovacco «robota» che significa lavoro obbligato e servile. La parola fu coniata dallo scrittore boemo Karel Capek per il suo melodramma, RUR, che ebbe grande successo prima a Praga, nel 1921, poi nei teatri di Londra e New York, con il sottotitolo «Rossum's Universal Robots». Quando, sul palcoscenico di un teatro di Praga, il viso stupido di una macchina vestita da uomo pronunciò



la parola robot, gli spettatori capirono subito; «robota» nella loro lingua significa anche lavorare duro. Il robot, dunque, era l'operaio artificiale, l'uomo sintetico, l'automa. RUR parodiava la rivoluzione dei Soviet. Capek, infatti, aveva paragonato gli ottusi congegni che aveva messo in scena allo sciocco uomo-massa.

Automa, come sappiamo, è una parola ambigua. Se l'appiccichiamo ad un uomo, intendiamo dire che

quest'uomo si muove senza alcuna volontà, inconsciamente, spinto da una forza che non gli appartiene. Ma automa viene dall'antico greco «autòmaton», che vuol dire muoversi spontaneamente, di proprio impulso. L'automa a similitudine umana, l'androide, il robot, è un simbolo affascinante e terrificante insieme. Affascina perché ci dice che l'uomo è potente; spaventa perché in fondo ci dice che anche l'uomo può essere macchina.

Difficile dire quale fu il primo automa

o quale fu la più vecchia rappresentazione meccanica dell'uomo o dell'animale. Dobbiamo, forse, risalire alla preistoria e supporre che l'immobile immagine in legno dell'uomo primitivo sia il primo stadio dell'automa. A questa figura senza mobilità l'uomo ha sempre tentato di dare vita e movimento. Il primo Adamo da laboratorio pare sia cinese; un uomo di legno capace di correre, sedere, alzarsi, ballare, ma a quanto pare piuttosto imprudente.

Mostrato alla Corte di un Paese Lontano, fissò con i suoi occhi senz'anima lo splendido viso della sovrana del posto. Lo fece con molta insistenza — una insistenza quasi umana — al punto che il sovrano, incollerito, lo ruppe in centinaia di pezzi con un randello. Anche in Grecia, nel Terzo Secolo a.C., fiorirono i costruttori di macchine automatiche. Filone di Bisanzio ed Erone furono famosi progettisti di automi prima dell'era volgare. Erone per far marciare le sue macchine usava pesi, aria compressa e vapore. Gli arabi Avicenna e Al-Djazari non furono da meno, poi, dopo un medioevo davvero oscuro per i costruttori di androidi, vennero gli studi e le realizzazioni di Leonardo.

Ma la grande stagione degli androidi è il secolo dei lumi, il 700, con i grandi costruttori come Jaques Vaucanson, paragonato da Voltaire a Prometeo.

La costruzione degli automi, nel 700, anticipa la Rivoluzione industriale che adopererà molte delle soluzioni meccaniche escogitate per muovere queste copie, allucinanti nella loro perfezione, degli esseri umani. Costruire automi, infatti, nel 700 e anche prima di allora, non è solo un gioco; è soprattutto una ossessione, un sogno, una sfida filosofica dell'equiparazione dell'uomo alla macchina.

Gli autori di fantascienza insistono quasi sempre su questo punto. Il robot nelle loro storie, è un automa antropomorfo, vale a dire con figura e attributi umani. Ma nella realtà queste macchine sono molto differenti dall'uomo e non ne possiedono le capacità.

I ricercatori più avveduti sanno da tempo che l'affidarsi all'uomo come modello per il robot è una trappola.



Non bisogna presumere, infatti, che un robot debba avere una mano perché l'uomo ha una mano o le braccia perché l'uomo ha le braccia o un occhio perché l'uomo li ha o il cervello perché l'uomo lo ha e così via.

L'uomo, oggi, ha una visione stereo e mani abilissime forse perché molti milioni di anni fa doveva saltare di ramo in ramo, di albero in albero e se non avesse avuto una visione stereoscopica sarebbe caduto a terra con conseguenze facilmente immaginabili. Il robot, invece, vive nel mondo della fabbrica e le fabbriche sono organizzate e strutturate in maniera molto diversa dalle foreste dove l'uomo un tempo viveva. Nelle fabbriche il lavoro è regolare e le sorprese molto limitate. Non dobbiamo dimenticare che il robot, a differenza dell'uomo, è una macchina e che ci sono molte cose che l'uomo fa benissimo e che il robot fa malissimo e viceversa. Prendiamo, ad esempio, un robot che ormai quasi tutti hanno in casa: la lavapiatti. Questo robot, come sappiamo, non lavora come una persona che lava i piatti. La macchina non allunga il braccio per prendere un piatto alla volta, non usa lo straccio per asciugare e non prende il piatto successivo. Il robot lavapiatti si serve di un forte getto d'acqua, molto calda, per fare il suo lavoro, una cosa che una persona non può fare perché sarebbe sommersa dall'acqua che molto probabilmente allagherebbe anche la casa.

La macchina, in sostanza, lava i piatti a modo suo. Questo è un buon esempio per capire i tranelli di cui si è parlato a proposito dell'uomo come modello per costruire un robot. Ma non sono molte le persone che si rendono conto, ancor oggi, di avere già in casa un robot domestico.

I robots, quindi, sono solo delle macchine manipolatrici e da programmare per determinate operazioni. Difficilmente quindi i robots potranno sostituire completamente l'uomo nelle lavorazioni, sempre che non si tratti di operazioni particolari dove è in gioco la salute o l'incolumità degli esseri umani. Per arrivare alla totale intercambiabilità tra l'uomo e il robot sono necessari sforzi economici e tecnici, oggi inimmaginabili, in

almeno tre settori: l'intelligenza artificiale, i sistemi di controllo, le capacità sensoriali. Il calcolatore più progredito, oggi, è una scimmia idiota se lo si confronta con il cervello dell'uomo. Ma il nostro

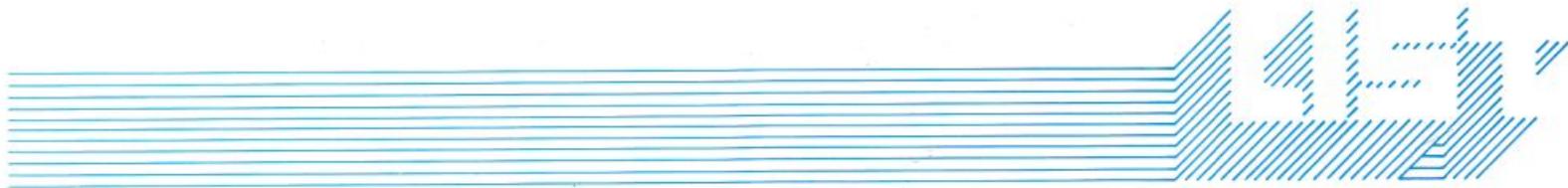
«interpretare» la realtà che ci circonda, la capacità di elaborare risposte agli stimoli esterni in modo rapido e coordinato, non possono ancora essere trasferiti ad una macchina. Probabilmente passerà



cervello ha circa tre miliardi di anni di evoluzione organica, mentre l'elaboratore, di anni, ne ha solo una trentina.

Le possibilità che hanno il polso e la mano dell'uomo e, la sua corteccia cerebrale (attraverso la visione tridimensionale e colorata) di

molto altro tempo prima che ciò avvenga, posto che possa avvenire, perché, dicono gli esperti per fare un esempio, la sola capacità degli attuali sistemi elettronici di visione che un robot possiede dovrebbe essere elevata di un fattore pari a 10^5 per competere con quella



umana.
Esistono già, sulle linee di produzione, robots dotati di occhi (telecamere) e di cervello (computer) in grado di scegliere pezzi per forma e dimensione e di metterli al posto giusto. Quando una macchina compie questi movimenti è già arrivata all'università dei robots. Un robot «laureato» ha cinque gradi di libertà, riesce cioè a muoversi come vuole poiché, di norma, bastano cinque assi per far assumere qualsiasi posizione ad un oggetto nello spazio. Robots di questo tipo danno notevoli vantaggi all'industria.

Nel mondo esistono attualmente circa 38 mila robots industriali. Il nostro paese è in buona posizione nella produzione di queste macchine.

Nella classifica dei produttori siamo quinti dopo Giappone, Stati Uniti, Germania Federale e Svezia. È, questa dei robots, una comunità mondiale ancora ristretta, impiegata soprattutto nei settori dell'automobile, dell'elettronica, degli elettrodomestici, della meccanica e dei mobili.

Un robot costa da 250 milioni ad alcuni miliardi ma non ha tempi morti e aumenta la produttività del 50% rispetto alle macchine tradizionali. I quattrini spesi per un robot si ammortizzano in fretta.

L'avvento dei robots ha creato grosse aspettative: i fisici li vogliono mettere al lavoro nelle centrali nucleari, gli agricoltori pensano di impiegarli nella semina e nella irrigazione. Persino i medici parlano di robots per le operazioni chirurgiche della massima precisione; uno di questi, di fabbricazione tedesca, il «lithotripter» comunemente conosciuto come lo «spaccapietre» polverizza già i calcoli renali con un sistema computerizzato ad ultrasuoni che evita l'intervento del chirurgo. Nella corsa alla robotizzazione, come abbiamo visto, i giapponesi sono in testa a tutti; amministrano bene il vantaggio che hanno accumulato nella prima fase dell'automazione.

I robots in Giappone sono dappertutto. I primi automi con il colletto bianco stanno arrivando anche negli uffici. I loro colleghi in tuta, invece, saldano, verniciano, montano motori nelle grandi

fabbriche. Ma il Giappone ha un grosso rivale negli Stati Uniti. I colossi statunitensi dell'elettronica stanno recuperando molto terreno e costruiscono già cervelli e sistemi nervosi per robots altamente sofisticati.

L'Europa, come sappiamo, ha una situazione più complessa con tre paesi all'avanguardia, Germania Federale, Svezia e Italia che possiedono l'80% dei robots industriali europei. La domanda di robots industriali in Europa, afferma una ricerca di mercato statunitense, triplicherà ogni cinque anni. Nel 1990 i robots europei saranno più di 20 mila; un mercato di alcune centinaia di miliardi di dollari.

Filoni autonomi di ricerca, capaci di vincere il confronto con i maggiori produttori del mondo, sono stati sviluppati in Italia negli ultimi anni. Era, questo, un fatto impensabile fino a pochissimo tempo fa, ma la crisi energetica, paradossalmente, ha spinto i paesi come l'Italia, che hanno maggiormente sofferto della crisi del petrolio, sulla strada della ricerca avanzata in questo campo. Oggi molte aziende italiane vendono all'estero robots industriali molto perfezionati.

Esistono in Italia tre poli produttivi del settore che, all'estero, riescono a piazzare il 35% dell'intera produzione nazionale. Sono, questi poli, il COMAU del gruppo Fiat, l'OCN-OSAI del gruppo Olivetti, e un terzo polo che comprende una quindicina di imprese private dotate di tecnologia molto sofisticata. Esiste anche un quarto polo pubblico (IRI-STET) che sta accorpando alcune imprese per renderle più competitive sul mercato internazionale.

Ma, incredibilmente, pur essendo tra i primi produttori del mondo, siamo in ritardo nella applicazione della robotica industriale alle nostre aziende. Siamo, in sostanza, dei buoni costruttori ma dei cattivi utilizzatori.

Nonostante la progressione sia stata di un buon 50% annuo negli ultimi cinque anni, siamo indietro di sei o sette anni non solo rispetto al Giappone, USA, Germania Federale, ma anche all'Unione Sovietica e alla Francia.

Le regioni maggiormente robotizzate in Italia sono il Piemonte, la Lombardia, la Liguria, il Veneto e

l'Emilia-Romagna. Il Piemonte è la regione leader, con la Fiat che ha già robotizzato i suoi impianti del 30% e che prevede di arrivare al 70% nei prossimi sette anni. Verso la fine degli anni 90, dunque, il nostro paese dovrà avere già affrontato questo diverso modo di produrre e la conseguente crisi del lavoro. Fin d'ora il sistema sindacale, economico e politico deve cercare una soluzione per affrontare il problema e per trovare una alternativa ai mestieri cancellati dalle macchine. Ma, dicono parecchi esperti, la temuta, gigantesca ondata di disoccupazione potrebbe non esserci.

Questo straordinario zoo fatto di membra meccaniche e di cervelli di silicio porterà ad un movimento di trasformazione molto lento e, quindi, abbastanza controllabile.

L'innovazione tecnologica dunque colpisce la fantasia ma inquieta per le conseguenze che può avere. Gli economisti a volte sono incerti e disarmati quando devono analizzare il fenomeno: esistono tra loro ottimisti e pessimisti. I pessimisti pensano che, nonostante il suo lento divenire, l'automazione lascerà senza lavoro dal 10 al 15% della popolazione attiva dell'Europa di qui al 1990. Gli ottimisti, più realisticamente forse, credono in una flessione momentanea dei posti di lavoro poi, come è avvenuto con le nuove tecnologie, in una successiva espansione. Gli studiosi, in effetti, non hanno ancora strumenti certi per valutare l'entità di questi fenomeni; l'automazione, nell'esperienza degli ultimi decenni, dicono, è un fatto secondario nella determinazione dei posti di lavoro. Pesano di più altri elementi quali l'evoluzione della domanda di beni e di servizi e l'incremento della produttività.

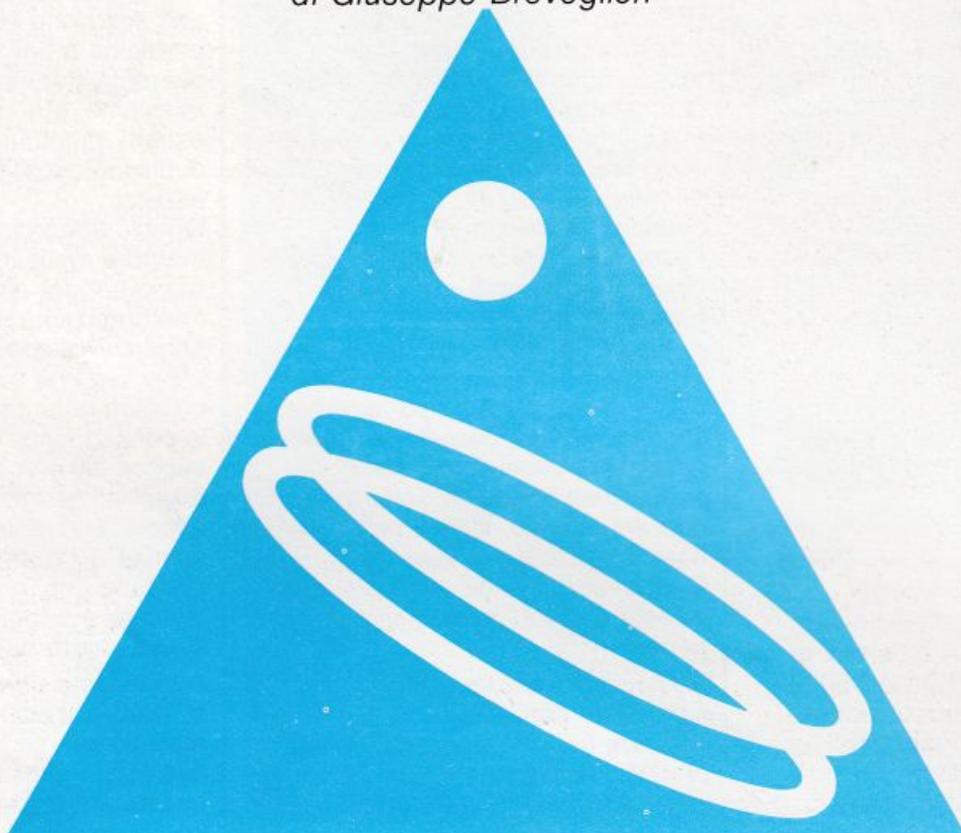
L'uomo, dunque, ha creato uno strumento meccanico che gli è utile ma è spaventato perché teme che il meccanismo diventi potente, pericoloso, capace di innescare processi incontrollabili. L'apprendista stregone dà vita alla sua immobile scopa ma poi non riesce a fermarla. Ma è più probabile (e augurabile) che la nostra civiltà futura sia simile ad una Atene senza schiavi, con l'uomo liberato dalle attività più avvilenti e i robots destinati ai lavori più pesanti e pericolosi.



TSUKUBA

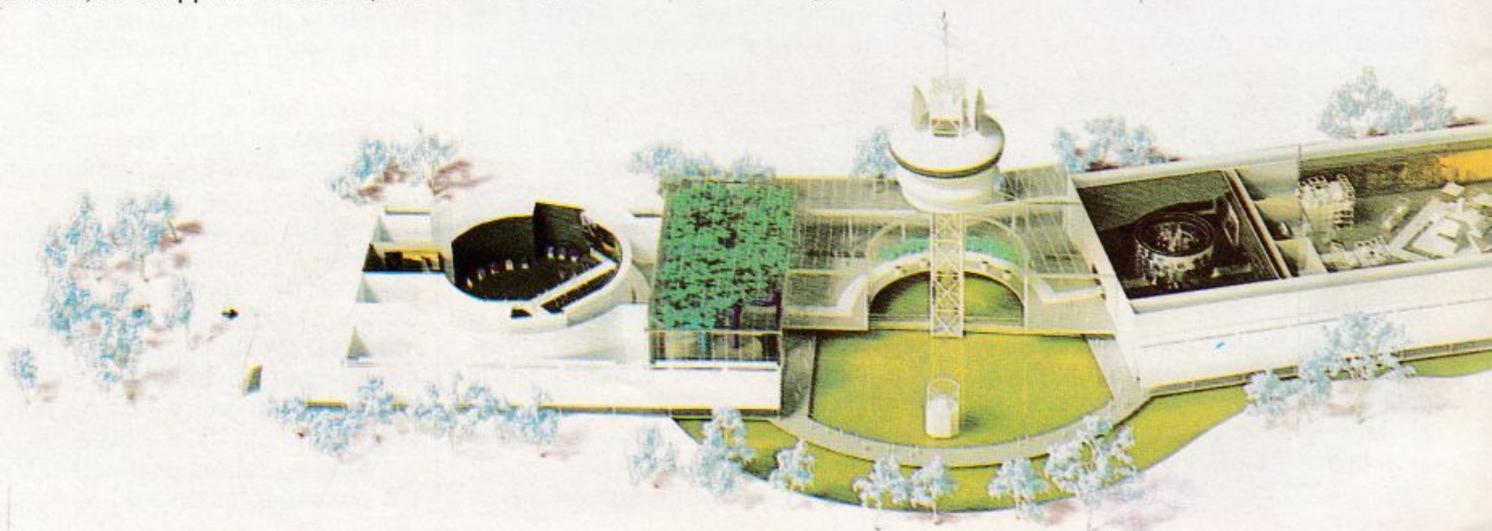
la città di Mazinga e del treno magnetico

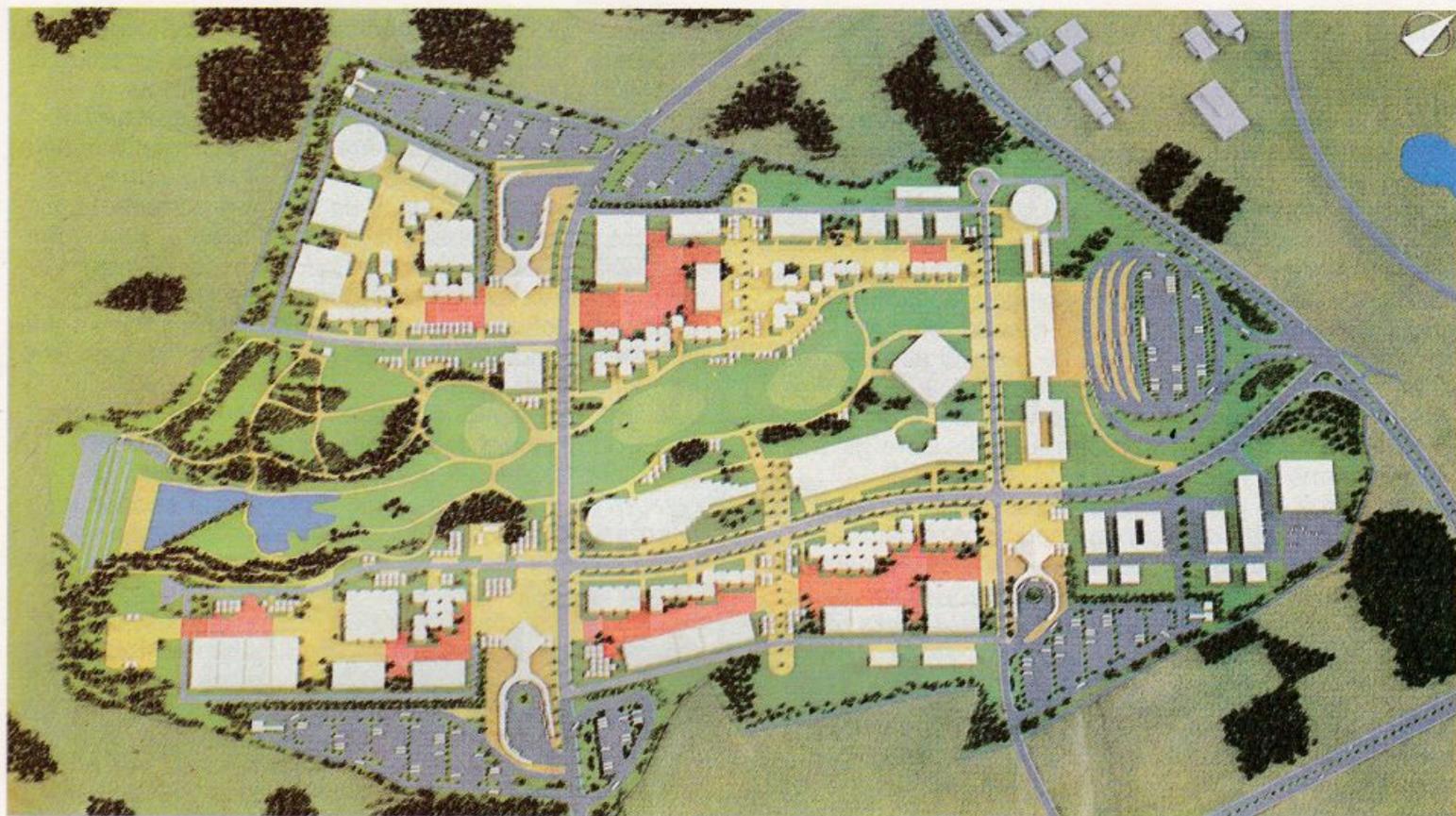
di Giuseppe Breveglieri



Le Esposizioni Universali sembrano essere il debole delle civiltà occidentali. La prima Esposizione di questo tipo si svolse a Londra nel 1851, poi Parigi, divenuta la capitale alla quale tutto il mondo guardava, ne organizzò una valanga: ben sei tra il 1855 e il 1937. Un paio di altre Esposizioni Internazionali si svolsero a Bruxelles e Montreal nel 1958 e nel 1967 e una, finalmente, in Giappone a Osaka, nel 1970. Adesso i

giapponesi ci riprovano con l'Expò di Tsukuba, la cittadella della scienza. La Fiera della tecnologia di Tsukuba è certamente la più imponente e avveniristica manifestazione internazionale mai avvenuta in questo campo. In questa sorta di Metropolis del 2000, il Giappone ha investito cinquemila miliardi di lire. Tsukuba è anche una sfida all'Occidente tecnologizzato; una sfida nella corsa a un domani che





è già oggi e che i giapponesi intendono ormai solo come una gara tesa a sviluppare quantità enormi di tecnologia.

Su una superficie di 102 ettari, a una cinquantina di chilometri da Tokyo, Tsukuba è dal 1966 un centro nazionale per la ricerca scientifica e tecnologica e per l'istruzione in questi particolari settori. Prima era un centro agricolo; adesso a Tsukuba hanno la loro sede un terzo degli istituti nazionali di ricerca del Giappone. Il 50% dei ricercatori nipponici abita ormai stabilmente qui. Tsukuba aspira a diventare un centro accademico unico nel suo genere, di carattere internazionale, per lavori di ricerca di altissimo livello. Era quindi la sede ideale per ospitare questa esposizione della scienza e della tecnologia al servizio dell'uomo, nella casa e nel suo ambiente.

Una cinquantina di paesi di cinque continenti e una quarantina di organizzazioni internazionali, oltre alle

rappresentanze nipponiche, hanno creato a Tsukuba un pianeta del futuro in sedicesimo.

Tra l'altro c'è a Tsukuba l'HSST, un treno a propulsione lineare, un avanzato sistema di trasporto ferroviario che si muove scivolando su una pista a un centimetro da terra. La sua velocità massima è di 300 Km/ora.

Invece di correre sui binari questo treno rivoluzionario scivola senza intoppi, sollevato da terra per effetto della forza magnetica. In assenza di attrito si possono raggiungere elevate velocità e il rumore viene quasi completamente eliminato. Il prototipo in mostra a Tsukuba — la pista è lunga solo 350 metri e la velocità, per ragioni evidenti, limitata — è il frutto delle ricerche fatte dalla JAL, la compagnia aerea nipponica, ma anche la JNR, le ferrovie giapponesi, hanno realizzato un loro treno super-rapido, lo «Shinkansen» che, in prova, è arrivato alla fantastica velocità di 517



Theme Pavillon image sketch / The variety of nature



km/ora, senza equipaggio, e a 262 km/ora con passeggeri a bordo.

La differenza tra i due treni è data dal fatto che le Ferrovie Giapponesi usano un sistema a magneti superconduttori raffreddati con elio liquido, mentre la JAL utilizza dei magneti normali.

Il mondo fantascientifico di Tsukuba, dove la tecnologia giapponese ovviamente detta legge, è popolato anche da un esercito di automi che sanno fare tutto: suonano Bach, recitano Shakespeare, dipingono, giocano a palla, ballano. Tra loro c'è il più grande robot del mondo, che si chiama Fanuc; 25 metri di altezza per 25 tonnellate di peso.

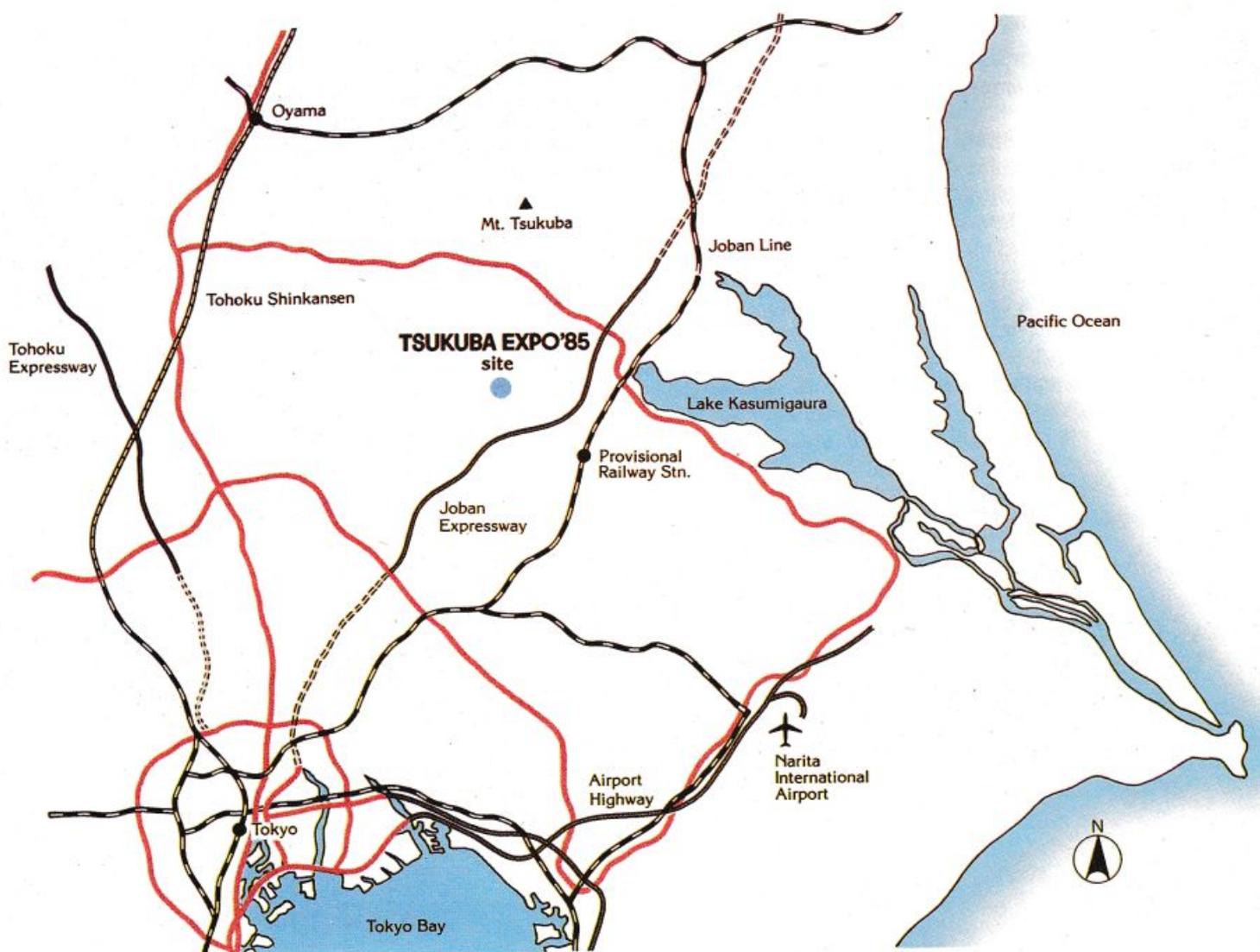
Ma anche gli automi sono a loro volta dominati dalla orwelliana figura di Jumbo Tron, un televisore gigante con schermo di mille metri quadri, 40 metri di lunghezza e 25 di altezza, realizzato dalla Sony.

Tsukuba ha una sua filosofia che traspare dal modo con il quale la cittadella della scienza è stata realizzata e condotta: scienza e tecnologia come veicolo per far circolare liberamente idee, amicizia e tolleranza tra i popoli.

Nelle intenzioni degli organizzatori Tsukuba deve diventare una pietra miliare nella storia dell'umanità. È ben vero che gli organizzatori di questo tipo di manifestazioni lo hanno sempre detto, in apertura di ogni Fiera Internazionale, ma per Tsukuba tutto questo è, forse, un poco più vero.

La scienza e la tecnologia, un tempo duramente contestate per incomprensione o per paura, hanno qui in Giappone la loro definitiva consacrazione. In poche parole dalla scienza e dalla tecnologia l'umanità aspetta le soluzioni per risolvere problemi annosi come la fame nel mondo, la siccità, la sovrappopolazione, per fare solo pochi esempi.

Tsukuba può sembrare anche un enorme Luna Park, una Coney Island della realtà, una Disneyland per bambini cresciuti, un grande show, per capirci, che celebra e consacra l'avvento, sulla ribalta economica internazionale, di quel fenomeno incredibile che si chiama Giappone; ma è anche un segno di speranza, per un domani dove l'atomo avrà solo influenze benefiche, per dimenticare altri segni, incancellabili, lasciati dalla cecità dell'uomo a Hiroshima e Nagasaki.





PARTE QUINTA

SCUOLA

E

COMPUTER

(a cura di Paolo Ciancarini)

Il software degli home-computer: UN QUADRO GENERALE

Negli articoli scorsi abbiamo passato in rassegna l'hardware disponibile per i sistemi home-computer, in relazione con le funzionalità più interessanti nel campo della scuola.

Spero di essere riuscito a chiarire che un computer va scelto e comprato non tanto per le caratteristiche della base (processore, memoria), quanto per il sistema completo che si può costruire: il più bell'elaboratore del mondo è abbastanza inutile senza periferiche adeguate. D'altra parte, tutto l'hardware del mondo è inutile, se non sapete come usarlo, come «parlargli», per fargli fare ciò che vi occorre: come vedremo, il software è il «soffio vitale» del vostro sistema.

È dunque venuto ora il momento di parlare di ciò che nella lezione introduttiva definimmo «l'anima» di qualsiasi elaboratore elettronico: il software, cioè l'insieme dei programmi che «girano» sugli home-computer.

Metteremo naturalmente in risalto quei tipi di programmi che più ci sembrano utili e interessanti in ambito scolastico. Non sarà un limite particolarmente forte: vedrete che parleremo di quasi tutto il software che esiste (non esclusi i videogiochi!).

Noterete nel seguito che è stata fatta una scelta di fondo: noi parleremo per lo più di software commerciale. Questo per vari motivi:

— è più intercambiabile: spesso lo stesso programma

esiste per più calcolatori di marche diverse. In particolare, anche se cercheremo di fare una panoramica generale, naturalmente il nostro punto di riferimento sarà il software per i due home-computer attualmente più diffusi in Italia: il Commodore 64 e la serie Sinclair. Sappiate che i programmi esistenti per questi due sistemi sono complessivamente più di 10.000;

- è meglio valutabile, perché è più facile fare confronti tra programmi analoghi di ditte concorrenti;
- è più generale: i programmi commerciali sono fatti apposta per risolvere il maggior numero di esigenze, diverse ma dello stesso tipo.

Classificazione

È innanzitutto importante avere ben chiare le differenze di funzionalità tra i vari tipi di software esistenti.

Premesso che parleremo unicamente di programmi per home-computer, possiamo grosso modo distinguere le seguenti classi:

1. Il software di base

Costituisce il corredo indispensabile per controllare tutte le potenzialità della macchina. Alcuni dei programmi di questa classe sono «residenti» in memoria centrale, su ROM: così all'accensione diventano immediatamente attivi. Possiamo individuare due sottocategorie:

a) *il sistema operativo* e le sue estensioni. È il programma che gestisce tutte le comunicazioni tra le



varie componenti del vostro sistema. Quando, ad esempio, su un C64 scrivete

LOAD"GHOSBUSTERS",8

ordinate al sistema operativo di mettersi in comunicazione col disk drive (unità 8), e di chiedere al DOS (Disk Operating System, sistema operativo dell'unità a dischi) di trasmettere il codice che costituisce il programma GHOSTBUSTERS. Altre funzionalità tipiche sono l'esecuzione di un programma (RUN), o il salvataggio su memoria esterna (SAVE).

b) *il software di sistema*: è composto da una serie di programmi, ciascuno adibito ad una specifica funzione. Ne citiamo qualcuno (li descriveremo con cura nei prossimi articoli):

- editor dei programmi;
- monitor per linguaggio macchina;
- assembler;
- programmi per telecomunicazioni (per pilotare modem);
- programmi di copia (sì, parleremo anche di questi!).

2. Programmi per programmare: interpreti e compilatori

Sono i programmi che permettono di programmare il vostro elaboratore in maniera «umana». Come sapete, la vera lingua dei computer è in codice binario: i programmi dovrebbero essere costituiti da smisurate sequenze di 0 e 1. E, in effetti, così si faceva agli albori dell'informatica (non più di 30 anni fa!).

È possibile per fortuna scrivere dei programmi che traducono in codice binario altri programmi scritti in linguaggi in qualche modo più adatti agli uomini. Per questo motivo questi linguaggi sono detti di alto livello. I programmi traduttori sono di due tipi:

a) *Interpreti*: sono programmi che controllano, traducono ed eseguono istruzione per istruzione il programma di partenza (detto codice sorgente). Linguaggi come il BASIC, il LOGO, il FORTH, il PILOT, il COMAL, il PROLOG, hanno bisogno di un interprete.

Programma sorgente	Interprete	Esecuzione
--------------------	------------	------------

b) *Compilatori*: sono programmi che prima controllano e poi traducono tutto il codice sorgente. Il risultato della traduzione è un programma in codice macchina, detto programma oggetto, che viene poi eseguito a cura del sistema operativo. Esistono compilatori per il BASIC, il FORTH, il PASCAL, il FORTRAN, il COBOL.

Programma sorgente	Compilatore	Programma in codice macchina
--------------------	-------------	------------------------------

3. *Programmi per elaborazione di testi*: detti pure word processor, trasformano l'elaboratore in una macchina per scrivere dalle funzionalità eccezionali, superiori a quelle di qualsiasi macchina per scrivere «reale». Chiaramente hanno bisogno di una stampante per poter produrre output su carta.

4. *Software grafico*: per creare ed eventualmente riprodurre su carta (tramite stampante o plotter) disegni, figure, progetti, grafici, istogrammi.

5. *Software musicale*: per creare e riprodurre suoni, parole, musica.

6. Applicazioni specifiche

Facciamo rientrare in questa categoria tutta una serie di cosiddetti «programmi d'utilità», quali i seguenti:

- a) basi di dati: cioè programmi per gestire grossi archivi di dati;
- b) fogli elettronici (o spreadsheet): parleremo di Multiplan, e vedremo come usare i programmi di questo tipo anche per applicazioni non strettamente finanziarie;
- c) software matematico;
- d) software finanziario;
- e) software scientifico in genere.

7. *Giochi educativi*: questi sono i programmi che dovrebbero «insegnare divertendo». Abbiamo incluso in questa categoria anche giochi veri e propri, come gli adventures, per le loro evidenti applicazioni nel campo dello studio delle lingue:

- giochi matematici;
- giochi enciclopedici;
- simulazioni;
- giochi d'avventura in lingue straniere.

Parlare del software per home computer in termini così generali è un po' come tentare di fare il riassunto di un'enciclopedia. Noi comunque ci proveremo, nella convinzione di fare un'utile operazione d'informazione e divulgazione nei confronti di tutti i lettori. Nei prossimi articoli, ci sforzeremo soprattutto di dimostrare che già oggi gli home-computer sono assai più che semplici console per videogiochi.

Naturalmente siamo a disposizione dei lettori per soddisfare qualsiasi esigenza specifica, ed eventualmente per segnalare le idee più interessanti che ci verranno proposte. Scriveteci!



PARTE QUINTA

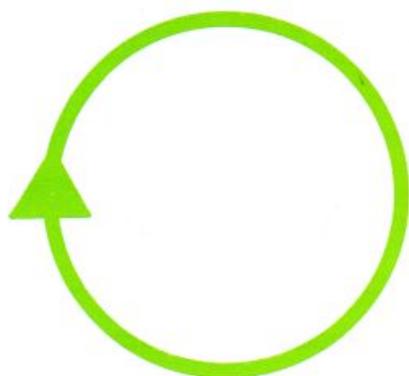
I SPEAK LOGO

A cura di Paolo Ciancarini

Funzioni ricorsive

Abbiamo ormai una certa familiarità con due dei concetti chiave del linguaggio LOGO: le *procedure ricorsive* come la procedura CERCHIO:

```
PER CERCHIO
  AVANTI 1
  DESTRA 1
  CERCHIO
FINE
```



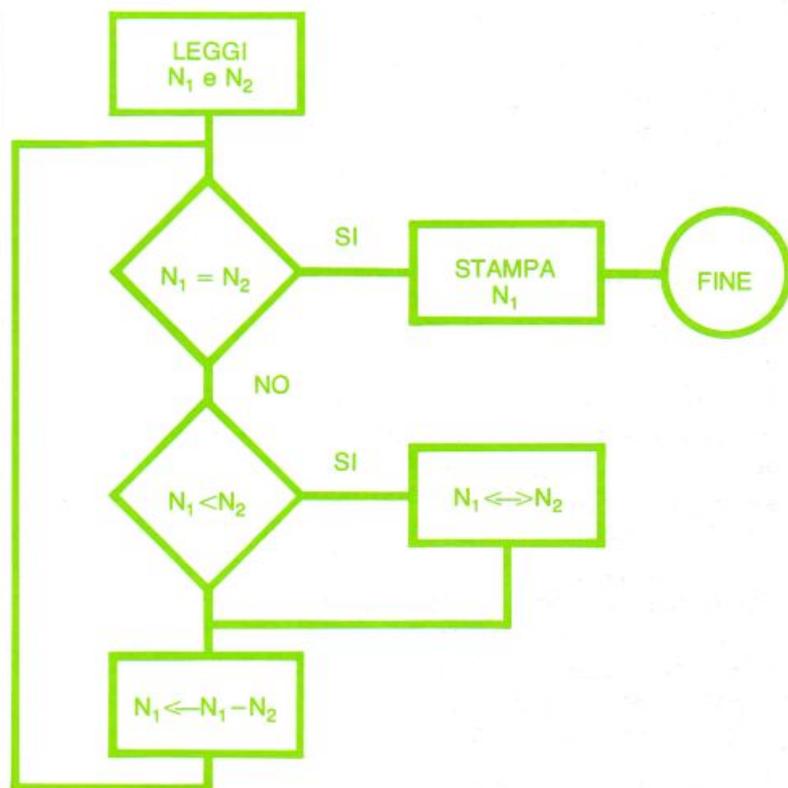
e le *procedure-funzione*, cioè quelle che *riportano* un risultato, come la procedura PITAGORA:

```
PER PITAGORA :CATETO1 :CATETO2
  RIPORTA RADQ (:CATETO1 * :CATETO1 + :CATETO2 *
    :CATETO2)
FINE
```

Vogliamo provare a mettere insieme queste due idee? Stavolta parleremo infatti di *procedura-funzioni ricorsive*. Chi di voi ha letto libri o seguito corsi introduttivi alla programmazione avrà visto, come primo esempio di algoritmo (ricordate? Algoritmo significa «metodo preciso ed effettivo per risolvere un certo problema»), il seguente:

1. siano dati due numeri; chiamateli N1 e N2;
2. se sono uguali, l'algoritmo termina e il risultato è N1;
3. se N1 è più piccolo di N2, scambiateli: ovvero N1 diventa N2 e viceversa;
4. calcolate la differenza N1-N2: chiamatela N1;
5. ricominciate da capo l'algoritmo con il nuovo N1 e il vecchio N2.

Una forma forse più famosa per descrivere algoritmi è quella dei diagrammi a blocchi. Vediamola, per confronto:



Questo algoritmo è noto sotto il nome di Algoritmo Euclideo, perché venne descritto per la prima volta dal grande matematico Euclide. Per capire come funziona, proviamo ad eseguirlo su due numeri qualsiasi (per esempio 12 e 30):

N ₁	N ₂	
12	30	all'inizio
30	12	per il passo 3
18	12	per il passo 4
6	12	per il passo 4
12	6	per il passo 3
6	6	fine, per il passo 2: il risultato è 6.

Facendo altre prove, vi accorgete che il numero che risulta alla fine è esattamente il massimo comun divisore dei due numeri di partenza. Notate, tra l'altro, che questo algoritmo NON fa uso di moltiplicazioni e divisioni. Come vedete, al passo 5 c'è un'istruzione che somiglia parecchio ad una chiamata ricorsiva dell'algoritmo stesso, esattamente come farebbe una procedura LOGO.

Vediamo dunque una procedura-funzione LOGO che descrive l'algoritmo euclideo:



```

PER MCD :N1 :N2
  SE :N1 = :N2 ALLORA RIPORTA :N1
  SE :N1 < :N2 ALLORA RIPORTA MCD :N2 :N1
  RIPORTA MCD (:N1 - :N2) :N2
FINE

```

Un esempio d'invocazione:

```

MCD 12 30
RISULTATO: 6

```

Il condizionale

Il comando SE...ALLORA... si chiama condizionale: quel che segue la parola-chiave ALLORA (sulla stessa riga) viene eseguito se è vera la condizione compresa tra SE e ALLORA. Il comando condizionale è uno dei più importanti in qualsiasi linguaggio di programmazione.

Notate che nella procedura MCD il primo SE...ALLORA... entra in azione solo se i due numeri dati come argomenti della procedura sono uguali. In tal caso la procedura-funzione si ferma (perché così funziona RIPORTA), dando come risultato :N1. Questa istruzione è dunque l'equivalente del passo 2.

Il secondo SE...ALLORA... entra in azione solo se il primo argomento, :N1, è più piccolo del secondo, :N2. In tal caso il risultato dell'invocazione originaria MCD :N1 :N2 viene calcolato invocando ricorsivamente MCD :N2 :N1, scambiando cioè gli input. In pratica questo significa che il massimo comun divisore di 8 e 12, ad esempio, si calcola riapplicando daccapo l'algoritmo, avendo però come input 12 come :N1, e 8 come :N2. Notate che questo chiaramente equivale anche a stabilire in astratto che l'operazione di massimo comun divisore è commutativa, cioè che

$$\text{MCD}(n,m) = \text{MCD}(m,n)$$

Se infine nessuna delle due condizioni è vera, si chiama ricorsivamente la procedura-funzione MCD su due nuovi argomenti: il primo è costituito dalla differenza di :N1 e :N2, che diventa il nuovo :N1; :N2 invece rimane intatto. Nel caso di 12 e 8, per esempio, si passa a calcolare il massimo comun divisore di 4 e 8. Infatti si può dimostrare che

$$\text{MCD}(n,m) = \text{MCD}(n-m,m) \text{ se } n > m$$

Naturalmente la sequenza di chiamate ricorsive termina solo quando i due numeri sono uguali, essendo il massimo comun divisore, in tal caso, ovvio:

$$\text{MCD}(n,n) = n$$

Bello, vero?

Usando la procedura MCD è facile calcolare il minimo comune multiplo: vale infatti una formula, assai poco nota ma elegantissima, che permette di calcolare il minimo comune multiplo di due numeri a partire dal massimo comun divisore:

$$\text{mcm}(n,m) = \frac{n * m}{\text{MCD}(n,m)}$$

```

PER MCM :N1 :N2
  RIPORTA :N1 * :N2 / MCD :N1 :N2
FINE

```

Moltissime funzioni utili possono venir calcolate in LOGO mediante procedure ricorsive. Il fattoriale, per esempio, che una funzione matematica definita come

$$\text{fattoriale}(n) = 1 * 2 * 3 * \dots * (n-1) * n$$

Per esempio, il fattoriale di 1 è 1, quello di 2 è 2, quello di 3 è 6, quello di 4 è 24, e così via (è una funzione che diventa rapidamente molto grande, come sa chi gioca al Totocalcio!).

Vediamo la corrispondente procedura-funzione LOGO che calcola il fattoriale di qualsiasi numero:

```

PER FATTORIALE :N
  SE :N = 1 ALLORA RIPORTA 1
  RIPORTA :N * FATTORIALE (:N - 1)
FINE

```

In un comando condizionale potete verificare condizioni di 3 tipi:

a) condizioni matematiche semplici:

```

:A > 0   il valore di A è maggiore di 0?
:B = :C  il valore di B è uguale a quello di C?
:D < 3   il valore di D è minore di 3?

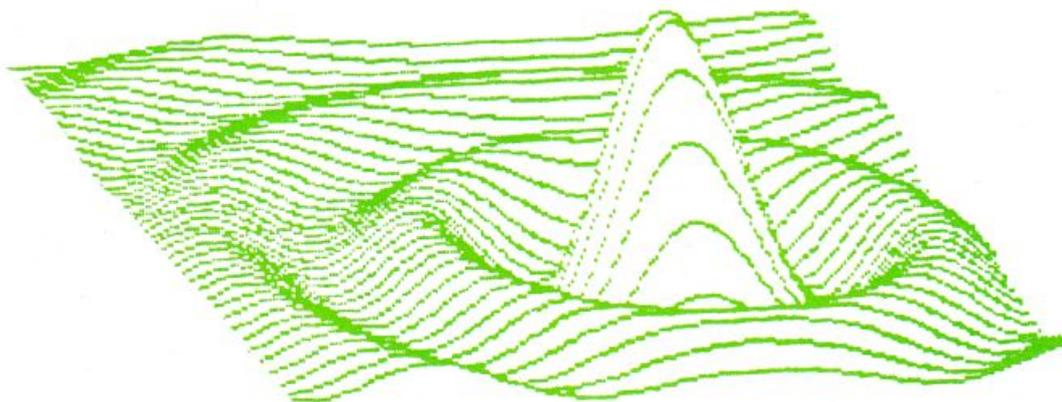
```

b) condizioni complesse:

```

NON (:A = :B)           A è diverso da B?
TUTTIVERI? (:A > 0 :B > :A)  è vero che A>0 e contemporaneamente B>A?
UNOVERO? (:A > 0 :B < 4)    almeno uno tra A>0 e B<4 è vero?

```





c) altre condizioni: le vedremo più avanti, quando impareremo ad usare le parole e le liste del LOGO.

Nota per i BASIC-dipendenti: NON corrisponde all'incirca al NOT del BASIC, TUTTIVERI? all'AND, mentre UNOVERO? è il corrispondente dell'OR.

Vediamo qualche esempio.

Supponiamo che :A valga 3, :B valga 0 e :C valga 6. Allora:

```
NON :A = 0 è vero
NON :C > 0 è falso
TUTTIVERI? (:A > 0 :B = :C) è falso
TUTTIVERI? (:A > :B :C > 0) è vero
UNOVERO? (:A > 0 :B = :C) è vero
UNOVERO? (:B > :A :B = :C) è falso
```

Proviamo a scrivere una procedura che decide se un certo anno è bisestile o no. Le regole sono queste (notate che diamo praticamente un algoritmo!):

Sia N l'anno. Se N è divisibile per 4, è bisestile, a meno che N sia divisibile per 100: allora non è bisestile, a meno che N sia divisibile per 400: allora è bisestile.

Ad esempio, 1984 è bisestile, come 2000; invece 1700, 1800 e 1900 non lo sono stati. La ragione va ricercata nel fatto che un anno solare dura 365 giorni e «un pezzetto», che è inferiore di poco a 6 ore: esattamente 5h, 48m e 46s.

Un tempo queste regole non valevano, ed alla lunga le

stagioni «sballavano». Pensate che, per far tornare i conti delle festività religiose (la Pasqua è legata all'equinozio di primavera), circa 4 secoli fa per ordine del Papa vennero letteralmente saltati 11 giorni del calendario giuliano (introdotto da Giulio Cesare) allora in vigore.

Ma torniamo al nostro programma. Per maggiore chiarezza, scriviamo prima una procedura DIV?, che decide se un certo numero N è divisibile per un altro numero D. Ricordiamo che la primitiva RESTO calcola il resto della divisione intera tra due numeri:

```
PER DIV? :N :D
  RIPOSTA (RESTO :N :D = 0)
FINE
```

```
PER BISESTILE? :ANNO
  RI TUTTIVERI? (DIV? :ANNO 4) (UNOVERO? (DIV? :ANNO 100)
  (NON DIV? :ANNO 400))
FINE
```

Diamo, per concludere, qualche esempio d'invocazione della procedura:

```
BISESTILE? 1984
RISULTATO: VERO
```

```
BISESTILE? 2000
RISULTATO: VERO
```

```
BISESTILE? 1985
RISULTATO: FALSO
```

```
BISESTILE? 1900
RISULTATO: FALSO
```

STUDIO PICCA ROMA

SOPIN

società per l'informatica spa

**INFORMATICA
TELEMATICA
OFFICE AUTOMATION
SICUREZZA FISICA E
LOGICA
RICERCA APPLICATA
FORMAZIONE E
ADDESTRAMENTO**

ROMA VIA DEL SERAFICO 200 TEL 5477



Z80

LINGUAGGIO MACCHINA

a cura di Massimo Momo

Nell'articolo precedente abbiamo introdotto lo Z-80, parlando di due dei tre Bus che gli permettono di scambiare dati con il mondo esterno (BUS dei DATI ad 8 bit e BUS degli INDIRIZZI a 16 bit) ed inoltre abbiamo accennato alla presenza, all'interno del microprocessore, di diversi registri ad otto bit, il più importante dei quali è l'Accumulatore; abbiamo anche puntualizzato il fatto che lo Z-80 esegue instancabilmente ISTRUZIONI.

Abbiamo concluso il suddetto articolo domandandoci dove trova lo Z-80 tali istruzioni e come fa, una volta terminata una, a passare a quella successiva. Per rispondere a questa domanda dobbiamo parlare di un registro speciale contenuto nel microprocessore chiamato «PROGRAM COUNTER» (Contatore di programma, indicato con la coppia di lettere PC) che, a differenza dell'Accumulatore e dei registri ad otto bit accennati precedentemente, è un registro a sedici bit, in quanto non è utilizzato per memorizzare dati (che abbiamo visto essere ad otto bit) ma piuttosto indirizzi (che abbiamo visto essere a sedici bit): in particolare, il Program Counter contiene l'indirizzo della cella di memoria in cui è contenuto il codice della istruzione che deve essere eseguita.

Spieghiamoci con un esempio: supponiamo di voler far eseguire allo Z-80 un'istruzione di «caricamento dell'Accumulatore» con il valore 10 (ricordiamo, fino alla nausea, che, essendo l'Accumulatore un registro ad otto bit, esso può contenere un dato compreso tra 0 e 255). In pratica vogliamo caricare nell'Accumulatore il valore 10.

Per far questo, supponiamo di essere riusciti a caricare nella cella di memoria RAM, per esempio di indirizzo 27000, il numero 62 mentre nella cella successiva (evidentemente di indirizzo 27001) il valore 10.

Supponiamo inoltre di avere il registro Program Counter caricato con il valore 27000 (ricordiamo che, essendo quest'ultimo un registro a 16 bit, esso può contenere numeri compresi tra 0 e 65535).

Abbiamo ora tutti gli elementi per capire quello che avviene dentro lo Z-80: esso depositerà sul Bus degli Indirizzi il contenuto del Program Counter ed eseguirà automaticamente un «ciclo di acquisizione di una istruzione»; questo significa che sul «BUS di

CONTROLLO» (terzo bus presente in FIG. 1 dell'articolo precedente), lo Z-80 depositerà opportuni segnali tali da informare i dispositivi esterni che esso ha intenzione di leggere (cioè di acquisire) il dato contenuto sul bus dei dati.

Quindi i segnali presenti sul Bus di Controllo, uniti all'indirizzo presente sul Bus degli Indirizzi (come abbiamo visto esso è 27000) farà sì che la decodifica della RAM sia in grado di abilitare esclusivamente la cella di memoria di indirizzo 27000 che, chiamata in causa, depositerà a sua volta sul Bus dei Dati il proprio contenuto di otto bit (precedentemente posto pari a 62); lo Z-80 leggerà tale valore e, essendo in corso un ciclo di acquisizione di un'istruzione, lo depositerà in un suo particolare registro (detto registro delle Istruzioni, ma comunque poco importante per il programmatore, perché non è possibile utilizzarlo direttamente) che ha il compito di decodificare l'istruzione, cioè capire che cosa si richiede al microprocessore.

Contemporaneamente lo Z-80 incrementerà di uno il contenuto del registro PC che verrà così a contenere il valore 27001.

Sottolineiamo che tutto questo, per nostra fortuna, avviene automaticamente, cioè è «trasparente» al programmatore.

A questo punto il microprocessore avrà decodificato l'istruzione il cui codice è 62 ed avrà capito che è nostra intenzione caricare nell'accumulatore un dato, che però lo Z-80 ancora non conosce: ebbene esso si aspetta di separare tale dato nella cella di memoria successiva a quella in cui ha letto il codice dell'istruzione: nel nostro caso, se lo aspetta nella cella di indirizzo 27001; non a caso precedentemente abbiamo caricato in tale cella il valore 10, cioè proprio il dato che vogliamo caricare in Accumulatore.

Lo Z-80 effettuerà a questo punto un «ciclo di lettura» allo scopo di leggere il contenuto della cella di memoria il cui indirizzo è immagazzinato nel Program Counter: a tale fine esso depositerà sul bus degli indirizzi il contenuto del Program Counter e sul bus di controllo opportuni segnali atti ad avvertire i dispositivi esterni sulla sua volontà di leggere dati; tali segnali, opportunamente decodificati dalla decodifica della RAM, faranno sì che solo la cella di memoria di

indirizzo 27001 depositerà sul bus dei dati il proprio contenuto (pari naturalmente a 10). Il microprocessore potrà quindi leggere tale dato ed avrà tutte le informazioni necessarie per completare l'istruzione; a questo punto lo Z-80 caricherà nell'accumulatore il valore 10, incrementerà ancora il Program Counter (che conterrà 27002) e sarà di nuovo pronto ad eseguire una nuova istruzione, il cui codice dovrà essere contenuto nella cella 27002.

Abbiamo così visto come viene eseguita un'istruzione; nel caso particolare questa era «a due bytes», cioè occupava due celle di memoria: infatti nella cella di memoria 27000 vi era il «CODICE OPERATIVO» dell'istruzione (62) mentre in quella successiva vi era il dato (10). Nel set di istruzioni dello Z-80 vi sono istruzioni ad un solo byte (quello contenente il codice operativo) che non necessitano di alcun dato, istruzioni a due bytes (uno per il codice operativo e uno per il dato) che necessitano di un dato ad otto bit, ed istruzioni a tre bytes (uno per il codice operativo e due per il dato) che necessitano di un dato a sedici bit, cioè un indirizzo. Per la verità vi sono anche istruzioni a quattro bytes, che però sono del tutto analoghe a quelle a tre bytes, con la sola differenza che il codice operativo occupa due bytes invece di uno: quindi, anche per queste ultime, il dato è un indirizzo.

Comunque torneremo ampiamente sull'argomento man mano che analizzeremo le varie istruzioni.

Passiamo ora a vedere i diversi registri presenti nello Z-80 (vedi FIG. 1):

- Il registro A (Accumulatore) è il più importante in quanto è quello utilizzato in tutte le istruzioni aritmetiche (somma, differenza, scorrimento, ecc.)
- Il registro F (Flags) è un registro molto particolare, nel quale ognuno degli otto bit serve per indicare particolari condizioni avvenute a causa di un'istruzione, come un risultato maggiore, il minore oppure uguale a zero e così via. Tali bit possono essere verificati mediante opportune istruzioni simili alla «IF» del BASIC
- Vi sono inoltre i registri B, C, D, E, H e L che sono tutti ad 8 bit e possono essere utilizzati più o meno indifferentemente. Come si vede dalla FIG. 1 vi sono due set diversi di registri ad 8 bit (il secondo set è composto da altri 8 registri aventi lo stesso nome dei corrispondenti registri del primo set con a fianco il pedice) che possono essere utilizzati alternativamente mediante opportune istruzioni che vedremo in seguito
- Il registro PC (Program Counter) è stato già analizzato
- Il registro SP (Stack Pointer), a 16 bit come il PC, è impiegato per gestire una CATASTA (STACK), importantissimo strumento che ci permette di utilizzare le subroutine con un'istruzione equivalente ad «GOSUB» del BASIC
- Vi sono ancora i due registri IX e IY a 16 bit chiamati «REGISTRI INDICE» in quanto possono essere utilizzati come PUNTATORI di tabelle e liste di dati
- Vi sono infine due registri ad 8 bit molto particolari: il registro I (Interrupt) viene utilizzato per la gestione delle «INTERRUZIONI» di cui parleremo molto più in là; il registro R (Refresh) viene utilizzato per

«rinfrescare» le memorie di tipo dinamico, che però esulano dalla nostra trattazione.

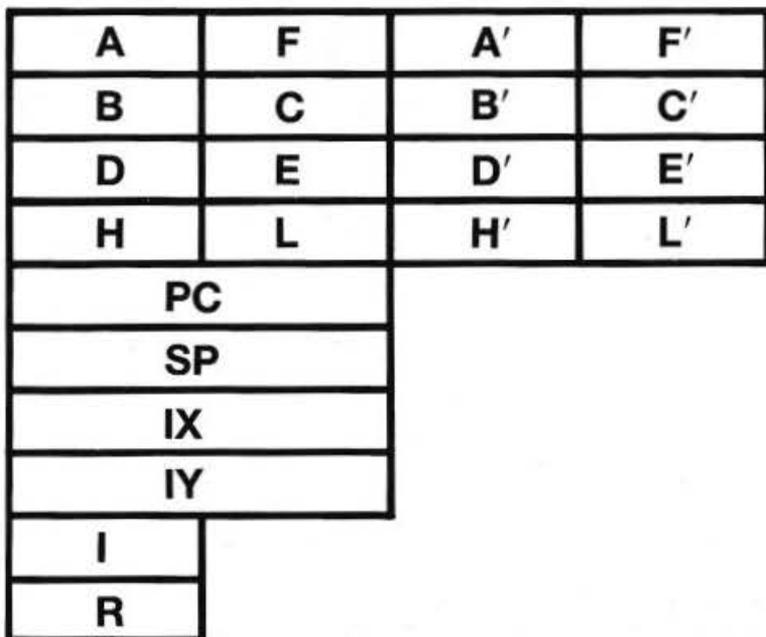


FIGURA 1

Per poter vedere come le diverse istruzioni agiscono sui registri dello Z-80, presentiamo questa volta un programma per lo Spectrum (al solito misto tra Basic e linguaggio macchina) chiamato «VEDIREGIS» il cui scopo è di analizzare e modificare i vari registri. Una volta lanciato il programma con RUN, esso offrirà sullo schermo la situazione completa dei registri dello Z-80 dando, per i registri ad otto bit, il contenuto in decimale ed in binario. Il registro «F», come già evidenziato, ha un'importanza particolare e per questo viene visualizzato separatamente, con il significato di ciascuno dei suoi otto bit; torneremo a parlare dettagliatamente del registro F in un articolo successivo.

Dopo di questo, il programma continua chiedendo la funzione desiderata; sono possibili diverse scelte:

- È possibile modificare uno qualsiasi dei registri (escluso il registro F) inserendo il nuovo valore subito dopo il nome del registro stesso; così A10 carica in Accumulatore il valore 10, PC30000 carica nel Program Counter il numero 30000 e così via
- Premendo il tasto «F» si esce dal programma
- È anche possibile analizzare i registri dopo l'esecuzione di una routine in L.M.; a tale scopo è sufficiente concludere la routine con tre bytes (205, 208, 108) che, come vedremo successivamente, corrispondono ad eseguire la chiamata alla subroutine (equivalente al GOSUB del Basic) che inizia all'indirizzo 27856 e che ha il compito di salvare il contenuto dei vari registri prima di ritornare al Basic. Premendo il tasto «R» il programma inizia l'esecuzione della routine in L.M. a partire dall'indirizzo contenuto nel PC
- Per rendere il programma autosufficiente, è prevista anche la funzione «IN» che permette di inserire in memoria i codici delle istruzioni senza dover ricorrere al programma «CARICAMEMO» presentato nell'articolo precedente, che invece ci sarà indispensabile quando caricheremo routines in



L.M. complete e funzionanti (ricordo che il programma «VEDIREGIS» serve esclusivamente per controllare quello che accade nei registri dello Z-80 dopo l'esecuzione di una o più istruzioni). Dopo aver introdotto «IN» il computer chiede l'indirizzo di partenza della routine che si vuole caricare e si mette in attesa dei vari codici; una volta terminata l'introduzione di questi ultimi, è sufficiente premere il tasto «F» (Fine) per tornare alla visualizzazione dei registri.

Nel prossimo articolo cominceremo ad analizzare una per una le istruzioni dello Z-80; per il momento ve ne anticipiamo tre che serviranno per giocare un po' con il programma «VEDIREGIS». La prima serve per caricare un numero nell'Accumulatore (è composta dal codice operativo 62 seguito dal numero che si vuole memorizzare); la seconda serve per caricare un numero nel registro B (è composta dal codice operativo 6 seguito dal numero); la terza serve per caricare un numero nel registro C (è composta dal codice operativo 14 seguito dal numero). Se decidiamo di far partire la routine dall'indirizzo 28000 dobbiamo inserire (quindi funzione «IN») i seguenti codici:

62	nella cella di indirizzo	28000
numA	nella cella di indirizzo	28001
6	nella cella di indirizzo	28002
numB	nella cella di indirizzo	28003
14	nella cella di indirizzo	28004
numC	nella cella di indirizzo	28005
205	nella cella di indirizzo	28006
208	nella cella di indirizzo	28007
108	nella cella di indirizzo	28008

dove numA, numB e numC sono i numeri che vogliamo caricare nei registri A, B e C rispettivamente. Gli ultimi tre bytes ci permettono di visualizzare i contenuti dei registri dopo l'esecuzione delle tre istruzioni. Una volta caricati tali codici, sarà sufficiente controllare che il registro PC contenga 28000 (in caso contrario, eseguire PC28000) e quindi premere il tasto «R» per l'esecuzione; i tre numeri numA, numB e numC verranno visualizzati nei corrispondenti registri a riprova dell'avvenuto caricamento. Provate a cambiare questi numeri e prendete dimestichezza con il programma. Potreste anche visualizzare i registri alla fine di ciascuna istruzione: per far questo, dovrete introdurre l'insieme dei tre bytes 205, 208, 108 alla fine di ogni istruzione e premere quindi tre volte «R»; comunque questo ve lo lasciamo come esercizio. A presto.

STRUTTURA PROGRAMMA

10	Aggiornamento RAMTOP e forzatura del modo C (caratteri maiuscoli)
20-40	Caricamento in memoria di due routines in L.M. necessarie per la gestione dei registri; il listato Assembler di tali routines è mostrato in LISTING 2
50-100	Inizializzazione dei registri
110	Visualizzazione di tutti i registri

120-250	Attesa di una funzione e conseguente smistamento alle varie parti del programma
260-310	Inserimento nel registro desiderato del nuovo dato
320	Lancio del programma in L.M.
330-380	Subroutine di aggiornamento di un registro a 8 o 16 bit
390-600	Subroutine di visualizzazione di tutti i registri
610-660	Subroutine che fornisce la rappresentazione binaria dei registri ad 8 bit
670-720	Gestione della funzione «IN» per l'inserimento dei codici in memoria

VARIABILI

T =	Indirizzo a partire dal quale vengono salvati i contenuti dei registri
D =	Vettore a 8 componenti in cui viene memorizzata la rappresentazione binaria di un registro a 8 bit
V =	Valore del numero da caricare nel registro
OS =	Indica, con il suo valore, su quale registro si sta lavorando
IPS,IMS =	Rappresentano gli 8 bit, rispettivamente più significativi e meno significativi, di un registro a 16 bit.

```

10 CLEAR 27999: POKE 23658,8
20 DATA 237,115,168,108,237,12
3,154,108,221,42,156,108,253,42,
158,108,42,152,1
08,229,6,4,33,160,108,94,35,86,3
5,213,16,249,225,209,193,241,201
30 DATA 221,34,156,108,253,34,
158,108,34,166,108,237,83,164,10
8,237,67,162,108
,245,193,237,67,160,108,225,34,1
52,108,237,115,154,108,237,123,1
68,108,201
40 FOR I=0 TO 74: READ A: POKE
27819+I,A: NEXT I
50 LET T=27800
60 DIM D(8): LET V=28000: LET
OS=0: GO SUB 330
70 LET V=65365: LET OS=2: GO S
UB 330
74 LET V=0: LET OS=4: GO SUB 3
30
78 LET V=23610: LET OS=6: GO S
UB 330
80 FOR I=1 TO 8
90 LET V=0: LET OS=I+7: GO SUB
330
100 NEXT I
110 GO SUB 390
120 INPUT "INSERISCI FUNZIONE:

```

```

";A#
130 IF A$(1)="R" THEN GO TO 32
0
150 IF A$(1)="F" THEN STOP
160 IF A$(1)="A" THEN LET OS=9
: GO TO 260
170 IF A$(1)="B" THEN LET OS=1
1: GO TO 260
180 IF A$(1)="C" THEN LET OS=1
0: GO TO 260
190 IF A$(1)="D" THEN LET OS=1
3: GO TO 260
200 IF A$(1)="E" THEN LET OS=1
2: GO TO 260
210 IF A$(1)="H" THEN LET OS=1
5: GO TO 260
220 IF A$(1)="L" THEN LET OS=1
4: GO TO 260
230 IF A$(1 TO 2)="PC" THEN LET
OS=0: GO TO 260
240 IF A$(1 TO 2)="SP" THEN LET
OS=2: GO TO 260
244 IF A$(1 TO 2)="IX" THEN LET
OS=4: GO TO 260
248 IF A$(1 TO 2)="IY" THEN LET
OS=6: GO TO 260
250 IF A$(1 TO 2)="IN" THEN GO
TO 670
255 BEEP 0.2,24: GO TO 120
260 IF OS>=8 THEN GO TO 290
270 LET V=VAL A$(3 TO ): IF V<0
OR V>65535 THEN GO TO 250
280 GO SUB 330: GO TO 310
290 LET V=VAL A$(2 TO ): IF V<0
OR V>255 THEN GO TO 250
300 GO SUB 330
310 GO TO 110
320 RANDOMIZE USR 27819: GO TO
110
330 IF OS>=8 THEN GO TO 370
340 LET IPS=INT (V/256): LET IM
S=V-256*IPS
350 POKE (T+OS),IMS: POKE (T+OS
+1),IPS
360 GO TO 380
370 POKE (T+OS),V
380 RETURN
390 CLS : PRINT TAB 4;"SITUAZIO
NE REGISTRI Z80"
393 PRINT : PRINT "PC ";PEEK T+
256*PEEK (T+1)
395 PRINT "SP ";PEEK (T+2)+256*
PEEK (T+3)
400 PRINT "IX ";PEEK (T+4)+256*
PEEK (T+5);TAB 9;"_____BIT__
_____"
410 PRINT "IY ";PEEK (T+6)+256*

```

```

PEEK (T+7);TAB 9;" 7 6 5 4 3
2 1 0"
420 LET V=PEEK (T+9): GO SUB 61
0: PRINT : PRINT "A ";V;TAB 9;
430 FOR I=8 TO 1 STEP -1: PRINT
D(I);" ";: NEXT I
440 LET V=PEEK (T+11): GO SUB 6
10: PRINT AT 8,0;"B ";V;TAB 9;
450 FOR I=8 TO 1 STEP -1: PRINT
D(I);" ";: NEXT I
460 LET V=PEEK (T+10): GO SUB 6
10: PRINT AT 9,0;"C ";V;TAB 9;
470 FOR I=8 TO 1 STEP -1: PRINT
D(I);" ";: NEXT I
480 LET V=PEEK (T+13): GO SUB 6
10: PRINT AT 10,0;"D ";V;TAB 9;
490 FOR I=8 TO 1 STEP -1: PRINT
D(I);" ";: NEXT I
500 LET V=PEEK (T+12): GO SUB 6
10: PRINT AT 11,0;"E ";V;TAB 9;
510 FOR I=8 TO 1 STEP -1: PRINT
D(I);" ";: NEXT I
520 LET V=PEEK (T+15): GO SUB 6
10: PRINT AT 12,0;"H ";V;TAB 9;
530 FOR I=8 TO 1 STEP -1: PRINT
D(I);" ";: NEXT I
540 LET V=PEEK (T+14): GO SUB 6
10: PRINT AT 13,0;"L ";V;TAB 9;
550 FOR I=8 TO 1 STEP -1: PRINT
D(I);" ";: NEXT I
560 PRINT : PRINT : PRINT TAB 1
4;"FLAGS": PRINT
570 PRINT TAB 9;"S Z - H -
P/V N C"
580 LET V=PEEK (T+8): GO SUB 61
0: PRINT AT 19,0;"F ";V;TAB 9;
590 FOR I=8 TO 1 STEP -1: PRINT
D(I);" ";: NEXT I
600 RETURN
610 LET N1=V
620 FOR I=1 TO 8
630 LET N2=INT (N1/2)
640 LET D(I)=N1-2*N2: LET N1=N2
650 NEXT I
660 RETURN
670 CLS : INPUT "INDIRIZZO INIZ
IALE(>=28000):",IN: IF IN<28000
OR IN>65535 THEN
BEEP 0.2,24: GO TO 670
680 LET I=IN
690 INPUT ("CODICE IN ";I;":")
; LINE A#
700 IF A$(1)="F" THEN GO TO 11
0
710 POKE I,VAL A$: PRINT I,PEEK
I
720 LET I=I+1: GO TO 690

```

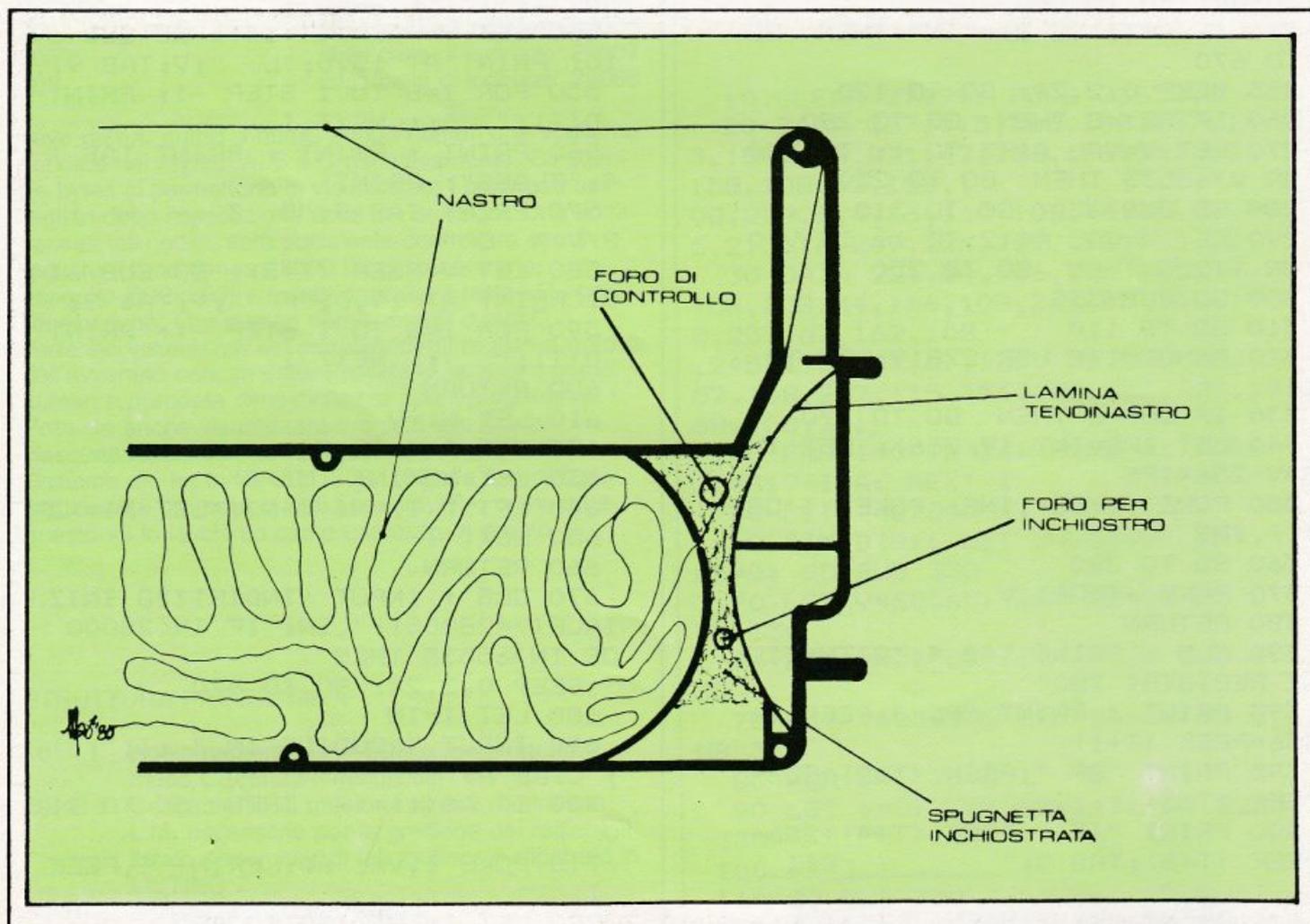


LUNGA VITA... AL NASTRO!

LE PAGINE DELL'ASSISTENZA

Ovvero: come rendere il nastro della propria stampante «immortale».

Scena: dopo aver passato ore ed ore alla stesura di un programma, al momento della stampa vi accorgete che, nonostante l'aver sostituito recentemente la cartuccia, lo scritto, tendente al grigio, è quanto mai poco chiaro e leggibile: che la cartuccia debba essere nuovamente sostituita? Quanto descritto nelle seguenti pagine offre una economica soluzione al «costoso» problema.



DENTRO LA CARTUCCIA

Prendiamo a campione una cartuccia tipo Epson che si trova, naturalmente, su tutte le stampanti Epson e loro similari.

La prima cosa da tenere presente è che la cartuccia può essere aperta e, quindi, il nastro può essere sostituito. La cartuccia deve essere aperta molto attentamente, cercando di non rompere i dentini di plastica di chiusura. Generalmente non si incontrano grandi difficoltà nel localizzare questi dentini che sono ben visibili; difficili da localizzare sono invece i fermi inseriti all'interno della cartuccia.

Attenzione quindi a distaccare le due parti della scocca lentamente e senza troppo forzare, anche se i dentini esterni sono stati liberati.

Aperta la cartuccia vi troverete davanti a metri e metri di nastro arrotolato; volendolo sostituire si dovrà, mentre lo si toglie, fare molta attenzione a come è montato in modo da poter

dall'uso, e quindi non più utilizzabili, sono in realtà «secchi»; e più la stampante rimane accesa — riscaldandosi! — più l'inchiostro del nastro tende a seccarsi. Ed eccoci dunque arrivati al punto del nostro articolo: quanto proposto servirà infatti a «rinfrescare» il nastro della stampante.

Prima di tutto bisogna procurarsi dell'inchiostro liquido adatto al tipo di nastro utilizzato (non dovrebbe essere difficile reperirlo in cartolerie specializzate o forniture per uffici), quindi procedere nel seguente modo:

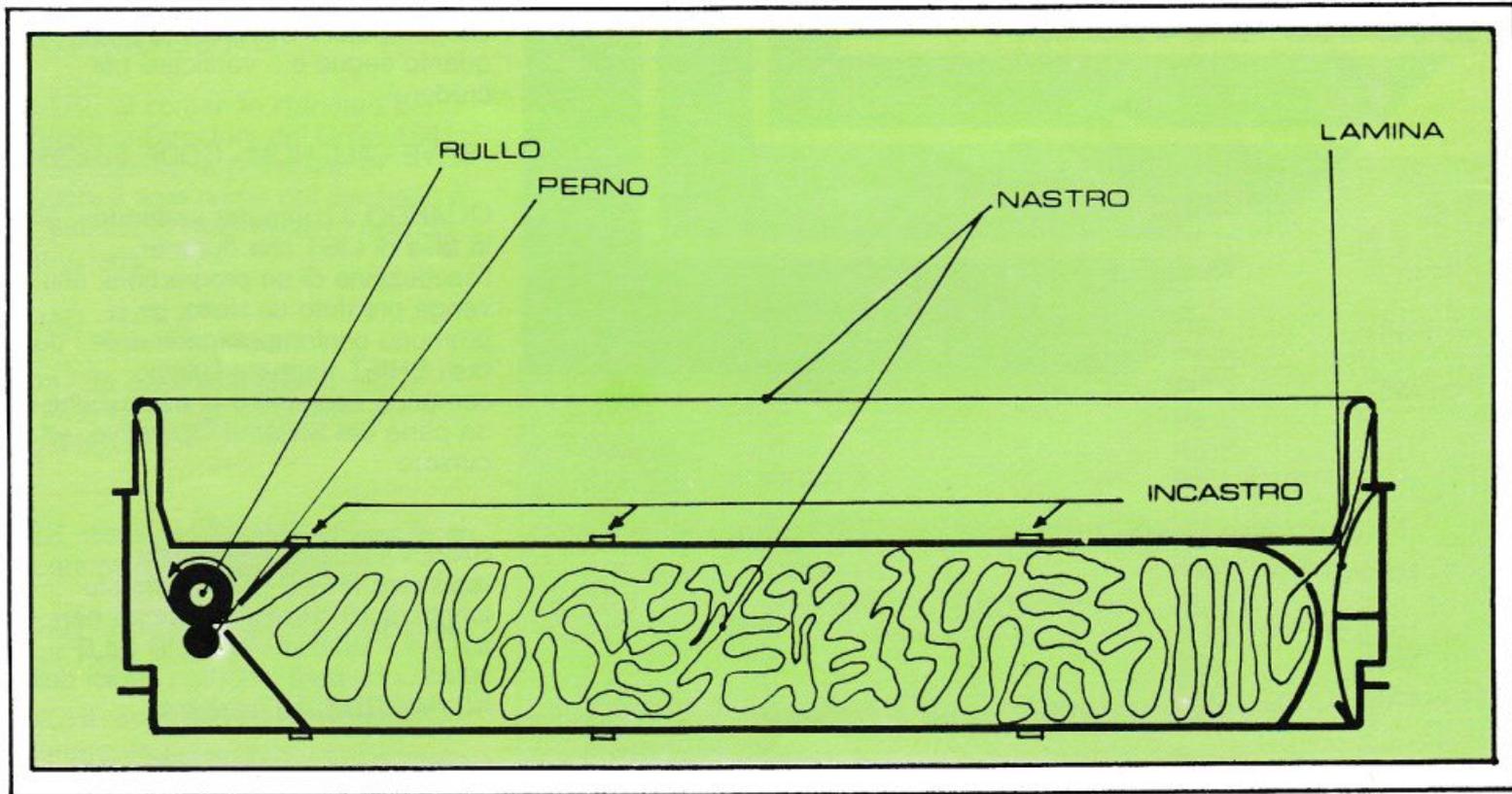
— sulla destra della cartuccia c'è abbastanza spazio per inserire, dopo averlo modellato, un tamponcino di spugna che verrà posizionato in modo che il nastro, prima di «uscire», lo sfiori. Il tamponcino bagnato di inchiostro manterrà il nastro ben umido. Naturalmente non si deve eccedere con la quantità di inchiostro utilizzato per bagnare la

anche rompere.

Giusta annotazione. Ed allora eccoci giunti alla seconda fase dell'operazione «Lunga Vita»: la perforazione della cartuccia per ottenere sulla scocca due piccoli fori (uno atto a controllare il tampone di spugna e l'altro per inserire eventuale altro inchiostro liquido senza dover riaprire nulla). Il foro di dimensione maggiore — circa 6 mm di diametro — sarà quello di controllo: attraverso questo sarete in grado di vedere il tampone e giudicare se è il caso di bagnarlo con altro inchiostro.

Il foro più piccolo servirà, come già detto, per inserire eventuale altro inchiostro: l'inserimento potrà essere effettuato mediante un normale contagocce.

Dovete tenere presente che la spugnetta è di valido aiuto onde evitare che il nastro si secchi; nel caso che questo già lo sia, sarà bene «inumidirlo» direttamente con qualche goccia di inchiostro. È inoltre consigliabile lasciar



correttamente riavvolgere a mano — o meglio a dito! — il nuovo, con tutti gli inconvenienti dovuti all'eccessiva quantità di inchiostro presente sul nastro.

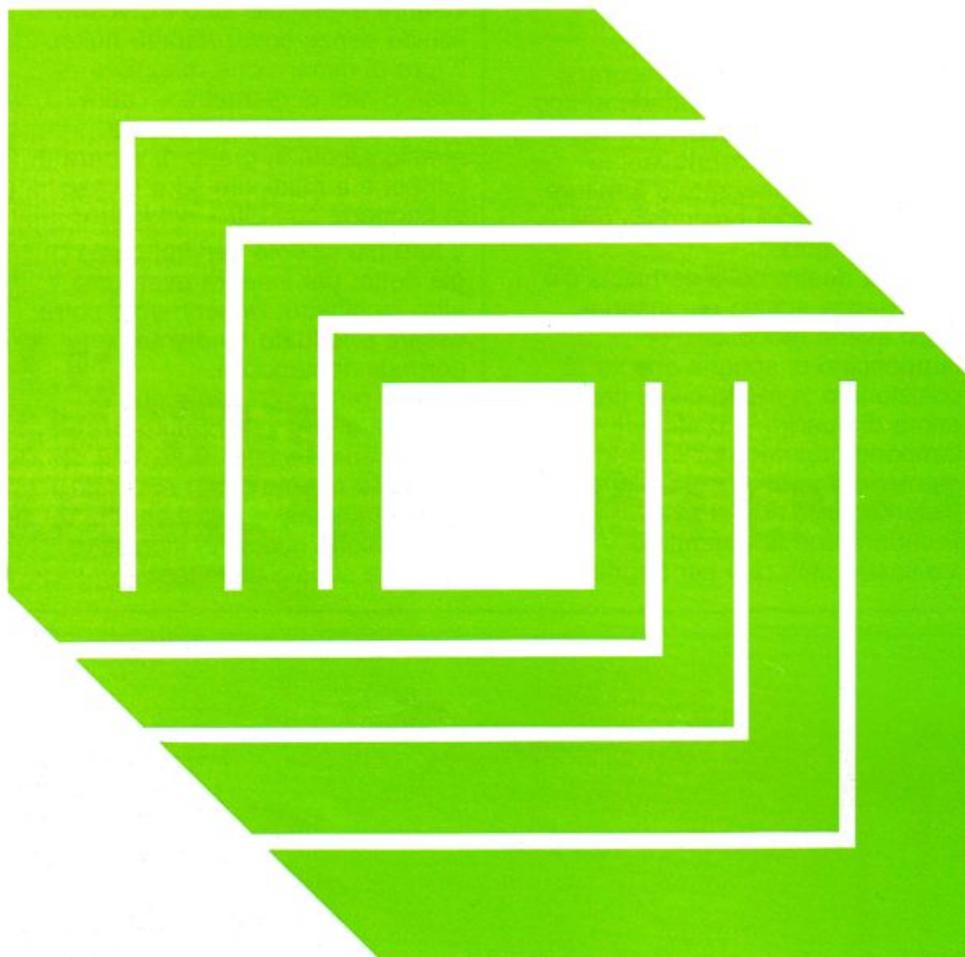
Ed è proprio l'inchiostro il problema principale: i nastri infatti ancor prima di essere consumati

spugnetta: questa deve sì essere imbevuta ma non si deve, nella cartuccia, formare un «laghetto»! Sì, direte voi, l'idea è buona, ma quando anche l'inchiostro della spugnetta è esaurito? Nell'aprire più volte la cartuccia i famosi dentini e perni si potrebbero

cadere una goccia di inchiostro sulla spugnetta, dal foro di controllo, ogni volta che la stampante viene usata. In questo modo il nastro della vostra stampante durerà a lungo ed otterrete sempre una stampa perfettamente leggibile.



TRUCCHI E SCOPERTE



ZX - SPECTRUM

È possibile, molti Sinclairiani si saranno chiesti, registrare il contenuto della memoria dello Spectrum?

Probabilmente quanti vi avranno provato saranno giunti alla conclusione che questo non sia possibile in quanto si saranno resi conto che all'atto pratico non si possono caricare più di 65279 bytes per registrazione.

Come fare dunque per accedere e registrare il contenuto di memoria del computer? Semplice: digitare quanto segue e... verificare per credere!

SAVE «ALL ROM» CODE 0,16384

QUANDO il computer «attende», sia in fase di LIST che durante l'esecuzione di un programma, che venga premuto un tasto, se si premono contemporaneamente i due tasti SHIFT riappare l'ultimo comando eseguito o in esecuzione da parte del Sistema Operativo, ed il cursore.

SAPEVATE che è possibile registrare titoli di programmi più lunghi di 10 caratteri? Perché non appaia il fatidico «INVALID FILE NAME» basterà inserire i codici delle KEYWORDS, ad esempio

SAVE «SAVE NEW GAME LINE 1»

Per fare ciò, digitate, dopo aver aperto le virgolette, THEN seguito dalla KEYWORD scelta e quindi cancellatelo con i tasti cursore e DELETE... un po' di fantasia in questi titoli ci vuole!

CON questa semplicissima procedura disporrete di un cursore «scattante» come una Ferrari!

Digitate la linea

```
1 POKE 23562,5 : POKE 23561,35 :  
STOP
```

e, poi, in modo diretto

```
POKE 23562,1 : POKE 23561,1
```

SE siete in possesso di uno Spectrum Issue 2 provate a digitare POKE 23738,0

Riuscirete ad effettuare eventuali correzioni, normalmente possibili solo con l'EDIT, portandovi con il cursore direttamente fra le linee del listato.

Per tornare alla normalità digitare POKE 23738,75

COMMODORE

OLTRE al comando standard per leggere la Directory del Drive 1541 vi segnaliamo la possibilità di leggere il solo nome del dischetto e la quantità di blocchi liberi, usando l'istruzione

```
LOAD «$#»,8
```

Non fate caso alla segnalazione «?FILE NOT FOUND ERROR» e digitate LIST.

PER ottenere l'autorun caricando un programma da disco digitate

```
LOAD «nome progr.», 8: e premete  
SHIFT+RUN/STOP
```

ANCHE sul CBM 64 è possibile ottenere un cursore «a Mach 2», digitando

```
POKE 56325,10
```

Ciò può essere particolarmente utile per rallentare la routine di SCROLL dell'istruzione LIST anche mediante il tasto SHIFT.

UN piccolo trucco per chi non ha problemi di carta con la propria 801...

Potete ottenere la doppia spaziatura utilizzando un numero di File maggiore di 127, ad esempio

```
OPEN 129,4 : CMD 129 : LIST  
PRINT # 129 : CLOSE 129
```

INPUT O GET...? Questo è il dilemma!

```
Digitando 10 OPEN 1,0  
20 PRINT «PROVA A  
PREMERE UN  
TASTO» ;:  
25 INPUT # 1, A$  
30 CLOSE 1
```

otterrete un'istruzione INPUT in cui non appare il punto interrogativo!

UN TASTO... UN'ISTRUZIONE

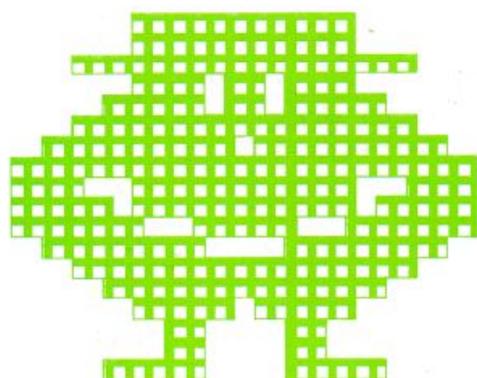
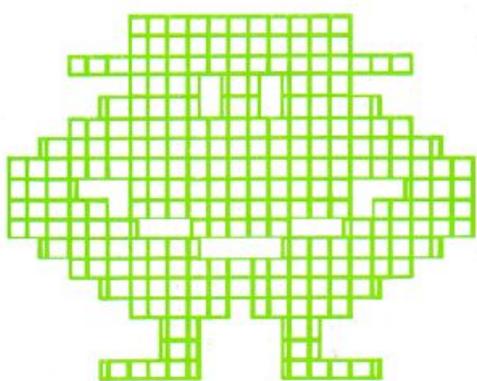
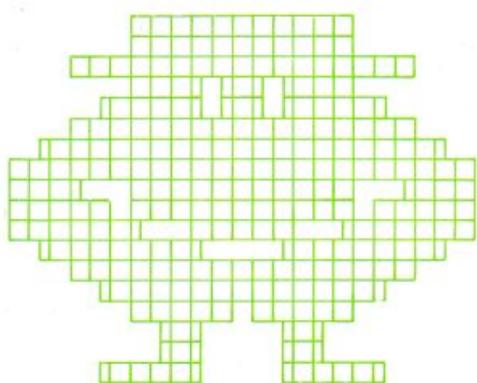
A molti di voi sarà capitato, mentre

digitavate i vostri programmi, di tentare di visualizzare simboli grafici dopo un'istruzione REM. Dopo aver listato il programma avrete notato come tutti i simboli grafici usati si siano trasformati in altrettante istruzioni BASIC contemplate dal sistema operativo del Commodore. Vi siete chiesti se sia possibile usare questa particolarità nella stesura dei programmi? La cosa è possibilissima ed infatti diamo di seguito l'elenco delle «KEYWORDS» ottenibili digitando l'istruzione REM seguita dai simboli grafici più opportuni. Fate attenzione, salvo diverse indicazioni (CBM+) i tasti si intendono premuti insieme allo SHIFT e quindi in modo grafico. Una volta finita la stesura del programma, non rimarrà altro da fare che cancellare la REM alla testa di ogni linea.

ABS	CBM+L	NEXT	N
AND	CBM+P	NOT	CBM+£
ASC	F	ON	—
ATN	A	OR	CBM+A
CHR\$	G	PEEK	B
COS	CBM+V	POS	CBM+O
DATA	O	READ	S
DIM	R	REM	+
EXP	CBM+X	RESTORE	X
FN	CBM+G	RETURN	Z
FOR	M	RIGHT\$	I
FRE	CBM+U	RND	CBM+F
GET	CBM+K	RUN	V
GO	K	SGN	CBM+H
GOSUB	Y	SIN	CBM+B
GOTO	U	SPC	CMB+(+)
IF	W	SQR	£
INPUT	Q	STEP	£
INPUT#	P	STOP	CBM+(-)
INT	CBM+J	STR\$	D
LEFT\$	H	TAB	CBM+T
LEN	C	TAN	*
LOAD	CBM+(*)	THEN	CBM+M
LOG	CBM+C	TO	CBM+£
MID\$	J	USR	CBM+Y
NEW	CBM+I	VAL	E



DIDATTICA



Apriamo un piccolo spazio dedicato ai nostri lettori più inesperti spiegando loro quali sono alcuni degli errori più comuni cui vanno incontro nella stesura dei programmi. Sono stati divisi per categorie e per la frequenza riscontrata nel «feedback» delle vostre lettere e telefonate.

COMMODORE

LINEE più lunghe di 80/88 caratteri. Devono essere digitate usando le abbreviazioni delle parole chiave... La spiegazione è contenuta a pag. 130 e 131 del manuale italiano per il Commodore 64 ed a pag. 132 e 133 per il VIC 20, sempre per il manuale italiano.

NOMI proibiti di variabili. Sono tutti quelli che ricordano o che ripetono le KEYWORD del sistema... Ad esempio non potrete far uso di una variabile SYS=9 oppure di una variabile CMD=197. In entrambi i casi il computer interpreterà il nome della variabile come un comando basic contemplato nel SET delle istruzioni standard.

CICLI di For-Next.
Uno strano caso si è verificato più volte nella stesura dei programmi per il VIC ed il 64. Nei cicli di FOR-NEXT «nidificati». Per intenderci quelli del tipo:
10 FOR X=1 TO 10
20 FOR Y=20 TO 30
30 FOR Z=25 TO 35
40 POKE 7680+X,Y:PRINT CHR\$(Z)
50 NEXT:NEXT:NEXT

Accade che se nella linea 50 non si specifica la variabile a cui il NEXT fa riferimento, compare sullo schermo la segnalazione «?NEXT WITHOUT FOR ERROR». Probabilmente ciò è dovuto al fatto che alcuni computers importati in Italia hanno standards leggermente diversi da quelli più adatti alle norme di regolamentazione del settore, vigenti nel nostro Paese. Il rimedio è fortunatamente piuttosto semplice.

Basta modificare la linea 50 nel modo seguente:
50 NEXT Z,Y,X

SEGA SC-3000 e TI 99/4A

Sono stati raggruppati insieme perché spesso i loro errori (a dire il vero gli errori commessi da chi usa questi sistemi) si somigliano.

SCAMBIO di lettere.
Tra i più comuni errori, lo scambio della lettera O maiuscola con lo zero e lo scambio del numero 8 con la lettera B specialmente nelle istruzioni PATTERN e CALL CHAR. CON riferimento al TEXAS ci è capitato di trovarci alle prese con uno strano problema lamentato da un nostro lettore. Digitando un programma pubblicato sulla rivista, sul suo computer accadevano cose stranissime. Il primo dei fenomeni più strampalati era l'accettazione o meno dei due punti (:) dopo l'istruzione PRINT, solo se in numero dispari o viceversa. Con grande curiosità si è chiesto di poter constatare personalmente questi strani fenomeni ed alla fine l'arcano mistero è stato svelato. Il lettore digitava un programma destinato al TI BASIC selezionando il TI EXTENDED BASIC del modulo inserito nello SLOT del suo computer. Non finisce qui, una volta finito di digitare il listato, lo salvava su nastro e poi lo caricava dopo aver rilesionato il TI BASIC. I risultati erano chiaramente disastrosi.

Chiedendo il listato, molte istruzioni erano mostrate come il contenuto delle locazioni di memoria in quel momento, perciò venivano visualizzati strani «segni» di colore diverso che poi mandando in esecuzione il programma (RUN) il computer non riusciva ad interpretare. Fate attenzione quindi, salvo diverse indicazioni i programmi per il TI 99 vanno digitati selezionando solo il TI BASIC.



HI-RES PLOT

VIC 20

Sono molte le richieste pervenute in redazione per soddisfare una giusta curiosità dei lettori utilizzatori del famoso VIC 20 che tanto successo ha riscosso in passato. Nel numero 3 (Maggio-Giugno '84), accennammo alla simulazione in BASIC di una seconda pagina grafica in alta risoluzione. Pressati dalle molte richieste abbiamo pensato bene di rispolverare l'argomento, proponendovi un semplice programmino di sole 27 linee scritto interamente in BASIC. Tale programma permette di creare, nella configurazione base del VIC (3,5K RAM), una pagina grafica di 127 * 127 pixels e «plottare» su di essa grafici di funzioni matematiche, disegni ed altro. Il procedimento si basa sulla possibilità di visualizzare sullo schermo il contenuto di una zona della memoria RAM. Agendo sui singoli bit di questa determinata zona si possono facilmente ricostruire determinate curve geometriche o comunque disegni. Se il procedimento vi sembra di scarsa comprensione, vi consiglio di consultare un precedente articolo apparso sul numero 6 (Nov.-Dic. '84) intitolato OUTLINE. Facendo riferimento a questo articolo, se si sostituisce al blocco di DATA un valore numerico calcolato dal VIC relativamente ad una determinata funzione matematica, sullo schermo vedremo il grafico descritto dai valori numerici calcolati nell'esecuzione. Approfondendo il discorso, per OUTLINE ogni istruzione DATA conteneva 8 valori riconducibili ai singoli byte che componevano il carattere. Per intenderci, ogni 8 bit si ottiene un byte e nel nostro caso con 8 bytes si identificava un carattere. Per la gestione grafica in HGR (leggi alta risoluzione) bisogna leggere ogni bit singolarmente ed identificare a quale degli 8 bytes che compongono ogni casella dello schermo appartiene. Se guardate il listato noterete che tale compito è affidato alla routine contenuta nelle linee 5 e 6; la prima identifica il contenuto del bit corrispondente all'asse X e la seconda il byte di appartenenza nell'asse Y. Insieme concorrono ad identificare la cella di memoria da visualizzare con una opportuna istruzione POKE. Nel nostro caso la variabile BY è l'argomento dell'istruzione POKE. Le linee da 13 a 19 contengono i limiti della finestra, l'asse X e la funzione da visualizzare.

STRUTTURA DEL PROGRAMMA

- 0- Riserva lo spazio in memoria da usare come pagina grafica; annulla tutte le variabili; salto incondizionato alla linea 10.
- 1- Pulisce la pagina grafica da tutto quello che era precedentemente contenuto.

- 2- Altera i puntatori di centratura schermo, lettura carattere, numero colonne, numero righe, colore fondo e colore bordo schermo.
- 3- Allocazione nelle cellule di memoria comprese tra 5632 e 5887 dei valori crescenti compresi tra 0 e 255; corrispondono alle possibili condizioni di visualizzazione all'interno di ogni byte.
- 4- Prepara il fondo per la visualizzazione dei PIXEL nel colore CO.
- 5-7 Identificazione delle colonne e delle righe, visualizzazione sullo schermo.
- 8-9 Ripristino delle normali condizioni dello schermo.
- 10-11 Alterazione dei puntatori di gestione dello schermo.
- 12- Ciclo principale.
- 13-19 Definizione limiti della finestra e curva da visualizzare.
- 20- Arresto dell'esecuzione del programma fino alla pressione di un tasto.
- 21- Cambio del colore di visualizzazione.
- 22- Arresto come alla linea 20.
- 23- Salto incondizionato alla routine di pulizia dello schermo.
- 24- Come alle linee 20 e 22.
- 25- Salto alla routine di ripristino delle normali condizioni di gestione dello schermo.
- 26- Fine programma.

LOCAZIONI DI MEMORIA USATE NEL PROGRAMMA

- 52 Puntatore di memorizzazione delle stringhe.
- 56 Puntatore di limite massimo della memoria RAM disponibile.
- 36864 Centratore orizzontale dello schermo.
- 36865 Centratore verticale dello schermo.
- 36866 Numero di colonne e matrice di indirizzi video.
- 36867 Numero di righe e formato caratteri (8 * 8 oppure 16 * 8).
- 36869 Puntatore di inizio mappa caratteri e indirizzi di memoria video.
- 36879 Colore bordo e fondo schermo.
- 648 Puntatore pagina di memoria dello schermo.
- 217-240 Puntatore tabella linee di schermo (158=termine — 30=continuazione).
- 241 Indirizzo fittizio di schermo.
- 242 Marcatore linea di schermo.
- 243-244 Puntatore mappa colori dello schermo.
- 245-246 Puntatore tastiera.
- 247-250 Puntatori di ricezione e trasmissione dati periferica RS-232.



PER OTTENERE GRAFICI DIVERSI DA QUELLO PRESENTE NEL LISTATO SOSTITUIRE LA LINEA 19. PER ESEMPIO, PROVATE COSÌ...:

```
19 FORX=0T0127:Y=INT(120*SIN(X/40)):GOSUB 5:NEXT X
```

```
0 POKE52,22:POKE56,22:CLR:GOTO10
1 FORI=6144T08191:POKEI,0:NEXT:RETURN
2 POKE36864,12:POKE36865,38:POKE36866,14
4:POKE36867,32:POKE36869,222:POKE36879,1
2
3 FORI=0T0255:POKE5632+I,I:NEXT:RETURN
4 FORI=0T0255:POKE38400+I,C0:NEXT:RETURN
```

```
5 CH=INT(Y/8)*16+INT(X/8)
6 RO=(Y/8-INT(Y/8))*8:BY=6144+8*CH+RO
-7 BI=7-(X-INT(X/8))*8:POKEBY,PEEK(BY)OR(
2+BI):RETURN
8 POKE36864,12:POKE36865,38:POKE36866,15
0:POKE36867,46:POKE36869,208:POKE36879,2
7
9 PRINTCHR$(147):RETURN
10 POKE36869,208:POKE648,22:FORJ=217T022
8:POKEJ,150:NEXT
```

```
11 FORJ=229T0250:POKEJ,151:NEXT
12 GOSUB2:C0=1:GOSUB4:GOSUB1
13 C0=3:GOSUB4:FORX=0T0127:Y=0:GOSUB5:Y=
64:GOSUB5:Y=127:GOSUB5:NEXTX
14 C0=3:GOSUB4:FORY=0T0127:X=0:GOSUB5:X=
127:GOSUB5:NEXTY
19 C0=1:GOSUB4:FORX=0T0127:Y=INT(64+63*
SIN(X/10)):GOSUB5:NEXTX:A#=""
20 GETA#:IFA#=""THEN20
21 C0=5:GOSUB4:A#=""
22 GETA#:IFA#=""THEN22
23 GOSUB1:A#=""
24 GETA#:IFA#=""THEN24
25 GOSUB8
26 END
```

READY.

Desideriamo dare spazio in questa rubrica al programma ed alle note introduttive inviateci dal lettore Giulio D'Urso di Firenze, che ha risolto lo stesso problema in modo diverso. Volentieri pubblichiamo il listato. Il programma «GRAPHIC» offre la possibilità di ottenere sul video il grafico di non più di tre funzioni contemporaneamente, ma ciascuna di un colore diverso! Difatti viene sfruttato il MULTICOLOR MODE in HI-RES, quindi da una pagina grafica di 96*128 pixels se ne ottiene in definitiva una di 96*64. Una volta lanciato il programma è necessario inserire fino ad un massimo di tre linee (n. 18, 19 e 20) per descrivere le funzioni che si desiderano visualizzare; dopodiché il programma richiede il numero totale di funzioni, gli estremi di rappresentazione, la scala dell'asse delle ordinate (l'overflow è protetto), la posizione desiderata per l'asse X (quella dell'asse Y è

automatica), infine se si desidera visualizzare anche gli assi oppure no. A scopo dimostrativo sono state lasciate inserite tre funzioni la cui visualizzazione migliore si ottiene rispondendo rispettivamente con la seguente sequenza:

```
GOTO 8;3;-3.14,3.14;45;0;SI;
```

Facile immaginare le indefinite applicazioni scientifiche e non (ad esempio «indagatrici») di questo programma.

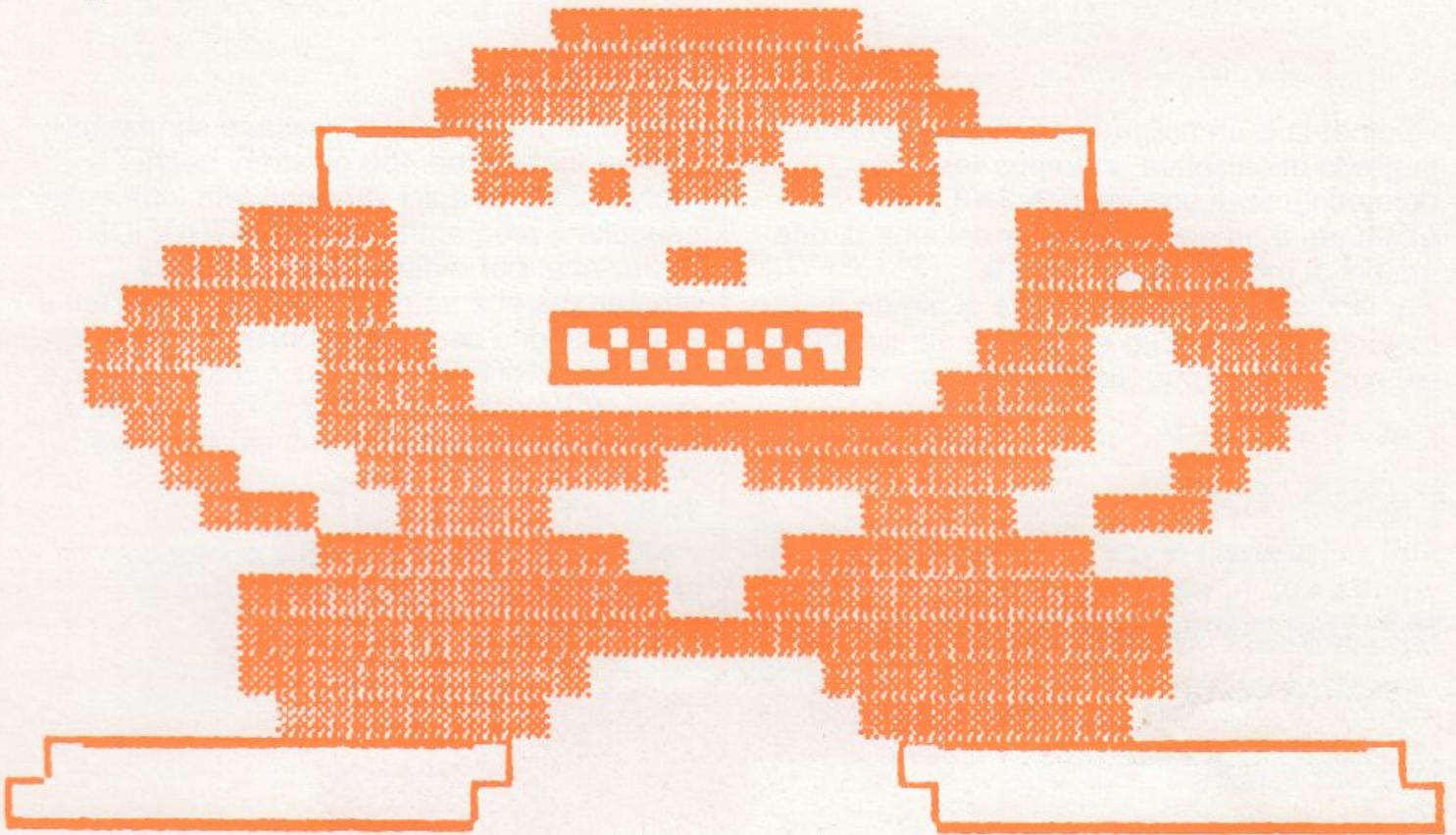
```
1 POKE36879,29:POKE52,24:POKE56,24:POKE3
6878,0
2 PRINT"GRAFIC >":PRINT"
PROGRAMMA GRAFICO"
3 PRINT"BY GIULIO D'URSO"
4 FORI=6144T07679:POKEI,0:NEXT:DIMY(3):F
ORI=1T03:Y(I)=0:NEXT
5 PRINT"FR A 18 E 20":PRINT"INSERI
SCI LE FUNZIONIY(X) ES: 19 Y(1)=F1(X)"
```

```
6 PRINT"PER ELIMINARE LE FUNZPRESENTI
CANCELLARE LE LINEE 18-20"
7 PRINT"INFINE DIGITA < GOTO8 >":END
8 INPUT"QUANTE FUNZIONI(0-3)":N
9 INPUT"ESTREMI A,B (ACB)",X1,X2
10 INPUT"SCALA ASSE Y":FY
11 INPUT"ASSE X (-47/+47)":OX
12 FX=(X2-X1)/62:X2=-X1/FX:OY=INT(X2):BI
=OY-INT(OY/4)*4:BI=192/(4+BI)
13 INPUT"ASSI X E Y (SI/NO)":A#
14 PRINT"WORKING":IFA#
="NO"THEN17
```

```
15 FORBY=6192-0XT07632-0XSTEP96:POKEBY,2
55:NEXT
16 FORBY=6144+INT(OY/4)*96T06239+INT(OY/
4)*96:POKEBY,PEEK(BY)ORBI:NEXT
17 FORT=0T062:X=X1+FX*T
18 Y(1)=SIN(X)
19 Y(2)=COS(X)
20 Y(3)=X/3
21 FORI=1TON:Y=48-FY*Y(I)-OX:IFY<0ORY>95
THEN24
22 CH=INT(T/4)*12+INT(Y/8):RO=Y-INT(Y/8)
*8:BY=6144+8*CH+RO
23 BI=I*2+16/4+(T-INT(T/4)*4):POKEBY,PEEK
(BY)ORBI
24 NEXT:NEXT
25 PRINT"POKE36869,254:FORL=0T015:FOR
M=0T011
26 POKE38513+M*22+L+C,12:POKE7793+M*22+L
+C,L*12+M:NEXT:NEXT
200 GOTO200
```

READY.

Keyboard codes



Un programma didattico per poter comprendere quali codici vengono inseriti nel buffer di tastiera alla locazione 197 del VIC. Il funzionamento è estremamente semplice e può essere adattato anche ad altre locazioni di memoria di cui si vuole conoscere il contenuto. Nel VIC esiste la particolarità di poter usufruire oltre che dell'istruzione GET...\$ per riconoscere il tasto premuto anche di una analoga PEEK(197) che, essendo codificata in decimale, può essere usata in tutti quei casi in cui è necessario

«mascherare» alcuni tasti che attivano determinate funzioni. Il programma permette di conoscere tutti i codici dei tasti che seguono tale regola. Inutile dire che se consultate il N. 2-85 alla rubrica HANDBOOK potrete dopo aver disattivato il tasto RUN-STOP, usarlo per funzioni speciali o segrete sfruttando un'istruzione del tipo:

```
IF PEEK (197) = ... THEN ...
```

Al posto dei puntini dovreste sostituire il codice del tasto RUN-STOP e la linea di programma in cui inizierà la routine speciale.

```
1 REM # KEYBOARD CODES
2 REM #
3 REM #   L I S T
4 REM #
5 REM # (C) 1985 MAX.
6 REM
10 PRINT "C":POKE36879,25
12 PRINT "COD. TASTIERA=PEEK(197)":PRINT
14 FOR T=1 TO 2500:NEXT T:PRINT "C"
15 PRINT "64 = NESSUN TASTO -NON ACCETTATO-":PRINT
20 PRINT:PRINT "PREMERE IL TASTO PER CONOSCERNE IL CODICE "
25 GETX$:IFX$="" THEN 25
27 PRINT:PRINT:P=PEEK(197)
28 IFF=32 THEN PRINT "TASTO NON ACCETTATO-":PRINT
```

```
29 IFF=31 OR P=23 THEN PRINT "TASTO NON ACCETTATO-":P:GOTO 40
30 PRINT "TASTO NON ACCETTATO-":P:PRINT "X$ "
31 IF PEEK(203)=64 THEN PRINT "TASTO NON ACCETTATO-":PRINT " "
32 GETX$:IFX$="" THEN 32
33 PRINT:PRINT:P=PEEK(197)
34 IFF=32 THEN PRINT "TASTO NON ACCETTATO-":P:PRINT " "
35 IFF=31 OR P=23 THEN PRINT "TASTO NON ACCETTATO-":P:PRINT
40 PRINT:PRINT
41 PRINT "TASTO NON ACCETTATO-":PRINT " "
42 GOTO 30
READY.
```



Geometria 1

Geometria è un programma molto elementare in grado di calcolare, in tempo reale, il determinante di una matrice 3*3 e di effettuare il prodotto righe per colonne di due matrici di ordine massimo 6*6.

Per il calcolo del determinante si sfrutta il metodo di Sarrus che è brevemente illustrato nel suo svolgimento, qui di seguito:

$$\begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} = F(n)$$

$$F(n) = [(a * e * i) + (d * c * h) + (b * f * g)] - [(c * e * g) + (h * f * a) + (b * d * i)]$$

Per il calcolo del prodotto tra matrici è valido il

quanto strani ed astrusi possano sembrare (per es. nella linea 400 gli strani segni racchiusi tra gli apici verranno letti, una volta eseguito il programma, come ISTRUZIONI). Attenzione: per digitare quella parte di programma che va dalla linea 2530 alla linea 2580, bisogna usare le abbreviazioni per le istruzioni PRINT e TAB. Sul manuale d'uso sono indicate a pagina 130 e 131 (versione italiana). In ogni caso le abbreviazioni sono: PRINT = ?

TAB(= T [SHIFT] A

Nell'abbreviazione per TAB è contenuta anche la parentesi, ricordarsi quindi di richiuderla.

seguito procedimento riportato nell'esempio:

$$\begin{vmatrix} a & b & | & | & | & x & x1 & x2 \\ | & | & | & g & h & i & | & | \\ c & d & | * & | & | = & | y & y1 & y2 \\ | & | & | & | & m & n & | & | \\ e & f & | & | & | & z & z1 & z2 \end{vmatrix}$$

$$x = (a * g) + (b * i)$$

$$y = (c * g) + (d * i)$$

$$z = (e * g) + (f * i)$$

$$x1 = (a * h) + (b * m)$$

$$y1 = (c * h) + (d * m)$$

$$z1 = (e * h) + (f * m)$$

$$x2 = (a * i) + (b * n)$$

$$y2 = (c * i) + (d * n)$$

$$z2 = (e * i) + (f * n)$$

Le istruzioni, nell'esecuzione del programma, compaiono scritte in minuscolo, quindi non preoccupatevi per gli strani simboli grafici che appaiono nel listato, basta fare attenzione a digitare esattamente gli stessi segni, per



Grafica 3-D

Una delle caratteristiche principali del Sega è, senza dubbio, la notevole quantità di istruzioni grafiche previste nel SET del sistema. Senza dubbio l'SC-3000 non ha nulla da invidiare a macchine ben più costose e sofisticate e permette una discreta programmazione anche in un campo decisamente «difficile» quale la grafica. Oltre alle solite possibilità, quali il tracciamento di singoli punti e di segmenti in qualsiasi modo orientati, la funzione che merita una particolare attenzione è quella che permette di tracciare cerchi e archi di circonferenza.

La funzione CIRCLE si avvale di ben 7 parametri e consente di tracciare anche ellissi variamente posizionate. L'ultimo parametro consente anche di disegnare dischi o comunque figure il cui interno sia colorato. Gli esempi che appaiono su queste pagine sono una chiara dimostrazione delle enormi possibilità a disposizione.

Osserviamoli con maggiore attenzione.
FIG. 1

Come si può vedere è possibile con poche linee di programma avere a disposizione una figura che ricordi un merletto, senza che lo stesso disegno si ripeta più di una volta. Il programma, un DEMO per il Digitek MPF II, è molto semplice.

Fornendo valori RANDOM ai parametri, si ha a disposizione una gamma infinita di figure.
FIG. 2

Questo secondo esempio è una particolare utilizzazione dell'istruzione LINE. La particolarità consiste nel fatto che i valori identificanti gli estremi di ogni segmento che compone la figura, sono ottenuti mediante alcuni calcoli ricavati dall'uso delle funzioni SENO e COSENO.

FIG. 3

Programma molto semplice che permette di valutare la risoluzione del monitor o del

televisore collegato al sistema. Volutamente è stato inserito un errore di allineamento sugli angoli in alto a destra ed in basso a sinistra. Se non si riesce a vederlo è giunta l'ora di regolare meglio il monitor o televisore ed, eventualmente, di sostituirlo.

FIG. 4

Questo esempio fornisce un quadro d'insieme molto generale sul tipo di figure circolari che è possibile disegnare alterando opportunamente i parametri.

FIG. 5

Un esempio di notevole effetto, peccato che la fotografia non riesca a rendere perfettamente la spettacolarità della figura. La sola funzione grafica usata è CIRCLE. Per ottenere l'andamento sinusoidale della FIGURA, si è variato il punto di centro dei cerchi secondo un ciclo FOR-NEXT per l'asse delle X, e mediante un'istruzione del tipo $Y=SEN(X)$ per l'asse Y.

FIG. 6

Anche in questo esempio si è fatto largo uso dell'istruzione CIRCLE.

Il programma, ribattezzato ampollosamente SATURN DRAWER, mostra però quella che nel gergo dei disegnatori viene chiamata aberrazione o distorsione visiva. Si noterà certamente che gli archi di circonferenza con cui sono rappresentati gli anelli del pianeta, cessano bruscamente prima che vengano in contatto con la superficie dello stesso. Ciò deriva dal fatto che non si sono variati i parametri riguardanti il punto d'inizio ed il punto di fine dell'arco di circonferenza per ogni diametro usato.

Invitiamo tutti i volenterosi a voler trovare un metodo di risoluzione del problema ed eventualmene a volercelo comunicare.

L'ultima foto mostra cosa si può riuscire a disegnare sul SEGA se solo si ha una buona dose di pazienza e volontà.

Un discorsetto a parte merita l'ultimo



programmino proposto. Si tratta di un esempio di grafica a tre dimensioni. In questo, come del resto in tutti i casi simili, vengono fuori alcuni problemi tipici della fascia di computer di cui anche il SEGA SC-3000 fa parte. Anche in questo caso, così come per macchine più illustri, il tempo per tracciare

HI-RES 1

```
10 CLS:SCREEN 2,2:CLS
20 F=INT(RND(1)*13)+2:K=1.2
30 M=RND(1)*30+3
40 N=RND(1)*M+1
50 COLOR0,1,(0,0)-(254,191),1
60 LX=140:LY=95
70 FOR A=0 TO 70 STEP RND(1)+.01
80 R=SIN((M/N)*A)*90
90 X=140+K*R*SIN(A):Y=95+R*COS(A)
100 LINE (LX,LY)-(X,Y),F:LX=X:LY=Y
110 NEXT A
120 X$=INKEY$:BEEP:IF X$="" THEN 120
130 GOTO 10
```

HI-RES 2

```
10 SCREEN 2,2:COLOR2,1,(0,0)-(254,191),1
20 CLS:P=3.14159:Q=180
30 FOR A=1 TO 1180 STEP 30
40 X1=200-(9*COS(P*A/Q)-2.5)/2 -50
50 Y1=(1*SIN(P*A/Q)+A/2)/10
60 X2=150-(9*COS(P*A/Q))/1.5-130
70 Y2=(2*SIN(P*A/Q)+A/2)/1.5+64
80 O=INT(RND(1)*13)+2
90 LINE (Y2*.5,X1)-(Y1+99,X2),O
100 NEXT A
110 GOTO 110
```

HI-RES 3

```
10 CLS:SCREEN 2,2:COLOR1,1,(0,0)-(255,191),1
20 CLS:FOR I=1 TO 127 STEP 3
30 LINE (64+I,I)-(191-I,I),5:NEXT I
40 FOR I=1 TO 127 STEP 3
50 LINE (I+64,I)-(I+64,127-I),5:NEXT I
60 GOTO 60
```

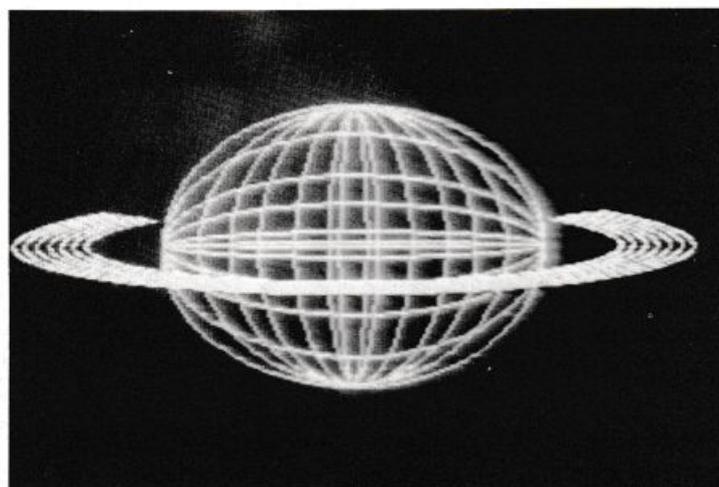
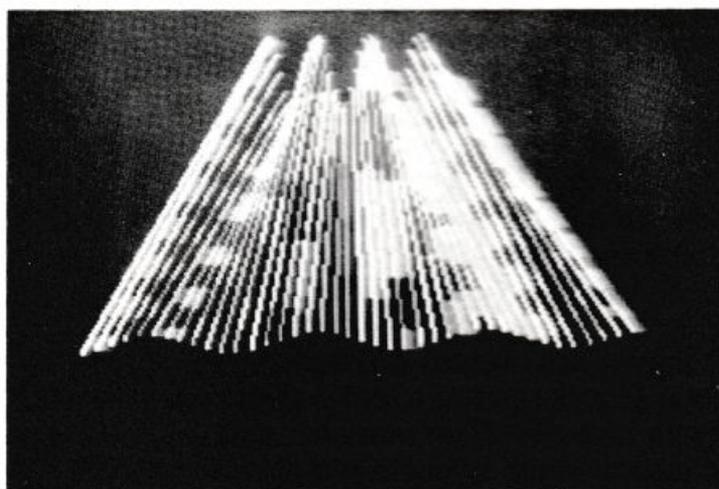
tutta la figura, raggiunge «traguardi» piuttosto elevati. Ciò è dovuto al fatto che la routine usata, traccia punto per punto tutta la figura e deve calcolare ed evitare di disegnare tutti i punti che si trovano — dietro — quanto già disegnato.

HI-RES 4

```
10 CLS:SCREEN 2,2:CLS:COLOR1,15,(0,0)-(255,191),15:E=191
20 FOR Q=1 TO 64
30 X=(RND(1)*205)+25:Y=(RND(1)*140)+25
:K=INT(RND(1)*14)+1
40 CIRCLE(X,Y),RND(1)*22,K,(RND(1)*2),0,1
50 NEXT Q
60 GOTO 10
```

HI-RES 5

```
10 CLS:SCREEN 2,2:CLS:COLOR1,1,(0,0)-(255,191),1:X=30
```



```

20 FOR I=0 TO 6.5 STEP .1
30 X=X+3 :Y=(SIN(I)*50)+90
40 CIRCLE(X,Y),20,5,1,0,1
50 NEXT I
60 GOTO 60

```

HI-RES 6

```

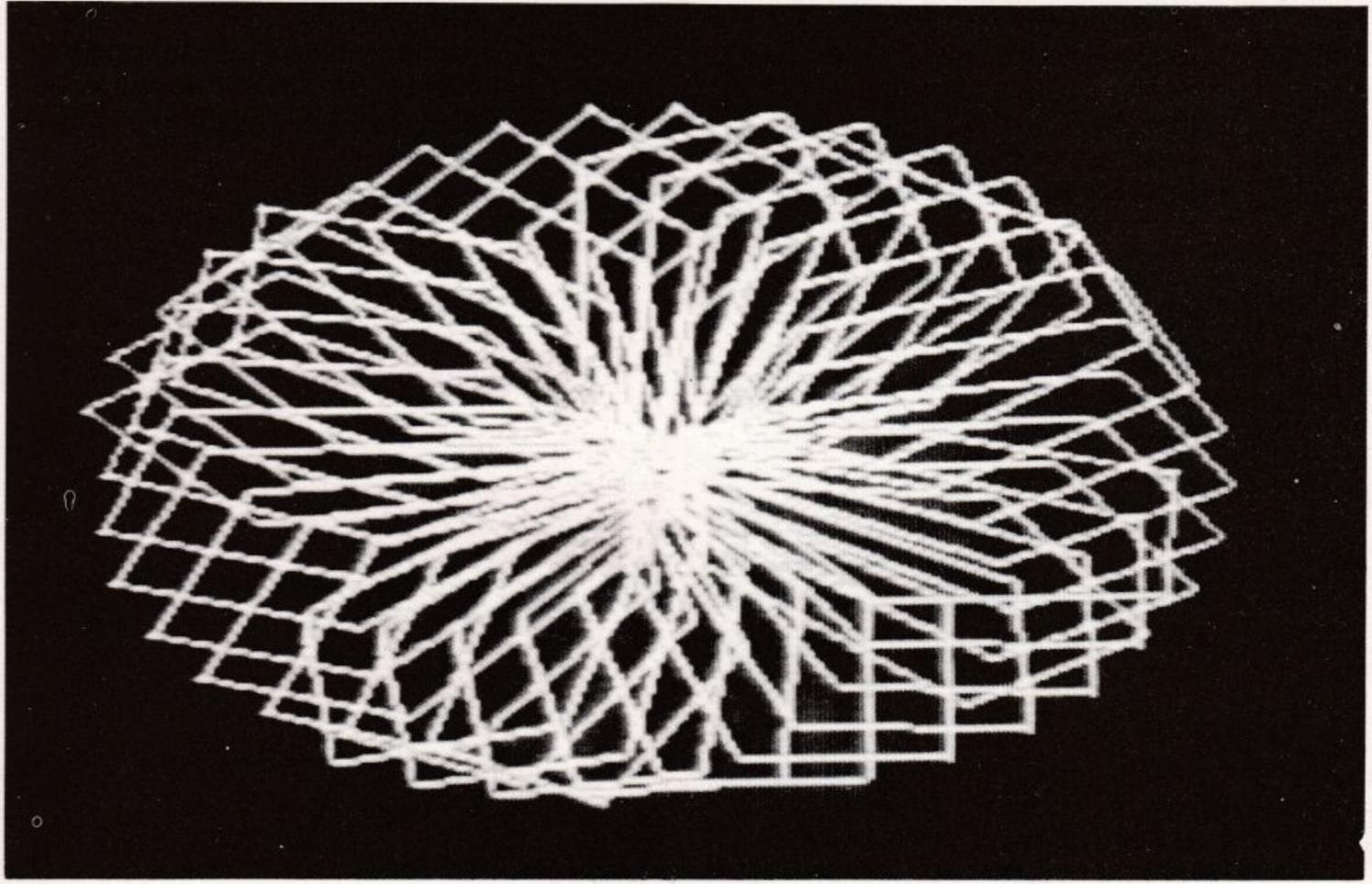
10 CLS
20 SCREEN 2,2:COLOR1,1,(0,0)-(255,191)
,1
30 CLS:CIRCLE(127,95),64,5,1,0,1
40 FOR I=1 TO 6:READ J
50 CIRCLE(127,95),64/J,5,J,0,1:NEXT I
60 DATA 1.2,1.5,2,3.2,7,12.8
70 FOR I=1 TO 5:READ K
80 CIRCLE(127,95),64,5,K,0,1:NEXT I
90 DATA .02,.1,.3,.5,.8,1
100 FOR I=0 TO 24 STEP 4
110 CIRCLE(127,95),90+I,8,.2,.88,.62:N
EXT I
120 GOTO 120

```

```

1 REM #####
2 REM #
3 REM # G R A F I C A #
4 REM #
5 REM # TRIDIMENSIONALE #
6 REM #
7 REM # SEGA SC-3000 #
8 REM #
9 REM #####
10 CLS:SCREEN 2,2:CLS:COLOR1,1,(0,0)-(
254,191),1
20 DIMA(254):T=15
30 FOR YY=-9.5 TO 9.5 STEP .7
40 FOR XX=-9.5 TO 9.5 STEP .08
50 Q=(SQR((1+XX*XX/16+YY*YY/9)*((1+XX*
XX/16+YY*YY/9)+1/2)))
60 Y=INT(((SIN(Q)/Q)*75+T*1.3)
70 X=INT((XX+9.5)*9.5)+T*.92-14
80 IF X>0 AND X<254 THEN IF Y>A(X) THE
N A(X)=Y:Y=191-Y:GOSUB 110
90 NEXT XX:T=T+2:NEXT YY
100 GOTO 100
110 IF X>254 OR X<0 OR Y>191 OR Y<0 TH
EN RETURN
120 K=INT(RND(1)*13)+2
130 PSET (X,Y),K
140 RETURN

```



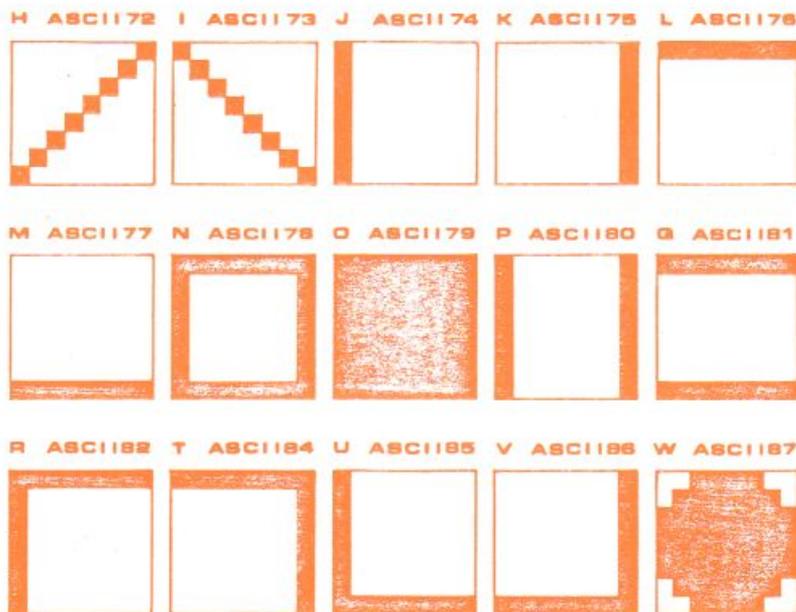


```
107:POKEM+2008,166:POKEM+1,106:POKEM+204
9,166:GOSUB5030
350 POKEM+2047,85:POKEM+2008,85:POKEM+20
49,85:NEXTJ
360 FORJ=-2T02:POKEM+J+2048,85:NEXT
370 Z1=Z:Z=9:S=1:FORR=1T09:E=E+40:ONDGOT
0420,430
380 FORQ=0T04:IFE=A+QTHENL=A+1:POKEE+204
8,17:POKEE+2047,17:POKEE+2046,17:I=0:Z=Z
1:S=0:GOSUB3350:GOTO240
390 NEXTQ
400 POKEE+2048,17:POKEE+2047,17:POKEE+20
46,17:NEXTR:I=0:S=0:Z=Z1:GOSUB3350
410 GOTO440
420 POKEE,F1:POKEE+2048,241:POKEE-1,F2:P
OKEE+2047,241:POKEE-2,F3:POKEE+2046,241:
GOSUB3200:GOTO380
430 POKEE,F3:POKEE+2048,241:POKEE-1,F2:P
OKEE+2047,241:POKEE-2,F1:POKEE+2046,241:
GOSUB3200:GOTO380
440 D=INT(RND(1)*2+1):FORK=0T031:POKE557
40+K,85:NEXT:ONDGOTO210,220
450 DATA C5,E3,G3,A4,E1,A3,R3,+C5,+D3,G3
,+C5,R3,A5,+C3,A3,G3,A3,B3,+C3,+E1,R2,G1
,#F3,G3,+D1,R2,G1,#F3,G3,+C5
3000 A=54080:B=86:C=177:RETURN
3100 FORW=0T02:POKEA+W+2048,177:NEXT:POK
EA,B:POKEA+1,B-2:POKEA+2,B-4:RETURN
3200 FORT=1T010-Z:TM=TM-1:CURSOR35,23:PR
INT[7,0]USING"####";TM;:IFTM=0THEN230
3210 IF(Y=0)*(I=0)THENL=E+40:Y=1
3220 IFY=1THENPOKEL,0:L=L+40:POKEL,238:P
OKEL+2048,113:IFL>54043THENY=0:POKEL,0
3230 IF(L=A)+(L=A+1)+(L=A+2)THEN240
3240 GETZ$:IFZ$=""THEN3380
3250 IF(Z$=CHR$(32))*(X=0)*(S=0)THENM=A-
39:POKEM,53:X=1:GOTO3310
3260 IFZ$="@"THENU=1:GOTO3380
```

```
3270 IFZ$="@"THENU=2:GOTO3380
3280 IFZ$="@"THENU=3:GOTO3380
3290 IFZ$="@"THENU=4:GOTO3380
3300 GOSUB3100
3310 IFX=1THENPOKEM,0:IF(M=E+40)+(M=E+39
)+(M=E+38)THEN340
3320 M=M-40:IFM<53733THENX=0:GOTO3340
3330 IFX=1THENPOKEM,53:POKEM+2048,113
3340 NEXTT:RETURN
3350 N=N+1:IF(N=15)+(N=50)+(N=100)THEN33
70
3360 CURSOR15,23:PRINT[7,0]N;:RETURN
3370 FORH=0T03000STEP25:CURSOR35,23:PRIN
T[7,0]USING"####";H;:GOSUB5030:NEXT:TM=3
000:GOTO3360
3380 ONVGT03390,3410,3430,3450
3390 POKEA+2050,17:A=A-1:IF(A=53732)+(A=
53772)+(A=53812)+(A=53852)+(A=53892)+(A=
53932)+(A=53972)+(A=54012)+(A=54052)THEN
A=A+1
3400 GOTO3300
3410 POKEA+2048,17:A=A+1:IF(A=53761)+(A=
53801)+(A=53841)+(A=53881)+(A=53921)+(A=
53961)+(A=54001)+(A=54041)+(A=54081)THEN
A=A-1
3420 GOTO3300
3430 POKEA+2048,17:POKEA+2049,17:POKEA+2
050,17:A=A-40:IFA<53762THENA=A+40
3440 GOTO3300
3450 POKEA+2048,17:POKEA+2049,17:POKEA+2
050,17:A=A+40:IFA>54092THENA=A-40
3460 GOTO3300
5000 MUSIC"+C1":RETURN
5010 MUSIC"A1B1C1D1E1F1G1":RETURN
5020 MUSIC"-F0":RETURN
5030 MUSIC"+C0":RETURN
5050 FORPS=1T025:NEXTPS:RETURN
5060 FORPS=1T0200:NEXTPS:RETURN
5070 FORPS=1T02000:NEXTPS:RETURN
```



Disegnare con il computer



Un programma su misura per novelli artisti informatici, che offre la possibilità di creare ben 71 caratteri ridefinendoli DA TASTIERA con i codici esadecimali, più 15 caratteri già predefiniti per l'uso immediato.

IL PROGRAMMA

Creare la «tavolozza» di base scegliendo il colore del video (da 1 a 16), ed immettendo nelle 11 istruzioni di INPUT successive, il codice del colore del carattere e, dopo una virgola, il codice del colore di sfondo dello stesso.

Fatto questo, saranno visualizzati il colore prescelto e 2 cursori.

Spostarli con i soliti tasti E, X, S, D disposti in modo da richiamare i movimenti ad essi assegnati e digitare uno dei tasti con i caratteri predefiniti, di cui più avanti si ritroverà il PATTERN ad essi assegnato (H, I, J, K, L, M, N, O, P, Q, R, T, U, V, W).

Il carattere apparirà sullo schermo nel punto di incontro dei due cursori; per cancellare tale carattere o qualsiasi altro, premere la barra spaziatrice.

CREAZIONE DI NUOVI CARATTERI

Premere contemporaneamente i tasti SHIFT e 1: si udirà un segnale acustico; premendo SHIFT e \$ apparirà il codice ASCII 89. Se

non si vuole ridisegnare tale carattere tenere premuti i suddetti tasti e le cifre scorreranno in ordine crescente sino al 159 per poi ritornare a 89 e così via.

Scelto il carattere da mutare, digitare normalmente le 16 cifre esadecimali che corrispondono al PATTERN dello stesso (vedi il manuale di istruzioni alle pagg. 83 e seguenti): in un istante il nuovo carattere verrà visualizzato nel punto di incontro dei cursori.

Da questo momento in poi basterà premere semplicemente il tasto corrispondente al codice ASCII prescelto, per vedere visualizzato il nuovo carattere ogni volta che occorrerà.

A tal proposito bisogna ricordare che per richiamare i caratteri dal 128 al 159 si deve digitare CTRL e , (virgola) per il 128; CTRL e . (punto) per il 155; CTRL e : (due punti) per il 156; CTRL e + (più) per il 157; CTRL e 8 per il 158; CTRL e 9 per il 159 ed infine per i codici compresi tra 129 e 154 basterà digitare CTRL e le lettere da A a Z in ordine alfabetico.

Dulcis in fundo, è possibile cambiare colore allo schermo fra i 16 a disposizione digitando SHIFT e + (più).

Buon divertimento per le vostre ore a tutto schermo!



VARIABILI PRINCIPALI

R	Riga cursore	X	Colonna dello schermo
C	Colonna cursore	ESA	Conteggio delle 16 cifre esadecimali
CA	Caratteri ridefinibili con il codice esadecimale	Z\$	Stringa esadecimale ottenuta
CA\$	Stringa del codice ASCII carattere	PATTERNS CARATTERI PREDEFINITI	

```

10 CALL CLEAR
20 CALL SCREEN(11)
30 PRINT "DISEGNARE COL COMPUTER
      DADO 1984"
40 FOR TTT=1 TO 2000
50 NEXT TTT
60 CALL CLEAR
70 INPUT "COLORE VIDEO ? ":X
80 INPUT "INS.6 CARATTERE ? SFONDO ? ":
C6,S6
90 INPUT "INS.7 CARATTERE ? SFONDO ? ":
C7,S7
100 INPUT "INS.8 CARATTERE ? SFONDO ? "
:C8,S8
110 INPUT "INS.9 CARATTERE ? SFONDO ? "
:C9,S9
120 INPUT "INS.10 CARATTERE ? SFONDO ? "
:C10,S10
130 INPUT "INS.11 CARATTERE ? SFONDO ? "
:C11,S11
140 INPUT "INS.12 CARATTERE ? SFONDO ? "
:C12,S12
150 INPUT "INS.13 CARATTERE ? SFONDO ? "
:C13,S13
160 INPUT "INS.14 CARATTERE ? SFONDO ? "
:C14,S14
170 INPUT "INS.15 CARATTERE ? SFONDO ? "
:C15,S15
180 INPUT "INS.16 CARATTERE ? SFONDO ? "
:C16,S16
190 CALL CLEAR
200 CALL SCREEN(X)
210 R=2
220 C=2
230 CA=8B
240 CALL CHAR(40,"AAAAAAAAAAAAAAAA")
250 CALL CHAR(41,"AAAAAAAAAAAAAAAA")
260 CALL COLOR(2,2,16)
270 CALL HCHAR(1,C,40)
280 CALL HCHAR(R,1,41)
290 CALL CHAR(72,"0102040810204080")
300 CALL CHAR(73,"8040201008040201")
310 CALL CHAR(74,"8080808080808080")
320 CALL CHAR(75,"0101010101010101")
330 CALL CHAR(76,"FF")
340 CALL CHAR(77,"00000000000000FF")
350 CALL CHAR(78,"FF818181818181FF")
360 CALL CHAR(79,"FFFFFFFFFFFFFFFF")
370 CALL CHAR(80,"8181818181818181")
380 CALL CHAR(81,"FF000000000000FF")
390 CALL CHAR(82,"FF80808080808080")
400 CALL CHAR(84,"FF01010101010101")
410 CALL CHAR(85,"8080808080808080")
420 CALL CHAR(86,"01010101010101FF")
430 CALL CHAR(87,"3C7EFFFFFFF7E3C")
440 CALL COLOR(3,2,16)
450 CALL COLOR(4,2,16)
460 CALL COLOR(5,2,16)
470 CALL COLOR(6,C6,S6)
480 CALL COLOR(7,C7,S7)
490 CALL COLOR(8,C8,S8)

```

```

500 CALL COLOR(9,C9,S9)
510 CALL COLOR(10,C10,S10)
520 CALL COLOR(11,C11,S11)
530 CALL COLOR(12,C12,S12)
540 CALL COLOR(13,C13,S13)
550 CALL COLOR(14,C14,S14)
560 CALL COLOR(15,C15,S15)
570 CALL COLOR(16,C16,S16)
580 CALL KEY(0,K,S)
590 IF S=0 THEN 580
600 IF (K=88)+(K=120)THEN 710
610 IF (K=69)+(K=101)THEN 760
620 IF (K=68)+(K=100)THEN 810
630 IF (K=83)+(K=115)THEN 860
640 IF K=43 THEN 1020
650 IF K=33 THEN 1070
660 IF K<>88 THEN 670
670 IF K<>69 THEN 680
680 IF K<>68 THEN 690
690 IF K<>83 THEN 910
700 GOTO 580
710 R=R+1
720 IF R>23 THEN 940
730 CALL HCHAR(R,1,41)
740 CALL HCHAR(R-1,1,32)
750 GOTO 580
760 R=R-1
770 IF R<2 THEN 960
780 CALL HCHAR(R,1,41)
790 CALL HCHAR(R+1,1,32)
800 GOTO 580
810 C=C+1
820 IF C>32 THEN 980
830 CALL HCHAR(1,C,40)
840 CALL HCHAR(1,C-1,32)
850 GOTO 580
860 C=C-1
870 IF C<2 THEN 1000
880 CALL HCHAR(1,C,40)
890 CALL HCHAR(1,C+1,32)
900 GOTO 580
910 CALL HCHAR(R,C,K)
920 CALL SOUND(100,K*10,10)
930 GOTO 580
940 R=23
950 GOTO 580
960 R=2
970 GOTO 580
980 C=32
990 GOTO 580
1000 C=2
1010 GOTO 580
1020 X=X+1
1030 IF X>16 THEN 1040 ELSE 1050
1040 X=1
1050 CALL SCREEN(X)
1060 GOTO 470
1070 CALL SOUND(100,1760,5)
1080 ESA=3
1090 CALL KEY(0,K,S)
1100 IF S=0 THEN 1090

```

```

1110 IF K=36 THEN 1580
1120 IF ((K>47)*(K<58))+((K<64)*(K<71))T
HEN 1140
1130 GOTO 1090
1140 CALL HCHAR(24,ESA,K)
1150 ESA=ESA+1
1160 IF ESA=19 THEN 1180
1170 GOTO 1090
1180 CALL GCHAR(24,3,A)
1190 A$=CHR$(A)
1200 CALL GCHAR(24,4,B)
1210 B$=CHR$(B)
1220 CALL GCHAR(24,5,CC)
1230 C$=CHR$(CC)
1240 CALL GCHAR(24,6,D)
1250 D$=CHR$(D)
1260 CALL GCHAR(24,7,E)
1270 E$=CHR$(E)
1280 CALL GCHAR(24,8,F)
1290 F$=CHR$(F)
1300 CALL GCHAR(24,9,G)
1310 G$=CHR$(G)
1320 CALL GCHAR(24,10,H)
1330 H$=CHR$(H)
1340 CALL GCHAR(24,11,I)
1350 I$=CHR$(I)
1360 CALL GCHAR(24,12,L)
1370 L$=CHR$(L)
1380 CALL GCHAR(24,13,M)
1390 M$=CHR$(M)
1400 CALL GCHAR(24,14,N)
1410 N$=CHR$(N)
1420 CALL GCHAR(24,15,O)
1430 O$=CHR$(O)
1440 CALL GCHAR(24,16,P)
1450 P$=CHR$(P)
1460 CALL GCHAR(24,17,Q)
1470 Q$=CHR$(Q)
1480 CALL GCHAR(24,18,RR)
1490 R$=CHR$(RR)
1500 X$=A$&B$&C$&D$&E$&F$&G$&H$
1510 Y$=I$&L$&M$&N$&O$&P$&Q$&R$
1520 Z$=X$&Y$
1530 CALL CHAR(CA,Z$)
1540 CALL HCHAR(R,C,CA)
1550 ESA=3
1560 CALL HCHAR(24,1,32,32)
1570 GOTO 580
1580 REM
1590 CA=CA+1
1600 IF CA>159 THEN 1670
1610 CA$=STR$(CA)
1620 FOR W=1 TO LEN(CA$)
1630 AS=ASC(SEG$(CA$,W,1))
1640 CALL HCHAR(24,(24+W)-LEN(CA$),AS)
1650 NEXT W
1660 GOTO 1080
1670 CA=8B
1680 CALL HCHAR(24,1,32,32)
1690 GOTO 1080

```



Supersnake

«C'era una volta un serpentello
che solo mosche voleva mangiare
pensando così di volare.
Ma non si rendeva conto, il poverello,
che anziché volare quel cibo
lo faceva allungare...»
Le istruzioni sono inserite nel listato.

```

1 REM *****
2 REM
3 REM   S U P E R S N A K E
4 REM
5 REM   Z X - 8 1 1 6 K
6 REM
7 REM *****
8 REM
9 REM ** COMANDI **
10 REM A = SINISTRA
11 REM D = DESTRA
12 REM O = BASSO
13 REM L = ALTO
14 REM I = PAUSA
15 REM
16 REM
17 DIM C(690)
18 RAND
19 GOSUB 6000
20 LET DF=16396
21 LET DF=67+PEEK DF+256*PEEK
22 (+1)
23 LET REC=0
24 LET CR=0
25 LET PU=0
26 LET TT=PU+40
27 REM
28 REM QUADRO INIZIALE
29 REM
30 CLS
31 PRINT "PUNTI = ";PU," CR
32 = ";CR,AT 0,31;"
33 LET DP=DF+693
34 LET DO=DF-33
35 FOR X=0 TO 31
36 POKE DO+X,120
37 POKE DP+X,120
38 NEXT X
39 LET DO=DF+31
40 FOR X=0 TO 20
41 POKE DF+33*X,120
42 POKE DO+33*X,120
43 NEXT X
44 REM
45 REM BRUCO
46 REM
47 LET W=690+DF
48 LET A=W
49 LET S=-33
50 LET C=(W-DF)=5
51 FOR X=1 TO 10+CR
52 POKE W,52
53 LET W=W+C(W-DF)
54 GOSUB USA 16514
55 LET U=PEEK (W+S)
56 IF U THEN GOTO 9000
57 LET C=(W-DF)=5
58 NEXT X
59 REM
60 REM INSETTI
61 REM
62 FOR X=1 TO INT (2*RND)+4
63 LET SA=INT (690*RND)
64 LET SAN=PEEK (DF+SA)
65 IF SAN THEN GOTO 720
66 POKE SA+DF,23
67 NEXT X
68 REM
69 REM LOOP PRINCIPALE
70 REM
71 POKE W,52
72 POKE A,0

```

```

830 GOSUB USA 16514
840 LET U=PEEK (W+S)
850 IF U=23 THEN GOTO 2000
860 IF U THEN GOTO 9000
870 LET A=A+C(A-DF)
880 LET C=(W-DF)=5
890 LET W=W+S
900 GOTO 600
1000 REM
1010 REM DIREZIONE
1020 REM
1030 RETURN
1040 LET S=-1
1050 RETURN
1060 LET S=1
1070 IF TT-PU<40 AND RND<.5 THEN
1080 GOSUB 3000
1090 RETURN
1100 LET S=33
1110 IF TT-PU<40 AND RND<.5 THEN
1120 GOSUB 3000
1130 RETURN
1140 LET S=-33
1150 IF TT-PU<40 AND RND<.5 THEN
1160 GOSUB 3000
1170 RETURN
1180 PAUSE 4E4
1190 RETURN
20000 REM
20005 REM MOSCA PRESA
20010 REM
20015 LET PU=PU+10
20020 PRINT AT 0,10;PU
20030 GOTO 600
20040 REM
20050 REM NUOVA MOSCA
20060 REM
20070 REM
20080 LET SA=INT (690*RND)
20090 LET SAN=PEEK (DF+SA)
20100 IF SAN THEN GOTO 3000
20110 POKE SA+DF,23
20120 LET TT=TT+10
20130 RETURN
20140 REM
20150 REM ROUTINE L.M.
20160 LET D$="2051670021240471980
20170 76079005004201"
20180 FOR X=16514 TO 16524
20190 POKE X,VAL D$( TO 3)
20200 LET D$=D$(4 TO )
20210 NEXT X
20220 RETURN
20230 REM
20240 REM CRASH
20250 REM
20260 POKE W,52
20270 LET CR=CR+1
20280 LET O=W+S
20290 PRINT AT 0,28;CR
20300 IF CR=3 THEN GOTO 9200
20310 FOR X=1 TO 10
20320 POKE O,U
20330 RAND (((((2**2))))))
20340 POKE O,160
20350 NEXT X
20360 GOTO 200
20370 REM
20380 REM FINE
20390 REM
20400 POKE O,160
20410 PRINT AT 10,3;"*****
20420 *****";AT 11,3;"*
20430 RECORD = ";AT 11,21;REC;
20440 AT 12,3;"*****";
20450 *";AT 15,0;"UOUI GIOCARE ANCORA
20460 ?
20470 IF PU>MAX THEN PRINT AT 11,
20480 16;"ERA"
20490 PRINT AT 14,5;"
20500 ";AT 16,5;"
20510 IF PU>MAX THEN PRINT AT 4,3
20520 :";TAB
20530 3;"
20540 TAB 3;"
20550 TAB 3;"
20560 IF PU>MAX THEN LET MAX=PU
20570 INPUT I$
20580 IF I$<>"N" THEN GOTO 100
20590 STOP

```



Il gatto e il topo

Un topolino è rimasto intrappolato in un labirinto formato da cubi. Per uscire dal labirinto il topo deve passare sui cubi in modo da far cambiare loro colore.

L'impresa sembrerebbe semplice, ma ad ostacolare il povero topolino ci sono delle pareti mobili, che si spostano casualmente all'interno del labirinto e un gatto che, naturalmente, dà la caccia al nostro eroe con l'intento di trasformarlo in un lauto pasto che soddisfi i suoi appetiti.

Ulteriori istruzioni sono inserite nel listato. Attenzione! Le lettere maiuscole inserite tra virgolette vanno digitate dopo essere entrati in modo grafico (Graphics).

```
1 REM #####
2 REM #
3 REM # IL GATTO E IL TOPO #
4 REM #
5 REM # ZX-SPECTRUM 16K/48K #
6 REM #
7 REM # L I S T 1 9 8 5 #
8 REM #
9 REM #####
10 REM
15 REM
20 CLEAR USR "a"-60: LET r=USR
  "a"-60: LET t=r+5: LET w=r+26
30 RESTORE : PAPER 7: INK 0: B
ORDER 7: CLS : GO TO 6000
1050 LET g=y+3*((INKEY$="w" AND
ATTR (y+2,x)=32)-(INKEY$="2" AND
ATTR (y-1,x)=32))
1060 LET h=x+3*((INKEY$="0" AND
ATTR (y,x+2)=32)-(INKEY$="9" AND
ATTR (y,x-1)=32))

1070 IF ATTR (g,h)=32 THEN LET b
=b+1: LET pu=pu+5: PRINT #0;AT 0
,15-LEN STR$ PU;PU
1080 PRINT PAPER 6;AT y,x);" ";A
T y+1,x);" ": LET y=g: LET x=h
1090 PRINT PAPER 6;AT y,x);b$(e T
O e+1);AT y+1,x);c$: RETURN
1120 PRINT PAPER 6;AT c,d);" ";A
T c+1,d);" "
1145 LET d=d+3*((d<x AND ATTR (c
,d+2)=32)-(d>x AND ATTR (c,d-1)=
32))
1160 PRINT FLASH 1; PAPER 8;AT c
,d);d$(e TO e+1);AT c+1,d);e$(e TO
e+1)
1165 IF ATTR (y,x)>159 THEN GO T
O 3100
1170 RETURN
1176 PRINT PAPER 8;AT c,d);" ";A
T c+1,d);" "
```

```
1180 LET c=c+3*((c<y AND ATTR (c
+2,e)=32)-(c>y AND ATTR (c-1,d)=
32))
1190 PRINT FLASH 1; PAPER 8;AT c
,d);d$(e TO e+1);AT c+1,d);e$(e TO
e+1)
1195 IF ATTR (y,x)>159 THEN GO T
O 3100
1200 RETURN
1300 PRINT AT 3*(1+INT (RND*6)),
1+3*(INT (RND*10)); PAPER 4);" "
: RETURN
1310 LET k=1+3*INT (RND*7): LET
j=3*(1+INT (RND*9)): PRINT PAPER
4;AT k,j);" ";AT k,j);" ";AT k+1,
j);" ": RETURN
1320 LET k=1+3*INT (RND*7): LET
j=3*(INT (RND*10)): PRINT AT k,j
;"M";AT k+1,j);"N": RETURN
1340 PRINT AT 3*(1+INT (RND*6)),
3*(INT (RND*10));"OPQ": RETURN
```

```
1600 RANDOMIZE USR t: RANDOMIZE
USR w: RETURN
2004 IF INKEY$="m" THEN GO SUB 4
000
2005 IF b>69 THEN GO TO 3000
2010 FOR e=1 TO 3 STEP 2
2050 GO SUB 1040
2060 GO SUB 1300: GO SUB 1310
2100 GO SUB 1115
2110 GO SUB 1300: GO SUB 1310
2115 NEXT e: FOR e=1 TO 3 STEP 2
2120 GO SUB 1040
2130 GO SUB 1320: GO SUB 1340
2150 GO SUB 1175
2999 NEXT e: GO TO 2000
3010 PRINT FLASH 1;AT 0,1);"
LABIRINTO COMPLETO "
3020 FOR l=20 TO 0 STEP -2: PRIN
T AT l,1;"AB";AT l+1,1;"CD";AT l
,26;"AB";AT l+1,26;"CD": BORDER
1: GO SUB 1600: BORDER 5: NEXT l

3030 FOR l=1 TO 6: FOR k=1 TO 3:
RANDOMIZE USR t: BORDER 1: NEXT
k: FOR k=1 TO 3: RANDOMIZE USR
w: BORDER 7: NEXT k: NEXT l
3099 GO TO 7500
3120 FOR l=0 TO 255 STEP 10: POK
E (w+7),l: RANDOMIZE USR w: BORD
ER 2: RANDOMIZE USR w: BORDER 7:
NEXT l: POKE (w+7),0
3130 LET vi=vi-1: IF vi=0 THEN G
O TO 3200
3199 PRINT PAPER 6;AT c,d);" ";AT
c+1,d);" ": GO TO 8000
3200 FOR l=9 TO 13: PRINT AT l,3
);" ": NEXT
l
3210 PRINT AT 10,10;"GAME OVER"
;AT 12,4;"PREMI 'Z' PER GIOCARE"
3220 PRINT #0;AT 0,24);" ";AT 1,
24);" 0"
```



```

3230 FOR l=0 TO 255 STEP 5: POKE
(t+7),l: RANDOMIZE USR t: NEXT
l: POKE (t+7),0
3240 IF INKEY$="z" THEN GO TO 32
40
3299 GO TO 7000
4010 PRINT PAPER 6;AT y,x);" ";A
T y+1,x);" "
4015 GO SUB 1600: GO SUB 1600
4016 LET pu=INT (pu-pu/10): PRIN
T #0;AT 0,15-LEN STR$ pu;pu
4019 LET x=29-x: LET y=20-y
4050 RETURN
6005 DATA 14,17,35,36,31,7,7,7,0
,30,225,241,145,242,236,192,1,4,
8,20,4,7,30,30,128,64,160,160,16
0,192,240,240,0,120,135,143,137,
79,55,3,112,136,196,36,246,224,2
24,224

```

```

6006>DATA 0,0,0,36,102,126,90,12
7,14,2,2,2,14,62,252,246,60,25,2
,4,9,16,4,0,244,210,164,136,0,0,
0,0,0,0,14,2,2,194,242,254,254
,166,152,20,18,36,8

```

```

6007 DATA 1,3,5,9,17,17,17,17,17
,17,17,17,17,17,17,17,17,19,23,3
1,31,0,0,0

```

```

6009 DATA 255,255,255,255,255,0,
0,0,255,254,252,248,240,0,0,0

```

```

6020 DATA 32,24,16,167,35,58,72,
92,15,15,15,30,0,243,211,254,238
,16,67,16,254,29,32,246,251,201

```

```

6021 DATA 58,72,92,15,15,15,30,1
,243,211,254,238,16,67,16,254,28
,32,246,251,201

```

```

6100 LET pu=0: LET rec=0: LET b$
="ABEF": LET c$="CD": LET d$="GH
GK": LET e$="IJIL"

```

```

6505 CLS : PRINT AT 0,2;"IL G A
T T O E I L T O P O"

```

```

6510 PRINT FLASH 1; OVER 1;AT 0,
0;"

```

```

6530 PRINT " IN QUESTO GIOCO T
U SEI UN TOPORIMASTO IMPRIGIONAT
O IN UN LABI-RINTO FORMATO DA CU
BI. PER VINCERE DEVI PA
SSARE SUI CU-BI IN MODO CHE QUE
STI CAMBINO COLORE."

```

```

6535 PRINT " NEL LABIRINTO CI
SONO DELLE PARETI MOBILI CHE O
STACOLANO IL TUO CAMMINO.

```

```

NEL LABIRINTO C'E'
ANCHE UN GAT-TO:SE QUESTI TI PRE
NDE,PERDI UNAVITA."

```

```

6536 PRINT "IN CASO DI EMERGENZA
PUI SPARIREMA CIO' TI COSTA IL 1
0% DEL TUO PUNTEGGIO."

```

```

6537 PAUSE 0
6539 CLS : INK 2: LET l=10

```

```

6540 PRINT 'TAB l-2; INVERSE 1;"
C O M A N D I "

```

```

6545 PRINT TAB l;"ALTO = 2"
6550 PRINT TAB l;"BASSO = U"
6551 PRINT TAB l;"SINISTRA = 9"
6552 PRINT TAB l;"DESTRA = 0"
6553 PRINT TAB l;"HELP = M"

```

```

6555 FOR l=USR "a" TO USR "q"+7:
READ k: POKE l,k: NEXT l

```

```

6556 FOR l=r TO r+46: READ k: PO
KE l,k: NEXT l: GO SUB 1600

```

```

6560 PRINT #0;AT 1,4; INVERSE 1;
INK 2;"PREMI 'X' PER INIZIARE"

```

```

6999 IF INKEY$<>"x" THEN GO TO 6
999

```

```

7009 IF pu>rec THEN LET rec=pu
7010 LET pu=0: LET vi=3

```

```

7600 PAPER 6: CLS : FOR l=26 TO
0 STEP -1

```

```

7610 PRINT AT 11,l;"AB GH";AT 1
2,l;"CD IJ"

```

```

7620 POKE (t+7),l*9+1: RANDOMIZE
USR t: RANDOMIZE USR t

```

```

7630 PRINT AT 11,l;"EF GK";AT 1
2,l;"CD IL": RANDOMIZE USR t: R

```

```

ANDOMIZE USR t: PRINT AT 11,l;"
";AT 12,l);" ": NEXT l:

```

```

POKE (t+7),0: INK 0: PAPER 7

```

```

7800 LET b=1: CLS : PRINT AT 0,2
;"IL G A T T O E I L T O P O"

```

```

7810 PRINT FLASH 1; OVER 1; INK
4;AT 0,1;"

```

```

7920 FOR l=1 TO 20: PRINT PAPER
4;AT l,1;"

```

```

": NEXT l

```

```

7925 FOR l=1 TO 19 STEP 3: PRINT
AT l,0;"M";AT l+1,0;"N";AT l+2,
0;"O": NEXT l

```

```

7926 PRINT AT 21,0;"OPQPQPQPQPQ
OPQPQPQPQPQPQPQP"

```

```

7927 FOR l=3 TO 27 STEP 3: FOR k
=3 TO 18 STEP 3: PRINT AT k,l;"O
": NEXT k: NEXT l

```

```

7960 RANDOMIZE : FOR l=1 TO 5: P
RINT AT 3*(1+INT (RND*6)),3*(INT
(RND*10));"OPQ": NEXT l

```

```

7980 FOR l=1 TO 5: LET k=1+3*INT
(RND*7): LET j=3*(1+INT (RND*9)
): PRINT AT k,j;"M";AT k+1,j;"N"

```

```

: NEXT l

```

```

8003 PRINT #0;AT 0,1;"PUNTI 00
000";AT 1,19;"VITE ";AT 1
,1;"RECORD 00000";AT 0,24;"

```

```

"

```

```

8004 PRINT #0;AT 0,15-LEN STR$ p
u;pu;AT 1,15-LEN STR$ rec;rec

```

```

8006 PRINT #0;AT 1,24; FOR l=1
TO vi: PRINT #0;"AB";: NEXT l

```

```

8090 LET y=10: LET x=4: LET c=1:
LET d=28: LET l=1: GO SUB 1600

```

```

8095 PRINT PAPER 6;AT y,x;"EF";A
T y+1,x;c$: GO TO 2000

```



Oric sequencer



Nella musica moderna, e specialmente nella disco-dance, è ormai diventato diffusissimo l'uso del SEQUENCER. Ma cosa è il SEQUENCER? Non è altro che un immagazzinatore di dati

che servono a far suonare una tastiera elettronica, in genere un sintetizzatore. I dati che gli devono venire forniti sono: le note, la durata delle note e le pause come si leggono da uno spartito musicale. Fatto ciò il SEQUENCER è in grado di suonare melodie e perfino intere canzoni

```

1 REM#####
2 REM# #
3 REM# O R I C - S E Q U E N C E R #
4 REM# #
5 REM# di Fabrizio Russo #
6 REM #
7 REM# list - 1985 #
8 REM#####
9 GOSUB6000
10 PAPER0:INK7
11 HIRE:PRINTCHR*(17)CHR*(6)
12 PRINTSPC(11)CHR*(4)CHR*(27)"NORIC-SEQ
UENCER"CHR*(4);
14 RE DISEGNO TASTIERA
15 CURSET17,101,3:DRAW0,-25,1:DRAW222,0,
1:DRAW0,25,1:DRAW-222,0,1
17 GOSUB155
20 FORT=23T0220STEP35
30 CURSETT,100,3
40 DRAW0,50,1:DRW5,0,1:DRAW0,-25,1
50 GOSUB150
60 CURMOV-2,49,3:DRW6,0,1:DRAW0,-25,1:G
OSUB150
70 CURMOV-2,49,3:DRW6,0,1:DRAW0,-49,1:C
URMOV0,49,3
80 DRW5,0,1:DRAW0,-25,1:GOSUB150
90 CURMOV-2,49,3:DRW6,0,1:DRAW0,-25,1:G
OSUB150
100 CURMOV-2,49,3:DRW6,0,1:DRAW0,-25,1:
GOSUB150
110 CURMOV-1,49,3:DRW5,0,1:DRAW0,-49,1
120 NEXT
140 GOTO200
150 CURMOV-1,-25,3:DRAW0,25,1:CURMOV1,1,

```

```

3:DRAW0,-25,1:CURMOV1,25,3:
DRAW0,-25,1
151 GOSUB155:RETURN
152 REM SUONO MOTIVETTO INIZIALE
155 READA,B
156 IFA=11THEN158
157 MUSIC1,B,A,0:PLAY1,0,1,3000:WAIT10
158 READA,B:IFA=11THEN160
159 MUSIC1,B,A,0:PLAY1,0,1,2000
160 RETURN
200 A$="ORIC-SYNT"
210 CURSET170,90,3:GOSUB300

220 A$="Poly-3"
230 CURSET20,80,3:GOSUB300
237 Z=20:DIMD*(256),E*(256)
240 GOSUB1000:GOSUB1152:RUN237
300 FORT=1TOLN(A$)
310 CURMOV0,0,3
320 CHARASC(MID$(A$,T,1)),0,1
330 NEXTT
340 RETURN
997 REM
998 REM ROUTINE PRINCIPALE
999 REM
1000 GOSUB3100
1001 CLS:Q=0
1002 PRINT"1'=DO 2'=DO# 3'=RE 4'=RE#
5'=MI 6'=FA 7'=FA# ~8'
=SOL ";
1004 PRINT"9'=SOL# 0'=LA'-'=LA# '9'=SI
";B#;
1005 IFY=1THENGOSUB1152:GOTO1006
1006 N=1:O=3:H=1:T=1:A1=0:B1=0:C=131:C1=
140:Y=1:S=0
1008 GOSUB1160
1009 REM ##### RICHIESTA TASTO O NOTA
1010 GETK#
1020 A=ASC(K#)
1022 IFA=8THENO=0-1:GOTO1190
1025 IFA=9THENO=0+1:GOTO1190
1027 IFA=65THENO=0:GOSUB1152:GOSUB1160

```

```

1028 IFA=80THENO=Q+1:D*(Q)=14:GOTO1010
1030 IFA=48THENN=10:C=156:C1=140:H=1:GOT
O1150
1040 IFA=49THENN=1:C=131:C1=140:H=1:GOTO
1150
1050 IFA=50THENN=2:C=133:C1=120:H=0:GOTO
1150
1060 IFA=51THENN=3:C=136:C1=140:H=1:GOTO
1150
1070 IFA=52THENN=4:C=138:C1=120:H=0:GOTO
1150
1080 IFA=53THENN=5:C=141:C1=140:H=1:GOTO
1150
1090 IFA=54THENN=6:C=146:C1=140:H=1:GOTO
1150
1100 IFA=55THENN=7:C=148:C1=120:H=0:GOTO
1150
1110 IFA=56THENN=8:C=151:C1=140:H=1:GOTO
1150
1120 IFA=57THENN=9:C=153:C1=120:H=0:GOTO
1150
1130 IFA=45THENN=11:C=158:C1=120:H=0:GOT
O1150
1140 IFA=61THENN=12:C=161:C1=140:H=1:GOT
O1150
1141 IFA=13ANDF1=1THENC=Q:GOTO2000
1142 IFA=13THENCLS:RETURN
1143 IFA=32THENO=Q+1:D*(Q)=13:E*(Q)=0:G
OTO1144
1144 IFQ=256THENC=Q:GOTO2002
1145 GOTO1010
1150 C=C+S

1151 Q=Q+1:D*(Q)=N:E*(Q)=0:GOSUB1152:GOS
UB1160:GOTO1182
1152 IFH1=1THENN1=0ELSEH1=1
1155 CURSETA1,B1,H1:CIRCLE1,H1:RETURN
1160 CURSETC,C1,0:CIRCLE1,H:A1=C:B1=C1:H
1=H
1170 MUSIC1,0,N,0
1180 PLAY1,0,1,2000:RETURN
1182 IFQ=256THENC=Q:GOTO2000

```



automaticamente.

Il suo uso è quasi sempre ristretto a coprire il ruolo di accompagnamento, mentre il motivo principale viene eseguito ancora (per fortuna! n.d.r.) manualmente.

Nel caso specifico, i dati forniti al vostro ORIC I non sono che valori da assegnare ad

istruzioni MUSIC e WAIT che poi il computer eseguirà con il suo generatore sonoro.

La programmazione delle sequenze può essere spesso difficoltosa, specialmente per chi di solfeggio e teoria musicale ne sa poco, ma il programma può essere un valido aiuto per il loro apprendimento.

```

1185 GOTO1010
1190 IFA=8AND0=-1THEN0=GOTO1010
1200 IFA=9AND0=6THEN0=5:GOTO1010
1210 IFA=8THENS=S-35ELSE S=S+35
1220 GOTO1010
1997 REM
1998 REM RIPRODUZIONE SEQUENZA
1999 REM
2000 IFQ=0THENQ=1:GOTO2065
2002 CLS:C$=" PLAY "
2003 B$=CHR$(128)+CHR$(151)+C$+CHR$(144)
2005 PRINTSPC(90)B$;
2010 A3=2
2020 FORQ=1TOC
2025 IFD*(Q)=14THENMUSIC1,0,N,15:GOTO2050
2030 IFD*(Q)=13THENA3=2+2:GOTO2050
2032 N=D*(Q):O=E*(Q)
2040 MUSIC1,0,N,0
2045 PLAY1,0,1,2000
2050 WAITA3
2055 NEXT:WAIT50:PLAY0,0,0,0:WAIT50
2065 CLS:PRINT"N. Steps = ";Q-1:PRINT
2070 PRINTSPC(11)"Vuoi continuare?";
2075 A$=""
2080 GETA$:IFA$<"S"AND A$<"N"THEN2080
2090 IFA$="S"THENGOSUB3500:CLS:GOTO2000
2095 CLS:RETURN
2097 REM
2098 REM ROUTINE SCELTA MODO
2099 REM
3100 A$="Vuoi registrare(1) o solo suonare(2)?"
3105 CURSET(255-(LEN(A$)*6))/2,30,3
3110 FORT=1TOLEN(A$)
3120 CHAR(ASC(MID$(A$,T,1))),0,1
3130 CURMOV6,0,3:NEXT
3135 A$=""
3140 GETA$:IFA$<"1"AND A$<"2"THENPRINTCHR$(6):TEXT:END
3150 IFA$="1"THEN3152FLSE3160
    
```

```

3152 C$="RECORD "
3155 B$=CHR$(128)+CHR$(151)+CHR$(140)+C$+CHR$(144)
3157 F1=1:GOTO3170
3160 F1=0:B$=""
3170 FORT=1TO234STEP6
3180 CURSETT,30,3:CHAR127,0,0:NEXT
3200 RETURN
3497 REM
3498 REM RICHIESTA VARIAZIONE TEMPO
3499 REM
3500 CLS:PRINT:PRINT:PRINTSPC(8)"Vuoi variare il tempo?";
3505 C$="RECORD "
3507 B$=CHR$(128)+CHR$(151)+CHR$(140)+C$+CHR$(144)
3510 GETA$:IFA$<"S"AND A$<"N"THEN3510
3520 IFA$="S"THEN3540
3530 S=0:RETURN
3540 CLS:PRINT"Premi i tasti cursore su e giu' per"
3550 PRINT"variare il tempo."
3560 PRINT"Premi RETURN per tornare.";
3562 WAIT300:A3=2
3565 CLS:PRINT:PRINT"TEMPO = "Z;
3570 A$=KEY$:IFA$=" "THEN3570
3573 A=ASC(A$):IFA<10ANDA<11ANDA<13THEN3570
3580 IFA=13THENCLS:RETURN
3600 IFA=10ANDZ=0THEN3570
3605 IFB=10THENZ=2-1:GOTO3620
3610 IFZ=100THEN3570
3615 Z=Z+1
3620 CLS:PRINT:PRINT"TEMPO8 = "Z;
3630 GOTO3570
4997 REM
4998 REM NOTE MOTIVETTO INIZIALE
4999 REM
5000 DATA1,3,11,,3,3,11,,5,3,11,,1,3,11,
5005 DATA1,3,11,,3,3,11,,5,3,11,,1,3,11,
5010 DATA5,3,11,,6,3,11,,8,3,11,,11,,11,
    
```

```

5020 DATA5,3,11,,6,3,11,,8,3,11,,11,,11,
5030 DATA8,3,10,3,8,3,6,3,5,3,11,,1,3,11,
5040 DATA8,3,10,3,8,3,6,3,5,3,11,,1,3,11,
5050 DATA3,3,11,,8,2,11,,1,3,11,,11,,18,
5060 DATA3,3,11,,8,2,11,,1,3,11,
6000 TEXT:CLS
6005 DOKE#12,48004:PRINT"O R I C - S E Q U E N C E R":CLS
6010 PAPER3:INK0
6020 PRINT"Il tuo ORIC-I puo' registrare e ripro-";
6030 PRINT"dur e qualsiasi melodia o sequenza di note musicali ."
6040 PRINT:PRINT"Per fare questo devi conoscere almeno un po' di teoria musi";
6050 PRINT"cale ed avere un minimo di 'orecchio'."
6060 PRINT:PRINT
6065 PRINT"I tasti di cui avrai bisogno sono:"
6070 PLOT1,11,"?A?=Rinializza la sequenza"
6080 PLOT1,13,"?SPACE-BAR?=Pausa"
6090 PLOT1,15,"?P?=Allunga l'ultima nota suonata"
6100 PLOT1,17,"?ENTER?=Cambia modo"
6110 PLOT1,19,"?CURSORE-SIN.?=Scende di ottava"
6120 PLOT1,21,"?CURSORE-DES.?=Sale di ottava"
6150 FORT=0TO13:PRINT:NEXT
6160 PRINT"Insime ad essi ci sono i tasti per lenote indicati sotto ";
6170 PRINT"al disegno della tastiera.";
6180 GETA$
6190 RETURN
    
```



IL 1° DISCO-MIX
DI PROGRAMMI PER
IL TUO:
TEXAS TI 99/4A
ZX SPECTRUM
VIC 20

Disc

DISCO LIST: NOTE

Prima di tutto tenere presente che **NON SI TRATTA DI UN FLOPPY DISK**, quindi non cercare di utilizzarlo tramite DISK-DRIVE.

Solo per la versione ZX SPECTRUM è possibile caricare il programma direttamente da giradischi, secondo la seguente procedura:

- collegare il cavetto fra la presa per cuffia del giradischi e la presa EAR del computer;
 - regolare il volume di ascolto sul valore medio ed escludere, se possibile, le casse acustiche.
- In tutti gli altri casi, riversare il programma da disco a nastro.

Nel caso si possieda un impianto HI-FI, regolare il volume di registrazione su un valore medio (VUmeter = MAX. +1 dB).

Provare quindi a caricare il programma.

Nel caso si incontrassero delle difficoltà, registrare nuovamente il programma più volte anche a livelli diversi.

Può essere utile, in questo caso, provare ad ascoltare il «suono» di un altro programma preregistrato e cercare quindi di ottenere le stesse tonalità con il programma su disco.

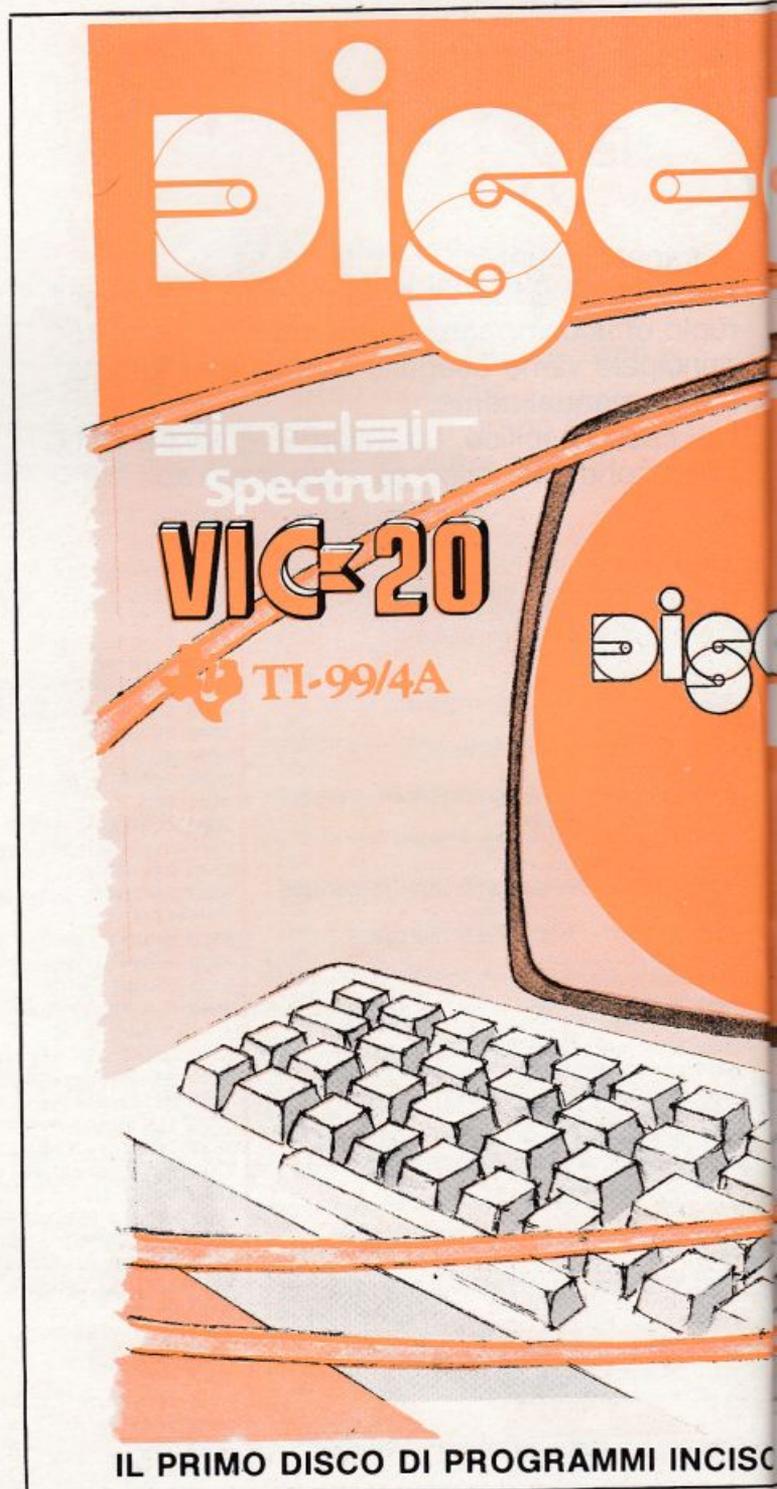
EGGSCAPE - LA FUGA DELL'UOVO

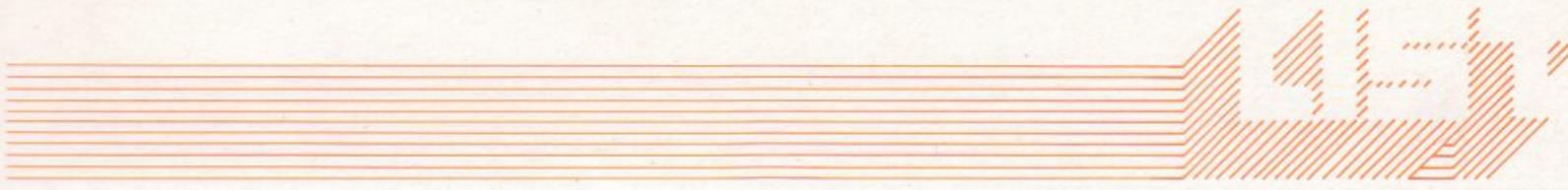
Un simpatico uovo di nome BERT deve essere guidato attraverso un labirinto buio per raggiungere l'entrata di un tunnel segreto.

Molti i pericoli che minacciano il povero BERT: JELLY il mostro gelatinoso a caccia di cibo, le GUARDIE DELLA MORTE ed i loro fidi PIPISTRELLI. Il gioco si svolge su due schermi e diversi livelli di difficoltà.

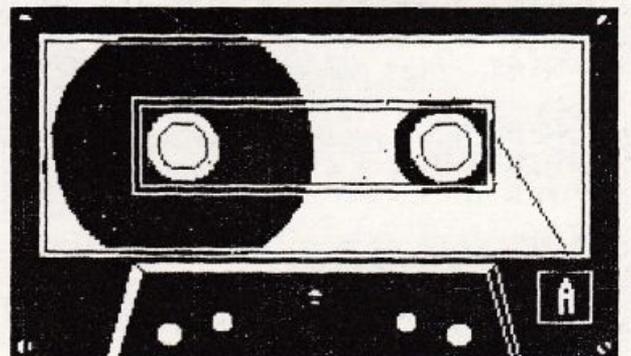
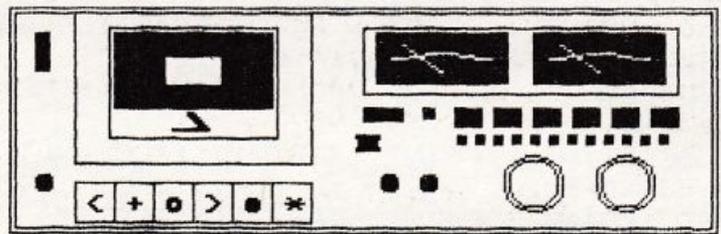
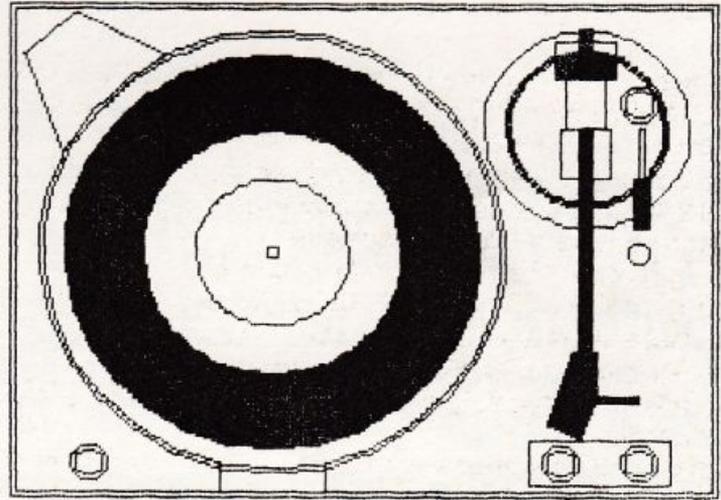
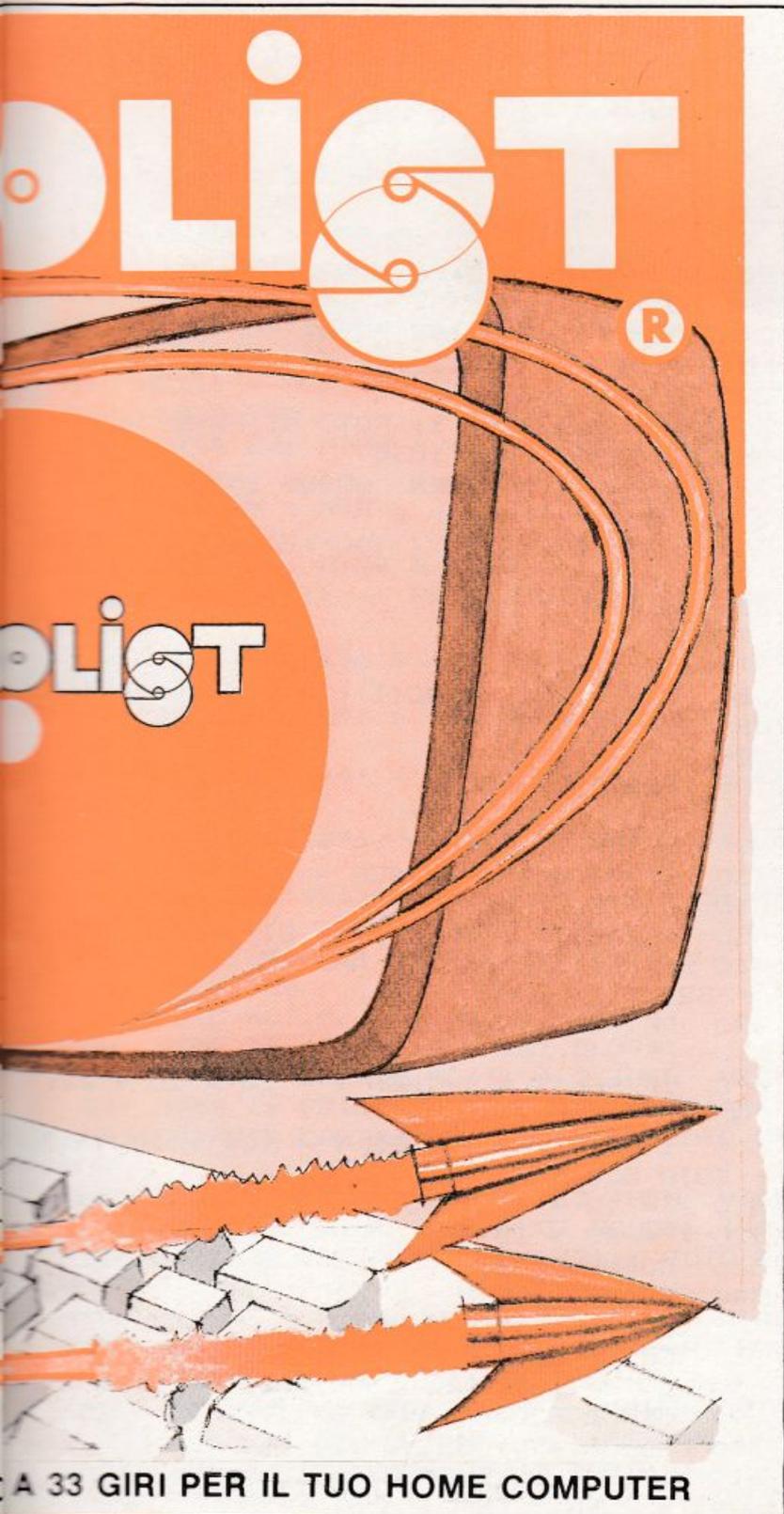
Ulteriori istruzioni sono contenute nel programma per i diversi modelli di computer.

- 1^a PARTE - VIC-20
- 2^a PARTE - TEXAS TI 99/4A
- 3^a PARTE - ZX SPECTRUM





COOLIST[®]





Black Jack

Un semplice programma per gli utenti del DIGITEK MPF II.

```
ILDAD BLACK
ILIST
```

```
10 HOME : VTAB 4
20 HTAB 5: INVERSE : PRINT "=====
=====": NORMAL
25 PRINT : PRINT TAB( 10);"B L A C K
J A C K"
30 PRINT : INVERSE : HTAB 5: PRINT "===
=====": NORMAL
50 POKE 800,32: POKE 801,67: POKE 802,2
40: POKE 803,141: POKE 804,39: POKE 805,
3: POKE 806,96
60 GOSUB 3200
70 POKE 232,145: POKE 233,29
100 HOME : VTAB 3: HTAB 15: INVERSE : P
RINT "BLACKJACK": NORMAL
105 GOSUB 3050
110 PRINT : PRINT TAB( 10)"MICROPROFES
SOR r^ MPF-II"
120 PRINT : PRINT : PRINT "DIGITEK COMP
UTER - BAGNOLO IN PIANO(RE)"
130 FOR I = 1 TO 39: PRINT "rq";: NEXT

135 PRINT : PRINT TAB( 12);"COMANDI PE
R GIOCARE"
140 PRINT : PRINT TAB( 14)"RESTO"
150 PRINT TAB( 14)"CARTA"
160 PRINT TAB( 14)"LASCIO (1/2)"
170 PRINT TAB( 14)"DOPPIO"
180 PRINT TAB( 14)"SPEZZO (SPLIT)"
190 PRINT TAB( 14)"SALVARSI (1/2)"
200 PRINT : PRINT "LE PUNTATE SARANNO C
OMPRESSE FRA $2...$50"
210 PRINT "PER SELEZIONARE I COMANDI US
A LE INIZIALI . * <RETURN>
PER CONFERMA"
220 PRINT
230 HTAB 6: PRINT "NUMERO DI GIOCATORI
(1-5) = ";: INPUT "":N#
240 N = VAL (N#):DN = N + 1
250 IF N < 1 OR N > 5 THEN 230
260 DIM V(12,12),D(207),S(12),NC(12),B(
12),BX(12)
```

```
270 BL39# = "
"
```

```
280 BL = 4:WH = 7:RD = 5
281 BU = 6
290 HOME : VTAB 10: PRINT " UNA MESCO
LATA ALLE CARTE....."
295 GOSUB 2500
300 HGR : SCALE= 1: ROT= 0: POKE 34,20:
GOSUB 2350: GOTO 1030
320 IF M > 206 THEN GOSUB 2560
330 M = M + 1:NUM = D(M)
340 SU = INT (NUM / 100):VA = NUM - 100
* SU
350 RETURN : REM

370 S = 0:S1 = 0:S2 = 0:ACE = 0
380 FOR J = 1 TO NC(I):VA = V(I,J): GOS
UB 420: NEXT
390 S(I) = S
400 RETURN : REM

420 IF VA > 10 THEN VA = 10
430 IF VA = 1 THEN ACE = 1
440 S1 = S1 + VA
450 IF ACE = 1 THEN S2 = S1 + 10
460 S = S1: IF S2 > 0 AND S2 < 22 THEN S
= S2
470 RETURN : REM

490 IF SPLIT = 1 THEN Y = 20 + 10 * JJ:
GOTO 520
500 IF SPLIT = 2 THEN Y = 100 + 10 * JJ
: GOTO 520
510 Y = 12 * (J + JJ)
520 X = 40 * ID
530 GOSUB 2300
540 RETURN : REM

560 GOSUB 860
570 ON OPT GOTO 830,640,1660
590 GOSUB 320:BX(I) = BX(I) * 2
600 GOSUB 490: GOSUB 720
610 IF S < 22 THEN 830
620 RETURN : REM
```

```

640 GOSUB 320
650 IF SPLIT = 1 THEN Y = 20 + 100 * JJ
: GOTO 680
660 IF SPLIT = 2 THEN Y = 100 + 10 * JJ
: GOTO 680
670 Y = 12 * (J + JJ)
680 X = 40 * ID: IF SPLIT < > 0 THEN JJ
= JJ + 1
690 GOSUB 2300: GOSUB 720
700 IF S > 21 THEN RETURN : REM

710 GOTO 560
720 NC(I) = NC(I) + 1
730 V(I,NC(I)) = VA
740 GOSUB 370
750 IF S < 22 THEN 780
760 VTAB 22: HTAB 6 * ID: PRINT "FUORI"

762 IF I = DN THEN SD = 5: GOSUB 3000:
GOTO 770
765 SD = 1: GOSUB 3000
770 GOSUB 790
780 RETURN : REM

790 IF NC(I) < > 0 THEN 810
800 RETURN : REM

810 NC(I) = NC(I) - 1
820 GOTO 790
830 VTAB 22: HTAB 6 * ID: PRINT " ";S(I
);"
"
840 RETURN : REM

860 VTAB 22: HTAB 6 * ID
865 CALL 800:K = PEEK (807):K$ = CHR$
(K)
870 IF ASC (K$) = 210 THEN OPT = 1: PR
INT "RESTO "
871 IF ASC (K$) = 195 THEN OPT = 2: PR
INT "CARTA "
872 IF ASC (K$) = 204 THEN OPT = 3: PR
INT "LASCIO"
873 IF ASC (K$) = 196 THEN OPT = 4: PR
INT "DOPPIO"
874 IF ASC (K$) = 211 THEN OPT = 5: PR
INT "SPEZZO"
940 CALL 800:K = PEEK (807): IF K < >
141 THEN 860
950 POKE 807,0
960 VTAB 24: HTAB 1: PRINT BL39$;
970 IF OPT = 3 AND NC(I) > 2 THEN VTAB
24: HTAB 1: PRINT "NON PUOI RINUNCIARE
(HAI CHIESTO CARTA)"; CHR$ (7);: GOTO 86
0
980 IF OPT = 4 AND NC(I) > 2 THEN VTAB
24: HTAB 1: PRINT "NON PUOI RADOPP. (HAI
GIA CHIESTO CARTA)"; CHR$ (7);: GOTO 86
0
990 IF OPT = 5 AND NC(I) > 2 THEN VTAB
24: HTAB 5: PRINT "SOLO COPPIE PARI PUO
I SPEZZARE"; CHR$ (7);: GOTO 860
1000 IF OPT = 5 AND SPLIT > 0 THEN VTA
B 24: HTAB 4: PRINT "NON E' CONCESSA UN'
ALTRA SPEZZATA"; CHR$ (7);: GOTO 860
1010 RETURN : REM

```

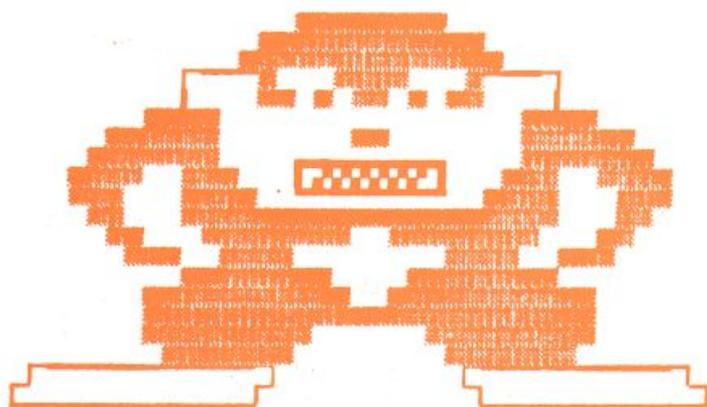
```

1030 FOR I = 1 TO DN:T(I) = 0: NEXT
1040 IF 2 * DN + M > 207 THEN GOSUB 25
60
1050 FOR I = 1 TO N:W(I) = 0: NEXT
1060 FOR I = 1 TO 12:B(I) = 0:BX(I) = 0
:S(I) = 0:NC(I) = 0: NEXT I
1070 I = I + DN
1080 VTAB 21: HTAB 1: PRINT "PUNT."
1090 VTAB 22: PRINT "SEL."
1100 VTAB 23: PRINT "TOT.=";: FOR I = 1
TO DN: HTAB 6 * I + 1: PRINT T(I);: NEX
T
1110 FOR I = 1 TO N
1120 VTAB 21: HTAB 6 * I + 1
1130 INPUT " ";A$
1135 B = VAL (A$)
1137 B(I) = B
1150 IF B < 2 OR B > 50 THEN VTAB 21:
HTAB 6 * I + 1: PRINT " "; GOTO 1120
1160 BX(I) = B(I)
1170 REM
1180 NEXT I
1190 COLOR= BU: FOR S = 0 TO 39
1192 VLIN 0,39 AT S
1194 VTAB 11: PRINT CHR$ (34);"C"; CHR
$ (34)
1195 PRINT : PRINT CHR$ (34);"R"; CHR$
(34): PRINT : PRINT CHR$ (34);"L"; CHR
$ (34): PRINT : PRINT CHR$ (34);"D"; CH
R$ (34)
1198 PRINT : PRINT CHR$ (34);"S"; CHR$
(34)
1200 NEXT
1210 FOR J = 1 TO 2: FOR I = 1 TO DN:ID
= I
1220 GOSUB 320:V(I,J) = VA:DD(I,J) = D(
M)
1230 IF J = 1 OR I < = N THEN GOSUB 4
90
1240 NEXT I: NEXT J
1250 HCOLOR= BL:X = 40 * DN + 1:Y = 24
1260 FOR JY = 0 TO 28 STEP 2: HPLLOT X,Y
+ JY TO X + 28,Y + JY: NEXT
1270 HCOLOR= RD: FOR JY = 1 TO 29 STEP
2: HPLLOT X,Y + JY TO X + 28,Y + JY: NEXT
1280 FOR I = 1 TO DN:NC(I) = 2: NEXT
1290 IF V(DN,1) > 1 THEN 1440
1310 VTAB 23: HTAB 1: PRINT BL39$;
1320 VTAB 24: HTAB 1: PRINT " TI VUOI S
ALVARE(1/2 DELLA PUNTATA) ?";
1330 VTAB 23: HTAB 1: PRINT "SALV.";
1340 FOR I = 1 TO N
1350 VTAB 23: HTAB 6 * I + 1
1360 INPUT " ";I$
1362 IF LEFT$(I$,1) = "N" THEN PRINT
"NO";:B(I) = 0: GOTO 1380
1370 PRINT "SI";:B(I) = BX(I) / 2
1380 CALL 800:K = PEEK (807): IF K <
> 141 THEN 1350
1390 REM
1400 VTAB 1: HTAB 6 * I + 1: PRINT B(I)
;" ";
1410 W(I) = B(I) * (3 * ((V(DN,2) > = 1
0)) - 1)

```



```
1420 NEXT I
1430 IF V(DN,1) = 1 AND V(DN,2) > 9 THE
N 1460
1440 IF V(DN,2) = 1 AND V(DN,1) > 9 THE
N 1460
1450 GOTO 1510
1460 X = 40 * DN:Y = 24: GOSUB 2300
1470 VTAB 24: HTAB 1: PRINT BL39#;
1480 VTAB 24: HTAB 12: INVERSE : PRINT
"FATTO BLACKJACK";: NORMAL
1485 SD = 3: GOSUB 3000: GOSUB 3000
1490 FOR I = 1 TO DN: GOSUB 370: NEXT
1500 GOTO 2090
```



```
1510 FOR I = 1 TO N:ID = I
1520 J = J + 1:JJ = 0:SPLIT = 0
1530 GOSUB 860
1540 ON OPT GOTO 1560,1700,1660,1640,17
20
1560 GOSUB 370
1570 IF S(I) < > 21 THEN 1620
1580 VTAB 22: HTAB 6 * I: INVERSE : PRI
NT "BLKJK";: NORMAL
1585 SD = 3: GOSUB 3000
1590 W(I) = W(I) + 1.5 * BX(I)
1600 BX(I) = 0
1610 GOSUB 790: GOTO 1950
1620 GOSUB 830: GOTO 1950
1640 GOSUB 370: GOSUB 590: GOTO 1950
1650 REM SURRENDER ROUTINE
```

```
1660 BX(I) = 0.5 * BX(I)
1670 S(I) = 0: IF SPLIT > 0 THEN RETURN
: REM
1680 GOTO 1950
1700 GOSUB 370: GOSUB 640: GOTO 1950
1720 IF V(I,1) = V(I,2) THEN 1760
1730 VTAB 24: HTAB 1: PRINT BL39#;
1740 VTAB 24: HTAB 8: PRINT "DIMEZZAMEN
TO NON CONCESSO!"; CHR# (7);
1750 GOTO 1530
1760 X = 40 * I:Y = 0: GOSUB 2470
1770 X = 40 * I:Y = 0: GOSUB 2470
1780 NUM = DD(I,1): GOSUB 340: GOSUB 230
0
1790 Y = 80:NUM = DD(I,2): GOSUB 340: GO
SUB 2300
1800 IN = I + DN:NC(IN) = 2:V(IN,1) = V(
I,2):BX(IN) = BX(I)
```

```
1810 Y = 10: GOSUB 320: GOSUB 2300
1820 V(I,2) = VA: GOSUB 370
1830 Y = 90: GOSUB 320: GOSUB 2300
1840 I = IN:V(I,2) = VA: GOSUB 370
1850 IF V(I,1) = 1 AND V(IN,1) = 1 THEN
1930
1860 I = IN - DN
1870 VTAB 24: HTAB 1: PRINT BL39#;
1880 VTAB 24: HTAB 15: PRINT "MANO #1";

1890 SPLIT = 1:JJ = 0: GOSUB 560
1900 I = I + DN
1910 VTAB 24: HTAB 15: INVERSE : PRINT
"MANO #2";: NORMAL
1920 SPLIT = 2:JJ = 0: GOSUB 560
1930 I = I - DN
1940 SPLIT = 0:JJ = 0
1950 NEXT I
1960 FOR I = 1 TO N
1970 IF NC(I) > 0 OR NC(I + DN) > 0 THE
N 2010
1980 NEXT I
1990 GOTO 2090
2010 I = DN:ID = DN:J = 3
2020 NUM = DD(DN,2): GOSUB 340: GOSUB 37
0
2030 X = 40 * DN:Y = 24: GOSUB 2300
2040 IF S(DN) > 16 THEN 2070
2050 GOSUB 320: GOSUB 490: GOSUB 720
2060 S(DN) = S
2070 IF S(DN) < 22 AND S(DN) < 17 THEN
2050
2080 IF S(DN) < 22 THEN VTAB 22: HTAB
6 * ID + 1: PRINT S(DN);
2090 IF S(DN) = 0 OR S(DN) > 21 THEN S(
DN) = 1
2110 VTAB 23: HTAB 1: PRINT BL39#;: VTA
B 23: HTAB 1: PRINT "TOT = ";
2120 VTAB 21: HTAB 1: PRINT BL39#;: VTA
B 23: HTAB 1: PRINT "TOTALE";
2130 FOR I = 1 TO N
2140 IF S(I) > 21 THEN S(I) = 0
2150 IF S(I + DN) > 21 THEN S(I + DN) =
0
2160 W(I) = W(I) + BX(I) * SGN (S(I) -
S(DN)) + BX(I + DN) * SGN (S(I + DN) -
S(DN))
2170 T(I) = T(I) + W(I)
2180 VTAB 21: HTAB 6 * I + 1: IF W(I) >
0 THEN INVERSE
2190 PRINT W(I);: NORMAL
2192 IF W(I) > 0 THEN SD = 2
2194 IF W(I) < 0 THEN SD = 1
2195 IF W(I) = 0 THEN SD = 4
2196 GOSUB 3000
2200 VTAB 23: HTAB 6 * I + 1: PRINT T(I
);
2210 GOSUB 790
2220 T(DN) = T(DN) - W(I)
2230 I = I + DN: GOSUB 790:I = I - DN
2240 NEXT I
2250 VTAB 23: HTAB 6 * DN + 1: PRINT T(
DN);
2260 VTAB 24: HTAB 1: PRINT "PREMI RETU
RN PER UN'ALTRA MANO";
```



```

2270 GET T#: IF ASC (T#) < > 13 THEN
2270
2280 GOSUB 790: GOSUB 2350: GOTO 1040
2300 HCOLOR= WH: FOR JY = 0 TO 29: HPLO
T X,Y + JY TO X + 29,Y + JY: NEXT
2310 HCOLOR= BL: DRAW VA AT X + 1,Y + 7
: DRAW VA AT X + 20,Y + 27: DRAW SU + 13
AT X + 6,Y + 22
2320 HCOLOR= BL: DRAW VA AT X + 2,Y + 7
: DRAW VA AT X + 21,Y + 27: DRAW SU + 13
AT X + 7,Y + 22
2330 IF SU > 2 THEN HCOLOR= RD: DRAW S
U + 13 AT X + 6,Y + 22
2340 RETURN : REM

2350 HOME : COLOR= = BU: FOR S = 0 TO
39
2354 VLIN 0,39 AT S: NEXT
2355 HTAB 12
2356 VTAB 14: INVERSE : PRINT "PUNTARE
DA 2 A 50 ": NORMAL
2360 HCOLOR= BL
2370 H PLOT 96,76 TO 90,76 TO 90,80 TO 9
6,80 TO 96,84 TO 90,84
2380 H PLOT 106,76 TO 100,76 TO 100,84 T
O 106,84
2390 H PLOT 110,84 TO 110,76 TO 116,76 T
O 116,84 TO 110,84
2400 H PLOT 120,84 TO 120,76 TO 120,79 T
O 124,76 TO 124,84 TO 124,79 TO 128,76 T
O 128,84
2410 H PLOT 132,84 TO 132,76 TO 132,79 T
O 136,76 TO 136,84 TO 136,79 TO 140,76 T
O 140,84
2420 H PLOT 150,84 TO 144,84 TO 144,80 T
O 148,80 TO 144,80 TO 144,76 TO 150,76
2435 H PLOT 160,76 TO 154,76 TO 154,80 T
O 160,80 TO 160,84 TO 154,84
2440 H PLOT 170,76 TO 164,76 TO 164,80 T
O 170,80 TO 170,84 TO 164,84
2450 H PLOT 180,76 TO 174,76 TO 174,84 T
O 180,84: H PLOT 178,80 TO 174,80
2455 H PLOT 186,76 TO 186,81: H PLOT 186,
84
2460 RETURN : REM

2470 HCOLOR= BU: FOR JY = 0 TO 54: HPLO
T X,Y + JY TO X + 29,Y + JY: NEXT
2480 RETURN : REM

2500 M = 0
2510 FOR K = 1 TO 4: FOR L = 1 TO 13: D(
M) = 100 * K + L: M = M + 1: NEXT : NEXT
2520 IF M < 208 THEN 2510
2530 FOR K = 207 TO 0 STEP - 1: M = IN
T ( RND (1) * 208): T = D(M): D(M) = D(K):
D(K) = T: NEXT
2540 M = - 1
2550 RETURN : REM

2560 VTAB 24: HTAB 1: PRINT BL39#:
2570 VTAB 24: HTAB 15: INVERSE : PRINT
"MAZZO UOVO": GOSUB 2500
2580 NORMAL : VTAB 24: HTAB 15: PRINT "
";
2590 RETURN : REM

3000 FOR J = 1 TO NN(SD)
3010 IF FQ(J,SD) > 255 THEN FQ(J,SD) =
4
3020 POKE 768,DU(J,SD): POKE 769,FQ(J,S
D): CALL 770
3030 NEXT J
3035 FOR PAUSE = 0 TO 200: NEXT PAUSE
3040 RETURN : REM
3050 FOR I = 0 TO 25: READ J: POKE 770
+ I,J: NEXT I
3060 DATA 172,1,3,174,1,3,169,4,32,168

3070 DATA 252,173,48,192,232,208,253,13
6,208,239
3080 DATA 206,0,3,208,231,96
3090 DIM FQ(9,5),DU(9,5)
3110 NN(1) = 2:NN(2) = 3:NN(3) = 8:NN(4)
= 1:NN(5) = 9
3120 FOR K = 1 TO 5: FOR I = 1 TO NN(K)
3130 READ FQ(I,K),DU(I,K)
3140 NEXT I: NEXT K
3150 DATA 144,2,80,2
3160 DATA 200,2,219,2,299,4
3170 DATA 211,1,211,1,211,1,229,2,211,1
,229,2,211,1,229,2
3180 DATA 172,4
3185 DATA 172,1,164,1,156,1,150,1,144,1
,135,1,126,1,117,1,106,2
3190 RETURN
3200 POKE 7569,18: POKE 7570,0: POKE 75
71,38: POKE 7572,0: POKE 7573,50: POKE 7
574,0: POKE 7575,65: POKE 7576,0: POKE 7
577,77:
3201 POKE 7578,0: POKE 7579,89: POKE 75
80,0: POKE 7581,101: POKE 7582,0: POKE 7
583,114: POKE 7584,0: POKE 7585,123: POK
E 7586,0:
3202 POKE 7587,136: POKE 7588,0: POKE 7
589,148: POKE 7590,0: POKE 7591,164: POK
E 7592,0: POKE 7593,173: POKE 7594,0: POK
E 7595,186:
3203 POKE 7596,0: POKE 7597,199: POKE 7
598,0: POKE 7599,50: POKE 7600,1: POKE 7
601,148: POKE 7602,1: POKE 7603,241: POK
E 7604,1:
3204 POKE 7605,95: POKE 7606,2: POKE 76
07,33: POKE 7608,36: POKE 7609,100: POKE
7610,12: POKE 7611,14: POKE 7612,14: POK
E 7613,54:
3205 POKE 7614,63: POKE 7615,119: POKE
7616,9: POKE 7617,46: POKE 7618,0: POKE
7619,41: POKE 7620,45: POKE 7621,45: POK
E 7622,216:
3206 POKE 7623,219: POKE 7624,16: POKE
7625,12: POKE 7626,12: POKE 7627,45: POK
E 7628,32: POKE 7629,28: POKE 7630,63: P
OKE 7631,30:
3207 POKE 7632,7: POKE 7633,0: POKE 763
4,1: POKE 7635,168: POKE 7636,45: POKE 7
637,5: POKE 7638,32: POKE 7639,28: POKE
7640,103:

```



3208 POKE 7641,12: POKE 7642,60: POKE 7643,63: POKE 7644,63: POKE 7645,0: POKE 7646,73: POKE 7647,33: POKE 7648,5: POKE 7649,56:

3209 POKE 7650,63: POKE 7651,39: POKE 7652,12: POKE 7653,12: POKE 7654,12: POKE 7655,54: POKE 7656,46: POKE 7657,0: POKE 7658,1:

3210 POKE 7659,112: POKE 7660,45: POKE 7661,5: POKE 7662,32: POKE 7663,228: POKE 7664,63: POKE 7665,39: POKE 7666,44: POKE 7667,45:



3211 POKE 7668,45: POKE 7669,0: POKE 7670,9: POKE 7671,45: POKE 7672,5: POKE 7673,32: POKE 7674,28: POKE 7675,63: POKE 7676,214:

3212 POKE 7677,36: POKE 7678,100: POKE 7679,12: POKE 7680,45: POKE 7681,5: POKE 7682,0: POKE 7683,33: POKE 7684,100: POKE 7685,12:

3213 POKE 7686,12: POKE 7687,228: POKE 7688,58: POKE 7689,63: POKE 7690,7: POKE 7691,0: POKE 7692,9: POKE 7693,45: POKE 7694,12:

3214 POKE 7695,228: POKE 7696,63: POKE 7697,214: POKE 7698,36: POKE 7699,32: POKE 7700,12: POKE 7701,45: POKE 7702,14: POKE 7703,54:

3215 POKE 7704,0: POKE 7705,41: POKE 7706,101: POKE 7707,12: POKE 7708,60: POKE 7709,63: POKE 7710,7: POKE 7711,32: POKE 7712,12:

3216 POKE 7713,45: POKE 7714,21: POKE 7715,46: POKE 7716,0: POKE 7717,33: POKE 7718,36: POKE 7719,36: POKE 7720,108: POKE 7721,9:

3217 POKE 7722,45: POKE 7723,14: POKE 7724,54: POKE 7725,54: POKE 7726,30: POKE 7727,63: POKE 7728,7: POKE 7729,32: POKE 7730,36:

3218 POKE 7731,36: POKE 7732,0: POKE 7733,1: POKE 7734,112: POKE 7735,45: POKE 7736,5: POKE 7737,32: POKE 7738,36: POKE 7739,36:

3219 POKE 7740,4: POKE 7741,0: POKE 7742,9: POKE 7743,109: POKE 7744,28: POKE 7745,223: POKE 7746,108: POKE 7747,13: POKE 7748,36:

3220 POKE 7749,228: POKE 7750,95: POKE 7751,191: POKE 7752,54: POKE 7753,7: POKE 7754,0: POKE 7755,33: POKE 7756,36: POKE 7757,36:

3221 POKE 7758,108: POKE 7759,9: POKE 7760,30: POKE 7761,30: POKE 7762,30: POKE 7763,14: POKE 7764,14: POKE 7765,14: POKE 7766,5:

3222 POKE 7767,0: POKE 7768,73: POKE 7769,9: POKE 7770,45: POKE 7771,45: POKE 7772,45: POKE 7773,229: POKE 7774,59: POKE 7775,63:

3223 POKE 7776,12: POKE 7777,109: POKE 7778,73: POKE 7779,56: POKE 7780,255: POKE 7781,59: POKE 7782,223: POKE 7783,63: POKE 7784,7:

3224 POKE 7785,40: POKE 7786,45: POKE 7787,109: POKE 7788,109: POKE 7789,45: POKE 7790,45: POKE 7791,5: POKE 7792,56: POKE 7793,63:

3225 POKE 7794,63: POKE 7795,63: POKE 7796,63: POKE 7797,63: POKE 7798,63: POKE 7799,63: POKE 7800,7: POKE 7801,40: POKE 7802,45:

3226 POKE 7803,45: POKE 7804,45: POKE 7805,45: POKE 7806,45: POKE 7807,45: POKE 7808,45: POKE 7809,45: POKE 7810,60: POKE 7811,63:

3227 POKE 7812,63: POKE 7813,63: POKE 7814,63: POKE 7815,63: POKE 7816,63: POKE 7817,63: POKE 7818,63: POKE 7819,44: POKE 7820,45:

3228 POKE 7821,45: POKE 7822,45: POKE 7823,45: POKE 7824,45: POKE 7825,45: POKE 7826,45: POKE 7827,45: POKE 7828,28: POKE 7829,63:

3229 POKE 7830,63: POKE 7831,63: POKE 7832,63: POKE 7833,63: POKE 7834,63: POKE 7835,63: POKE 7836,39: POKE 7837,45: POKE 7838,45:

3230 POKE 7839,45: POKE 7840,45: POKE 7841,45: POKE 7842,45: POKE 7843,45: POKE 7844,229: POKE 7845,63: POKE 7846,63: POKE 7847,63:

3231 POKE 7848,63: POKE 7849,63: POKE 7850,63: POKE 7851,103: POKE 7852,45: POKE 7853,45: POKE 7854,45: POKE 7855,45: POKE 7856,45:

3232 POKE 7857,229: POKE 7858,63: POKE 7859,63: POKE 7860,63: POKE 7861,63: POKE 7862,103: POKE 7863,45: POKE 7864,45: POKE 7865,45:

3233 POKE 7866,229: POKE 7867,63: POKE 7868,63: POKE 7869,103: POKE 7870,45: POKE 7871,229: POKE 7872,39: POKE 7873,45: POKE 7874,0:

3234 POKE 7875,73: POKE 7876,9: POKE 7877,45: POKE 7878,45: POKE 7879,45: POKE 7880,229: POKE 7881,59: POKE 7882,63: POKE 7883,12:

3235 POKE 7884,109: POKE 7885,73: POKE 7886,56: POKE 7887,255: POKE 7888,59: POKE 7889,223: POKE 7890,63: POKE 7891,7: POKE 7892,40:

3236 POKE 7893,45: POKE 7894,109: POKE 7895,109: POKE 7896,45: POKE 7897,45: POKE 7898,5: POKE 7899,56: POKE 7900,63: POKE 7901,63:

3237 POKE 7902,63: POKE 7903,63: POKE 7904,63:



904,63: POKE 7905,63: POKE 7906,63: POKE
7907,7: POKE 7908,40: POKE 7909,45: POK
E 7910,45:
3238 POKE 7911,45: POKE 7912,45: POKE 7
913,45: POKE 7914,45: POKE 7915,45: POKE
7916,45: POKE 7917,60: POKE 7918,63: PO
KE 7919,63:
3239 POKE 7920,63: POKE 7921,63: POKE 7
922,63: POKE 7923,63: POKE 7924,63: POKE
7925,63: POKE 7926,44: POKE 7927,45: PO
KE 7928,45:
3240 POKE 7929,45: POKE 7930,45: POKE 7
931,45: POKE 7932,45: POKE 7933,45: POKE
7934,45: POKE 7935,28: POKE 7936,63: PO
KE 7937,63:
3241 POKE 7938,63: POKE 7939,63: POKE 7
940,63: POKE 7941,63: POKE 7942,63: POKE
7943,103: POKE 7944,41: POKE 7945,109:
POKE 7946,45:
3242 POKE 7947,109: POKE 7948,45: POKE
7949,220: POKE 7950,27: POKE 7951,63: PO
KE 7952,255: POKE 7953,8: POKE 7954,45:
POKE 7955,45:
3243 POKE 7956,45: POKE 7957,56: POKE 7
958,63: POKE 7959,63: POKE 7960,63: POKE
7961,44: POKE 7962,45: POKE 7963,45: PO
KE 7964,45:
3244 POKE 7965,28: POKE 7966,63: POKE 7
967,63: POKE 7968,103: POKE 7969,45: POK
E 7970,229: POKE 7971,63: POKE 7972,0: P
OKE 7973,73:
3245 POKE 7974,73: POKE 7975,9: POKE 79
76,37: POKE 7977,255: POKE 7978,40: POKE
7979,45: POKE 7980,45: POKE 7981,60: PO
KE 7982,63:
3246 POKE 7983,63: POKE 7984,31: POKE 7
985,40: POKE 7986,45: POKE 7987,45: POKE
7988,45: POKE 7989,45: POKE 7990,60: PO
KE 7991,63:
3247 POKE 7992,63: POKE 7993,63: POKE 7
994,63: POKE 7995,31: POKE 7996,200: POK
E 7997,45: POKE 7998,45: POKE 7999,45: P
OKE 8000,45:
3248 POKE 8001,45: POKE 8002,45: POKE 8
003,37: POKE 8004,63: POKE 8005,63: POKE
8006,63: POKE 8007,63: POKE 8008,63: PO
KE 8009,63:
3249 POKE 8010,255: POKE 8011,40: POKE
8012,45: POKE 8013,45: POKE 8014,45: POK
E 8015,45: POKE 8016,45: POKE 8017,45: P
OKE 8018,45:
3250 POKE 8019,45: POKE 8020,60: POKE 8
021,63: POKE 8022,63: POKE 8023,63: POKE
8024,63: POKE 8025,63: POKE 8026,63: PO
KE 8027,63:
3251 POKE 8028,63: POKE 8029,76: POKE 8
030,45: POKE 8031,45: POKE 8032,45: POKE
8033,45: POKE 8034,45: POKE 8035,45: PO
KE 8036,37:
3252 POKE 8037,63: POKE 8038,63: POKE 8
039,63: POKE 8040,63: POKE 8041,63: POKE
8042,63: POKE 8043,103: POKE 8044,41: P
OKE 8045,45:
3253 POKE 8046,45: POKE 8047,45: POKE 8
048,45: POKE 8049,60: POKE 8050,63: POKE

8051,63: POKE 8052,63: POKE 8053,63: PO
KE 8054,76:
3254 POKE 8055,45: POKE 8056,45: POKE 8
057,45: POKE 8058,24: POKE 8059,63: POKE
8060,63: POKE 8061,103: POKE 8062,41: P
OKE 8063,60:
3255 POKE 8064,7: POKE 8065,0: POKE 806
6,73: POKE 8067,73: POKE 8068,9: POKE 80
69,37: POKE 8070,255: POKE 8071,40: POKE
8072,45:
3256 POKE 8073,45: POKE 8074,60: POKE 8
075,63: POKE 8076,63: POKE 8077,31: POKE
8078,40: POKE 8079,45: POKE 8080,45: PO
KE 8081,45:
3257 POKE 8082,45: POKE 8083,60: POKE 8
084,63: POKE 8085,63: POKE 8086,63: POKE
8087,63: POKE 8088,31: POKE 8089,40: PO
KE 8090,45:
3258 POKE 8091,45: POKE 8092,45: POKE 8
093,45: POKE 8094,45: POKE 8095,45: POKE
8096,60: POKE 8097,63: POKE 8098,63: PO
KE 8099,63:
3259 POKE 8100,63: POKE 8101,63: POKE 8
102,63: POKE 8103,31: POKE 8104,40: POKE
8105,45: POKE 8106,45: POKE 8107,45: PO
KE 8108,45:
3260 POKE 8109,45: POKE 8110,45: POKE 8
111,45: POKE 8112,45: POKE 8113,60: POKE
8114,63: POKE 8115,63: POKE 8116,63: PO
KE 8117,63:
3261 POKE 8118,63: POKE 8119,63: POKE 8
120,63: POKE 8121,63: POKE 8122,44: POKE
8123,45: POKE 8124,45: POKE 8125,45: PO
KE 8126,45:
3262 POKE 8127,45: POKE 8128,45: POKE 8
129,45: POKE 8130,45: POKE 8131,60: POKE
8132,63: POKE 8133,63: POKE 8134,63: PO
KE 8135,63:
3263 POKE 8136,63: POKE 8137,63: POKE 8
138,63: POKE 8139,63: POKE 8140,76: POKE
8141,45: POKE 8142,45: POKE 8143,109: P
OKE 8144,41:
3264 POKE 8145,45: POKE 8146,45: POKE 8
147,60: POKE 8148,63: POKE 8149,63: POKE
8150,223: POKE 8151,63: POKE 8152,63: P
OKE 8153,39:
3265 POKE 8154,45: POKE 8155,45: POKE 8
156,109: POKE 8157,41: POKE 8158,45: POK
E 8159,45: POKE 8160,220: POKE 8161,255:
POKE 8162,219:
3266 POKE 8163,27: POKE 8164,103: POKE
8165,43: POKE 8166,29: POKE 8167,77: POK
E 8168,73: POKE 8169,201: POKE 8170,37:
POKE 8171,255:
3267 POKE 8172,219: POKE 8173,27: POKE
8174,63: POKE 8175,0: POKE 8176,45: POKE
8177,45: POKE 8178,45: POKE 8179,45: PO
KE 8180,28:
3268 POKE 8181,63: POKE 8182,63: POKE 8
183,63: POKE 8184,12: POKE 8185,45: POKE
8186,45: POKE 8187,28: POKE 8188,63: PO
KE 8189,12:
3269 POKE 8190,5: POKE 8191,0: POKE 819
2,0
3270 RETURN

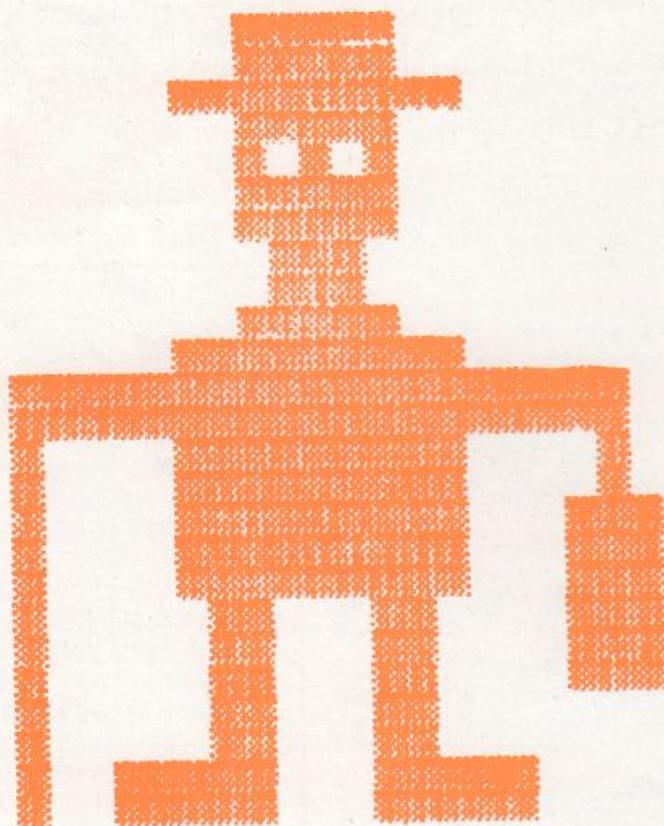


Calorie

La primavera preannuncia la prossima estate e, come al solito, il problema di molti sarà smaltire quel chilo in più che rovina tanto la linea.

Ecco LIST venirvi in aiuto con questo programma che permette di valutare

quantitativamente i consumi calorici per una serie di attività giornaliere abbastanza comuni. Conseguentemente sulle indicazioni riguardanti il consumo calorico giornaliero, non sarà difficile calibrare, con l'ausilio di adatte tavole caloriche, la dieta più opportuna.



```
1 REM # C A L O R I E #
2 REM # #
3 REM # (@) 1985 MAX. #
4 REM

5 PRINT "■":POKE36879,25
10 PRINT "■CALCOLO DEL CONSUMO CAL
ORICO"
20 PRINT "■":FORA=1TO20 PRINT "-":FORT=1
TO50
30 NEXTT,A:PRINT
40 PRINT "■IL PROGRAMMA PERMETTE DI CALC
OLARE IL CONSU-";
50 PRINT "MO ENERGETICO MEDIO, ESPRESSO
IN CALORIE, IN UN INDIVIDUO DI ETA'";
60 PRINT "MEDIA ED IN BUONE CONDIZION
I DI SALUTE, ";
70 PRINT "NEL PRATICARE ALCUNE ATTIVITA'
QUOTIDIANE";
80 PRINT "MOLTO COMUNI NELL'ARCO DI UNA GI
ORNATA. ";
90 PRINT "■■"
```

```
100 INPUT "PESO IN KG":KG
105 IFKG>130ORKG<=0THEN100
110 GOSUBS10
120 FORI=1TO20
130 READAT$,CF
140 IFAT$="END"THEN170
150 PRINT "■"CHR$(64+I)"-"AT$
160 NEXTI
170 I=I-1
190 GETA$:IFA$=""THEN190
200 IF(A$<"A"ORA$>CHR$(I+64))AND(A$<"0"
NDA$>CHR$(13))THEN190
210 IFA$<CHR$(13)THEN240
220 NX=NX+1:IFAT$="END"THENRESTORE:NX=0
230 GOTO110
240 RESTORE
250 IFA$="0"THEN500
260 FORI=1TONX*20+ASC(A$)-64
270 READAT$,CF
280 NEXTI
300 D=INT((22-LEN(AT$))/2)-1
305 PRINT "J":FORA=1TOD:D$=D$+"■":NEXTA
```

```

310 PRINTD$"/":FORA=1TOLEN(AT$):PRINT"-
":
311 FORT=1T050:NEXTT,A:PRINT"\n"
313 PRINTD$+"I"+AT$+"I"
315 PRINTD$"/":FORA=1TOLEN(AT$):PRINT"-
":
316 FORT=1T050:NEXTT,A:PRINT""
320 PRINT"QUESTA ATTIVITA'          RICHIEDE
":CF
330 PRINT"CALORIE AL MINUTO PER OGNI KG
DI PESO"
340 PRINT"PER QUANTI MIN. E'"
341 INPUT"PRATICATA":MI
350 IFMI=0THEN460
360 IFMI<0ORMI>1440THEN340
370 CN=KG*MI*CF
375 TE=TE+MI
376 FORA=0T021:PRINT"-":FORT=1T050
377 NEXTT,A
380 PRINT"CONSUMO CALORICO PER ATTIVITA
":CN
390 CA=CA+CN
400 PRINT"CONSUMO CALORICO          TOTALE:"
:CA
410 FORA=0T021:PRINT"-":FORT=1T050
411 NEXTT,A
412 PRINT"ARCO DI TEMPO:"PRINTTE;"MIN.
PARI A"
414 H=INT(TE/60):M=TE-(H*60)
415 PRINTH;"ORE E":M;"MIN."
416 TR=1440-TE:HR=INT(TR/60):MR=TR-(HR*6
0)
417 PRINT"TEMPO DA ANALIZZARE"
418 PRINTHR;"ORE E":MR;"MIN."
440 GETA$:IFA#<>CHR$(13)ANDA#<>"0"THEN44
0
450 IFA#=""THEN470
455 IFTR<1THENTR=0:GOTO470
460 MI=0:RESTORE:NX=0:GOTO110
470 PRINT"D/":FORA=1T020:PRINT"-":FORT
=1T050
471 NEXTT,A:PRINT"\n"

472 PRINT"IL CONSUMO CALORICO:"
473 PRINT"IL TOTALE AMMONTA A:"
474 PRINT"SPC(10)"CALORIE:"
475 PRINT"TRAC(4)CA
476 PRINT"SPESA INORE:"
477 PRINT"EMINUTI:"
478 PRINT"ITISPC(11)H
479 PRINT"ITIMSPC(5)M
480 PRINT"D/":FORA=1T020:PRINT"-":FORT
=1T050
490 NEXTT,A:PRINT""
500 END
510 PRINT"= ELENCO ATTIVITA' ="
515 D=0:D$=""
516 FORT=1T0900:NEXTT:PRINT"D"
520 RETURN
1000 DATABICI - DISCESA,.039

```

```

1010 DATABICI - PIANURA,.066
1020 DATABICI - SALITA,.150
1030 DATASCOLTO RADIO,.022
1040 DATABALLO SLOW,.064
1050 DATABALLO ROCK,.141

1060 DATACAMMINARE,.66
1070 DATACANTARE,.037
1080 DATACHIACCHIERARE,.024
1090 DATACORSA LENTA,.156
1100 DATACORSA VELOCE,.286
1110 DATACUCINARE,.028
1120 DATACUCIRE,.022
1130 DATAGIARDINAGGIO,.066
1140 DATADISEGNARE,.033
1150 DATADORMIRE,.015
1160 DATAED.FIS. INTENSA,.123
1170 DATAED.FIS.MODERATA,.066
1180 DATARIFARE LETTO,.068
1190 DATADOC CIA,.075
1200 DATALOTTA LIBERA,.2
1210 DATASPESA,.061
1220 DATABASKET,.103
1230 DATABOWLING,.061
1240 DATACALCIO,.139
1250 DATAGIOCO A CARTE,.024
1260 DATAPALLAVOLO,.079
1270 DATAPING-PONG,.057

1280 DATATENNIS-DOPPIO,.083
1290 DATATENNIS-SINGOLO,.101
1300 DATATELEVISIONE,.022
1310 DATAGUIDARE,.041
1320 DATALAV.PAVIMENTO,.070
1330 DATALAV.PIATTI,.033
1340 DATALAVARSI,.044
1350 DATALAVORARE MAGLIA,.022
1360 DATALAVORARE SEDUTO,.028
1370 DATALEGGERE,.024
1380 DATAMANGIARE,.024
1390 DATANUOTARE,.127
1400 DATAPASSEGGIARE,.048
1410 DATAPATTINARE,.079
1420 DATAPESCARRE,.035
1430 DATARIPOSO DISTESI,.017
1440 DATARIPOSO SEDUTI,.020
1450 DATASCALE,.132
1460 DATASCIARE,.130
1470 DATAUSO COMPUTER,.034
1480 DATASPOLVERARE,.022
1490 DATASTIRARE,.039

1500 DATASTUDIARE,.030
1510 DATASUONARE PIANO,.039
1520 DATASUONARE BATTERIA,.066
1530 DATATELEFONARE,.024
1540 DATAVESTIRSI,.066
1550 DATAEND.0,0

```

READY.



Sort 1

ELENCO VARIABILI

- TP Durata di un LOOP di ritardo FOR-NEXT
- X Posizione del primo carattere su cui si effettua l'ordinamento
- Y Posizione dell'ultimo carattere su cui si effettua l'ordinamento
- N Numero degli elementi (stringhe) da ordinare
- N\$ Stringhe
- K FLAG di scambio, ovvero variabile che cambia il suo valore da 0 a 1 se nell'esame delle stringhe c'è stata una inversione
- P Prima stringa da visionare
- U Ultima stringa da visionare

STRUTTURA PROGRAMMA

- 20- 40 Inizializzazione dei registri per effetti sonori
- 100- 180 Display presentazione
- 190- 300 Inserimento del criterio di sort (se ordinare sulla base di confronti su tutti i caratteri degli elementi o solo su parte di questi)
- 310- 380 Inserimento della posizione e del numero dei caratteri da confrontare
- 390- 410 Controllo della validità della scelta effettuata
- 420- 520 Display della scelta effettuata
- 530- 640 Inserimenti delle stringhe
- 650- 730 Esecuzione dell'ordinamento
- 740- 782 Lampeggio del segnale di fine sort
- 790- 990 Visione delle stringhe ordinate
- 1000-1020 Fine programma

Il compito che probabilmente riesce meglio ad un computer è quello di ordinare una grande quantità di dati, in funzione di una archiviazione o di una rapida elaborazione numerica. Proprio per tale motivo in Francia il computer viene definito ORDINATEUR, cioè ordinatore di dati. Le routines che si occupano dell'ordinamento degli elementi costituenti una lista vengono definite di SORT, che in inglese significa appunto catalogare, riordinare, e ne esistono di molti tipi, in relazione ai diversi algoritmi che risolvono il problema di un

veloce ed efficiente ordinamento.

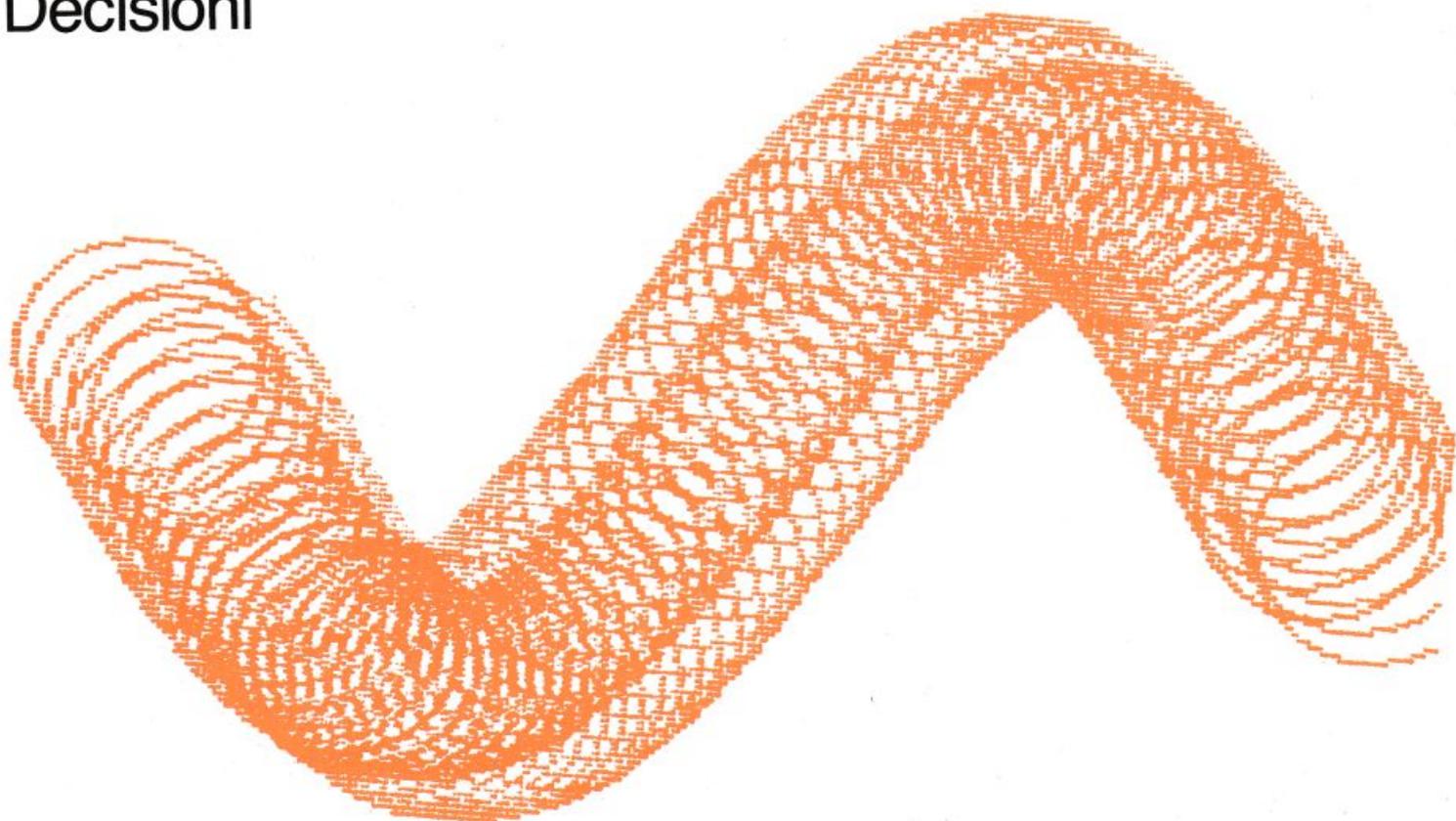
Il programma presentato sfrutta l'algoritmo di ordinamento a bolle (bubble sort), forse il più semplice. Crea una lista di stringhe, caricata mediante INPUT non ordinato — quindi casuale — la routine di bubble sort le posiziona in ordine alfabetico crescente. Gli elementi della lista vengono infatti confrontati a due a due, carattere per carattere, fino a trovare un carattere differente: l'elemento con il carattere avente codice ASCII minore viene considerato precedente l'altro. Se due elementi hanno un numero diverso di caratteri e la prima parte del più lungo è uguale all'intero altro elemento, quello più corto si considera precedente. È possibile operare il confronto su tutti i caratteri componenti le stringhe o su un numero limitato di questi. Il CBM 64 ha la facoltà di operare tali confronti in maniera molto semplice e rapida sfruttando l'istruzione MID\$(X\$,S,X).

Più in dettaglio, la routine di sort (linee 650-730) — passa — con un ciclo FOR-NEXT tutto l'elenco e confronta le coppie di stringhe adiacenti: se sono nell'ordine sbagliato, si scambiano; si ripete poi il processo fino a che tutte le coppie adiacenti siano in ordine; a questo punto la lista è ordinata. Ora è chiaro il motivo della denominazione BUBBLE SORT: ad ogni esame le stringhe che devono stare all'inizio dell'elenco si spostano verso l'alto, come una serie di bollicine d'aria che vengono a galla in un liquido. Tutto il resto del programma serve per introdurre le stringhe e visionarle una volta ordinate. Un segnale sonoro accompagna ogni scelta da effettuare da tastiera durante lo svolgimento del programma ed avvisa l'utente più distratto del termine del sort (*).

È evidente che la durata dell'ordinamento varia molto in funzione del numero degli elementi inseriti: se si vuole sfruttare la massima capacità del computer — immettendo circa 400 stringhe da 78 caratteri ciascuna — occorre munirsi di molta pazienza ed attendere parecchi minuti prima di ottenere la visualizzazione della lista ordinata. Per i lettori più esigenti verrà pubblicata in un prossimo numero, una versione di questo



Decisioni



Qualche tempo fa, con il boom del fenomeno home-computer, in molti negozi circolava un programma, di una certa efficacia, che voleva dimostrare come il computer potesse quasi «ragionare» autonomamente. Molti inesperti ed entusiasti acquirenti di allora, probabilmente si fecero affascinare un pochino troppo da quel programma, pensando quasi che il computer fosse un magico «marchingegno» a cui poter delegare le decisioni più difficili. Constatata l'amara realtà, molti hanno finito per accantonare questa «stupida» e strana macchinetta considerata purtroppo da molti come la panacea dei problemi che affliggono l'uomo della strada. Un peccato sottovalutare quel programma e con esso il computer.

Si tratta invece di un modo molto scientifico di accostarsi ai procedimenti logico-mentali che tutti noi compiamo giornalmente quando prendiamo una decisione, importante o no che sia. Il maggior pregio della «computermania» diffusasi in modo incontrollato nel nostro paese, avrebbe dovuto essere quello di comprendere ed assimilare quei processi logici che tutti i programmatori dovrebbero avere; per capirci, un po' come

succedeva con il latino o la matematica. Così purtroppo non è stato: sono molti coloro che fruiscono passivamente del software in commercio ormai anche dai fruttivendoli e non si chiedono nemmeno come possa funzionare e su quali basi sia costruito. Il programma presentato in queste pagine permette di prendere delle decisioni sulle basi più scientifiche possibili. Il procedimento, che si basa sul numero e sulla qualità dei fattori che intervengono nella decisione, sul loro ordine di importanza, sulla valutazione singola e totale rispetto alle varie alternative, alla fine fornisce un elenco ordinato delle percentuali assegnate alle varie alternative. Non pensiate però che tutto avvenga in modo automatico, in realtà durante l'esecuzione del programma è indispensabile fornire il proprio contributo indicando ad esempio l'ordine di importanza dei fattori o «aiutando» il computer a confrontare le varie alternative esprimendo le proprie valutazioni per ogni fattore. Al computer è destinata forse la parte più noiosa, quella dei calcoli, all'uomo come al solito in questi casi, quella più importante, quella di guida della macchina.



```

10 CLS:COLOR4,7
20 CURSOR0,12:PRINT"DECISION MAKER"
30 FOR T=1 TO 900:NEXT T
40 CLS:COLOR2,1
50 DIM L$(70),F$(20)
60 DIM U(20),C(20,20)
70 DIM D(20),Z(20)
80 E$="--":PRINT"~~~~~
~~~~~"

```

```

90 PRINT"1) SCELTA DI UNA TRA DIVERSE
ALTERNATIVE.":PRINT
100 PRINT"2) SCELTA DI UNA SERIE DI AZI
IONI TRA DIVERSE ALTERNATIVE.":PRI
NT

```

```

110 CURSOR3,8:INPUT "COSA SCEGLI ";R$
120 T=VAL(R$)
130 ON T GOTO 140,180
140 CURSOR3,10:PRINT"SU CHE TIPO DI QU
ESTIONE"
150 CURSOR3,11
160 INPUT "DEVI DECIDERE ";T$
170 GOTO 190
180 T$="SERIE DI AZIONI"
190 NI=0

```

```

200 PRINT:PRINT"OCCORRE UNA LISTA DI O
GNI ";T$
210 PRINT"IN ESAME. RISPONDI AD OGNI P
UNTO DI"
220 PRINT"DOMANDA E BATTI <";E$;"> ALL
A FINE."
230 PRINT
240 NI=NI+1
250 INPUT L$(NI)
260 IF L$(NI)<>E$ THEN 240
270 NI=NI-1
280 IF NI>=2 THEN 310
290 PRINT:PRINT"DEVI FARE ALMENO 2 SCE
LTE... RIPROVA!":BEEP
300 GOTO 190

```

```

310 PRINT:PRINT"QUESTA E' LA TUA LISTA
":PRINT
320 FOR J=1 TO NI
330 PRINT STR$(J);" ";L$(J)
340 NEXT J

```

```

350 PRINT:PRINT"ORA PENSA AI FATTORI C
HE CONSIDERI PIU' IMPORTANTI PER SC
EGLIERE IL/LA"
360 PRINT"MIGLIORE ";T$
370 PRINT
380 PRINT"BATTI <";E$;"> ALLA FINE."
390 NF=0
400 NF=NF+1
410 INPUT F$(NF)
420 IF F$(NF)<>E$ THEN 400
430 NF=NF-1
440 PRINT
450 IF NF>=2 THEN 490
460 PRINT:PRINT"DEVI FARE ALMENO 2 SCE
LTE... RIPROVA!"
470 NF=0
480 GOTO 400
490 CLS
500 COLOR2,1

```

```

510 PRINT"LISTA DEI FATTORI:":PRINT
520 FOR J=1 TO NF
530 PRINT STR$(J);" ";F$(J)

```

```

540 NEXT J
550 PRINT:PRINT"DECIDI QUAL'E' IL FATT
ORE PIU' IMPOR- TANTE ED INTRODUCI IL
SUO NUMERO."
560 PRINT:PRINT"USA 0 PER CORREGGERE."
:PRINT
570 INPUT A
580 IF A=0 THEN 350
590 IF A>0 AND A<(NF-1) THEN 620
600 PRINT
610 GOTO 570
620 IF NF=1 THEN 780
630 CLS
640 PRINT"SCALA DI IMPORTANZA DA 0 A 1
0.":PRINT
650 PRINT"DAREMO A ";F$(A);" UN VALORE
"

```

```

660 PRINT"PARI A 10 PERCHE' ";F$(A)
670 PRINT"E' PIU' IMPORTANTE DEGLI ALT
RI FATTO- RI."
680 PRINT"CHE VALORE HANNO GLI ALTRI F
ATTORI?":PRINT
690 FOR J=1 TO NF
700 Q=A
710 IF J=Q THEN 770
720 PRINT " ";F$(J);" ";
730 INPUT U(J)
740 IF U(J)>=0 AND U(J)<11 THEN 770
750 PRINT:PRINT"VALORE IMPOSSIBILE...R
IPROVA!"
760 GOTO 720
770 NEXT J
780 U(A)=3
790 Q=0
800 FOR J=1 TO NF
810 Q=Q+U(J)
820 NEXT J
830 FOR J=1 TO NF
840 U(J)=U(J)/Q
850 NEXT J
860 CLS

```

```

870 PRINT"ORA CONFRONTIAMO OGNI ";T$
880 PRINT"SECONDO UN LIVELLO D'IMPORTA
NZA E"
890 PRINT"CONSIDERANDO I FATTORI SINGO
LARMENTE."
900 PRINT"DOPO GIUDICHEREMO OGNI ";T$.
910 PRINT"NEI TERMINI DI QUEL SOLO FAT
TORE."
920 PRINT"DIAMO A ";L$(1);" UN VALORE"

```

```

930 PRINT"DI 10 SU TUTTE LE SCALE."
940 PRINT"AD OGNI ALTRO/A ";T$
950 PRINT"SARA' ASSEGNATO UN VALORE MI
NORE O MAGGIORE DI 10 IN BASE AL FA
TTO CHE"
960 PRINT"TU PRESUPPONGA SIA MIGLIORE/
PEGGIORE."
970 PRINT"DI ";L$(1)
980 PRINT:PRINT:PRINT"PREMI UN TASTO P
ER CONTINUARE."
990 X$=INKEY$:IF X$="" THEN 990
1000 FOR J=1 TO NF
1010 CLS
1020 PRINT"CONSIDERANDO SOLO ";F$(J)
1030 PRINT"ASSEGNIAMO 10 A ";L$(1)
1040 PRINT:PRINT"QUAL'E' IL VALORE PER
..."

```

```

1050 FOR K=2 TO NI
1060 PRINT L$(K);
1070 INPUT C(K,J)
1080 IF C(K,J)>=0 THEN 1110
1090 PRINT:PRINT"VALORE NEGATIVO! NON
VALIDO...":BEEP
1100 GOTO 1060
1110 NEXT K
1120 C(1,J)=10
1130 NEXT J

```

```

1140 CLS:PRINT"TRA POCO TI DARO' L'ARD
UA SENTENZA!:"
1150 FOR J=1 TO NF
1160 Q=0
1170 FOR K=1 TO NI
1180 Q=Q+C(K,J)
1190 NEXT K
1200 FOR K=1 TO NI
1210 C(K,J)=C(K,J)/Q
1220 NEXT K,J
1230 FOR K=1 TO NI
1240 D(K)=0
1250 FOR J=1 TO NF
1260 D(K)=D(K)+C(K,J)*U(J)
1270 NEXT J,K
1280 MX=0
1290 FOR K=1 TO NI
1300 IF D(K)<=MX THEN 1320
1310 MX=D(K)
1320 NEXT K
1330 FOR K=1 TO NI
1340 D(K)=D(K)*100/MX
1350 NEXT K
1360 FOR K=1 TO NI
1370 Z(K)=K
1380 NEXT K
1390 NM=NI-1
1400 FOR K=1 TO NI
1410 FOR J=1 TO NM
1420 N1=Z(J)
1430 N2=Z(J+1)
1440 IF D(N1)>D(N2) THEN 1470
1450 Z(J+1)=N1
1460 Z(J)=N2
1470 NEXT J,K
1480 CLS
1490 PRINT"-----
-----"

```

```

1500 PRINT"^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
^^^^^^^^^^^^^^^^"
1510 PRINT"-----
-----"
1520 PRINT:PRINT"ECCO NELL'ORDINE LA L
ISTA OTTIMALE CHE HO RICAUATO:"
1530 PRINT
1540 PRINT"-----
-----"
1550 PRINT:FOR J=1 TO NI
1560 Q=Z(J)
1570 PRINTINT(D(Q));" ";L$(Q)
1580 NEXT J
1590 PRINT
1600 PRINT"-----
-----"
1610 PRINT"^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
^^^^^^^^^^^^^^^^"
1620 PRINT"-----
-----"

```

```

1630 END

```




Gioco dei fiori

Un gioco adatto ai più piccini atto a spronare le loro capacità di osservazione.

GESTIONE DEL PROGRAMMA

Premere ogni volta il tasto P per far partire le 5 serie di fiori dai diversi colori, osservarne attentamente il numero (ogni riga ne contiene al massimo 7) e digitarlo: le cifre si apporranno automaticamente a partire dalla serie celeste.

Un accordo musicale sottolineerà la risposta esatta, facendo salire il punteggio; se invece la risposta sarà errata (anche una sola), sarà

visualizzata l'esatta serie numerica. Il gioco finisce dopo 20 schermi.

VARIABILI PRINCIPALI

QUE Conteggio quesiti
RIS Conteggio risposte
LIV Livello di difficoltà
LI Durata apparizione
CE RANDOM Numero fiori celesti
GI RANDOM Numero fiori gialli
VI RANDOM Numero fiori viola
RO RANDOM Numero fiori rossi
BL RANDOM Numero fiori bleu

```
10 CALL CLEAR
20 CALL SCREEN(12)
30 RR=3
40 QUE=0
50 RIS=0
60 A$="QUESITI RISPOSTE"
70 PRINT " GIOCO DEI FIORI (d
edicato a daniela)
DADO 1984"::
```

```
80 INPUT "LIVELLO FACILE PREMI 1 LI
VELLO MEDIO PREMI 2 LIVELLO SUPER
PREMI 3 ":LIV
90 IF LIV=1 THEN 120
100 IF LIV=2 THEN 140
110 IF LIV=3 THEN 160 ELSE 80
120 LI=600
130 GOTO 170
140 LI=200
150 GOTO 170
160 LI=30
170 CALL CLEAR
180 CALL SCREEN(16)
190 CALL CHAR(40,"00070F0F0F07327D")
200 CALL CHAR(96,"00070F0F0F07327D")
210 CALL CHAR(104,"00070F0F0F07327D")
220 CALL CHAR(112,"00070F0F0F07327D")
230 CALL CHAR(120,"00070F0F0F07327D")
240 CALL CHAR(41,"00C0E0E0E0C06CBE")
```

```
250 CALL CHAR(97,"00C0E0E0E0C06CBE")
260 CALL CHAR(105,"00C0E0E0E0C06CBE")
270 CALL CHAR(113,"00C0E0E0E0C06CBE")
280 CALL CHAR(121,"00C0E0E0E0C06CBE")
290 CALL CHAR(42,"7B7D7E3307070703")
300 CALL CHAR(98,"7B7D7E3307070703")
310 CALL CHAR(106,"7B7D7E3307070703")
320 CALL CHAR(114,"7B7D7E3307070703")
330 CALL CHAR(122,"7B7D7E3307070703")
340 CALL CHAR(43,"DEBE7ECCE0E0E0C0")
350 CALL CHAR(99,"DEBE7ECCE0E0E0C0")
360 CALL CHAR(107,"DEBE7ECCE0E0E0C0")
370 CALL CHAR(115,"DEBE7ECCE0E0E0C0")
380 CALL CHAR(123,"DEBE7ECCE0E0E0C0")
390 CALL CHAR(128,"0371793D1F0F0101")
400 CALL CHAR(129,"800E3C78F0E0")
410 FOR A=1 TO LEN(A$)
420 B=ASC(SEG$(A$,A,1))
430 CALL HCHAR(21,6+A,B)
440 NEXT A
450 FOR A=3 TO 19 STEP 4
460 CALL HCHAR(A,31,63)
470 NEXT A
480 CALL HCHAR(3,30,43)
490 CALL HCHAR(7,30,99)
500 CALL HCHAR(11,30,107)
510 CALL HCHAR(15,30,115)
520 CALL HCHAR(19,30,123)
530 CALL HCHAR(23,9,48,2)
540 CALL HCHAR(23,20,48,2)
550 CALL KEY(0,K,S)
560 IF S=0 THEN 550
570 IF K=80 THEN 590
580 GOTO 550
```

```
590 CALL SOUND(100,880,10)
600 FOR A=3 TO 19 STEP 4
610 CALL HCHAR(A,31,63)
620 NEXT A
630 FOR A=2 TO 18 STEP 4
640 CALL HCHAR(A,30,32)
650 NEXT A
660 CALL COLOR(2,16,16)
670 CALL COLOR(9,16,16)
680 CALL COLOR(10,16,16)
690 CALL COLOR(11,16,16)
700 CALL COLOR(12,16,16)
710 CALL COLOR(13,16,16)
720 RANDOMIZE
730 CE=INT(7*RND+1)
740 RANDOMIZE
750 GI=INT(7*RND+1)
760 RANDOMIZE
770 VI=INT(7*RND+1)
780 RANDOMIZE
790 RO=INT(7*RND+1)
800 RANDOMIZE
810 BL=INT(7*RND+1)
820 FOR D=3 TO (CE*4)-1 STEP 4
830 CALL HCHAR(1,D,40)
840 CALL HCHAR(1,D+1,41)
850 CALL HCHAR(2,D,42)
860 CALL HCHAR(2,D+1,43)
870 CALL HCHAR(3,D,128)
880 CALL HCHAR(3,D+1,129)
890 NEXT D
900 FOR D=3 TO (GI*4)-1 STEP 4
910 CALL HCHAR(5,D,96)
920 CALL HCHAR(5,D+1,97)
```



```

930 CALL HCHAR(6,D,98)
940 CALL HCHAR(6,D+1,99)
950 CALL HCHAR(7,D,128)
960 CALL HCHAR(7,D+1,129)
970 NEXT D
980 FOR D=3 TO (VI*4)-1 STEP 4
990 CALL HCHAR(9,D,104)
1000 CALL HCHAR(9,D+1,105)
1010 CALL HCHAR(10,D,106)
1020 CALL HCHAR(10,D+1,107)
1030 CALL HCHAR(11,D,128)
1040 CALL HCHAR(11,D+1,129)
1050 NEXT D

1060 FOR D=3 TO (RO*4)-1 STEP 4
1070 CALL HCHAR(13,D,112)
1080 CALL HCHAR(13,D+1,113)
1090 CALL HCHAR(14,D,114)
1100 CALL HCHAR(14,D+1,115)
1110 CALL HCHAR(15,D,128)
1120 CALL HCHAR(15,D+1,129)
1130 NEXT D
1140 FOR D=3 TO (BL*4)-1 STEP 4
1150 CALL HCHAR(17,D,120)
1160 CALL HCHAR(17,D+1,121)
1170 CALL HCHAR(18,D,122)
1180 CALL HCHAR(18,D+1,123)
1190 CALL HCHAR(19,D,128)
1200 CALL HCHAR(19,D+1,129)
1210 NEXT D
1220 CALL COLOR(2,8,16)
1230 CALL COLOR(9,11,16)
1240 CALL COLOR(10,14,16)
1250 CALL COLOR(11,10,16)

1260 CALL COLOR(12,5,16)
1270 CALL COLOR(13,13,16)
1280 FOR TE=1 TO LI
1290 NEXT TE
1300 FOR D=1 TO 19
1310 CALL HCHAR(D,1,32,28)
1320 NEXT D
1330 CALL KEY(0,K,S)
1340 IF S=0 THEN 1330
1350 IF S=-1 THEN 1330
1360 IF (K>48)*(K<56) THEN 1380
1370 GOTO 1330
1380 CALL HCHAR(RR,31,K)
1390 CALL SOUND(100,K*10,10)
1400 RR=RR+4
1410 IF RR>19 THEN 1420 ELSE 1330
1420 RR=3
1430 CALL GCHAR(3,31,M)
1440 MM=ASC(STR$(CE))
1450 IF M=MM THEN 1460 ELSE 1660
1460 CALL GCHAR(7,31,N)
1470 NN=ASC(STR$(GI))
1480 IF N=NN THEN 1490 ELSE 1660
1490 CALL GCHAR(11,31,D)
1500 DD=ASC(STR$(VI))
1510 IF D=DD THEN 1520 ELSE 1660
1520 CALL GCHAR(15,31,P)
1530 PP=ASC(STR$(RO))
1540 IF P=PP THEN 1550 ELSE 1660
1550 CALL GCHAR(19,31,Q)
1560 QQ=ASC(STR$(BL))
1570 IF Q=QQ THEN 1580 ELSE 1660
1580 CALL SOUND(300,831,10,523,10,622,10)
)

1590 RIS=RIS+1
1600 RI$=STR$(RIS)
1610 FOR D=1 TO LEN(RI$)
1620 L=ASC(SEG$(RI$,D,1))
1630 CALL HCHAR(23,(21+D)-LEN(RI$),L)
1640 NEXT D
1650 GOTO 1770
1660 CALL SOUND(200,-7,10)
1670 MM=ASC(STR$(CE))
1680 NN=ASC(STR$(GI))
1690 OO=ASC(STR$(VI))
1700 PP=ASC(STR$(RO))
1710 QQ=ASC(STR$(BL))
1720 CALL HCHAR(2,30,MM)
1730 CALL HCHAR(6,30,NN)
1740 CALL HCHAR(10,30,OO)
1750 CALL HCHAR(14,30,PP)
1760 CALL HCHAR(18,30,QQ)
1770 QUE=QUE+1
1780 QU$=STR$(QUE)
1790 FOR D=1 TO LEN(QU$)
1800 U=ASC(SEG$(QU$,D,1))
1810 CALL HCHAR(23,(10+D)-LEN(QU$),U)
1820 NEXT D
1830 IF QUE=20 THEN 1840 ELSE 550
1840 PRINT "FINE GIOCO
PREMI G PER GIOCARE          D F PER CONC
LUDERE"
1850 CALL KEY(0,K,S)
1860 IF S=0 THEN 1850
1870 IF K=71 THEN 10
1880 IF K=70 THEN 1900
1890 GOTO 1850
1900 END

```



Flush

Una invasione di formiche è in atto!
A vostra disposizione per cercare di bloccarla
non insetticidi, ma acqua.
Istruzioni complete contenute nel listato.

Attenzione! Digitate le lettere tra virgolette
dopo essere entrati nel modo grafico
(Graphics).

```
1 REM **          F L U S H !          **
2 REM **
3 REM **
4 REM **      ZX - SPECTRUM          **
5 REM **
6 REM **      16K / 48K              **
7 REM **
8 REM **      L I S T 1 9 8 5      **
9 REM **
10 REM
12 BORDER 7: PAPER 7: CLS
15 POKE 23658,8
18 GO SUB 8000
20 LET rec1=0: LET rec2=0: LET
rec3=0
25 LET n$="BOB": LET d$="SUSY"
: LET u$="JIM"
30 BORDER 5: FOR k=0 TO 24: LE
T J=USR 3582: BEEP .009,k: NEXT
k
40 PRINT AT 12,5;"ISTRUZIONI ?
(S/N)"
50 INPUT r$
60 IF r$="S" THEN GO TO 70
65 IF r$="N" THEN GO TO 100
68 GO TO 50
70 BORDER 7: FOR k=0 TO 24: LE
T J=USR 3582: PAUSE 1: NEXT k
80 LET i$="  SCOPO DEL GIOCO E
' DI EVITARE CHE DELLE FORMICHE
RIESCANO A SCAPPARE 'INNAFFIAN
DOLE' CON UN GETTO DI ACQUA CONT
ROLLATO DA VOI MEDIANTE UN RUB
INETTO.          TRE 'FUGGITIVE' A
MMESSE."
81 GO SUB 7100: PAUSE 100: CLS
: LET i$="  RAGGIUNTO IL 'TETT
O' DEI 200 PUNTI AUMENTANO,PER U
OI,LE DIF- FICOLTA'."
89 GO SUB 7100: PAUSE 100: CLS
: LET i$="COMANDI = 1 SINISTRA
          2 DESTRA
          0 'ACQUA'"
90 GO SUB 7100: PAUSE 50: PRIN

T "PREMERE UN TASTO PER INIZIARE
": PAUSE 0
100 CLS : LET pu=0: LET vite=3:
LET q=10: LET a$="AB
          CD
          EF
          GHI
          IKL  ": LET
b$=" MN
OP ": LET c$="0"
110 INPUT AT 12,0;"LIVELLO [1=d
iff. / 3=facile]";ld
120 IF ld<1 OR ld>3 THEN GO TO
110
130 IF ld=1 THEN LET d=.1
140 IF ld=2 THEN LET d=.5
150 IF ld=3 THEN LET d=.9
160 PAPER 5: BORDER 5: CLS
170 PRINT AT 0,q;a$
190 FOR i=19 TO 6 STEP -1
200 LET e=INT (RND*30)

210 PRINT AT i,e;b$
218 PLOT 0,0: DRAW 255,0: DRAW
0,135: PLOT 0,0: DRAW 0,135: DRA
W 255,0
220 BEEP d,i
230 IF INKEY$="1" AND q>0 THEN
GO SUB 4000
240 IF INKEY$="2" AND q<29 THEN
GO SUB 5000
250 IF INKEY$="0" THEN GO TO 60
00
260 PRINT AT i,e;"  ": PRINT A
T i+1,e;"  "
270 NEXT i
300 LET vite=vite-1: IF vite=0
THEN GO TO 6500
305 BORDER 7: PAPER 7: CLS : PR
INT INK 9;"ATTENTO! UNA FORMICA
E'FUGGITA!""VITE ";vite'"PUNTE
GGIO ";pu: PRINT "PREMERE 'C' PE
R CONTINUARE"
306 PAUSE 0
```



```

310 IF INKEY$="C" THEN GO TO 160
315 GO TO 305
4000 FOR h=0 TO 5: PRINT AT h,q;
" "; NEXT h: LET q=q-1: PRINT
AT 0,q;a$: PLOT 0,135: DRAW 255
,0: RETURN
5000 FOR h=0 TO 5: PRINT AT h,q;
" "; NEXT h: LET q=q+1: PRINT
AT 0,q;a$: PLOT 0,135: DRAW 255
,0: RETURN
6000 FOR f=6 TO 21: PRINT AT f,q
+3;c$: BEEP .01,f: PRINT AT f,q+
3;" ": NEXT f: PRINT AT 21,q+3;"
"
6010 IF q+3=e OR q+3=e+1 OR q+3=
e+2 THEN GO TO 6030
6020 GO TO 260
6030 FOR x=1 TO 5: PRINT AT i,e;
INK x;b$: BEEP .5,x: PRINT AT i
,e;" ": NEXT x: LET pu=pu+i*2:
GO TO 160
6040 IF pu<200 OR pu>=260 THEN G
O TO 160
6050 FOR l=2 TO 25: PRINT AT 10,
l;"BONUS": BEEP .010,l: PRINT AT
10,l;" ": NEXT l
6055 LET n=0
6060 FOR r=2 TO 27: PRINT AT 19,
r;b$: BEEP .009,r: IF INKEY$="0"
THEN GO TO 6200: PRINT AT 19,r;
" ": PRINT AT 20,r;" "
6061 NEXT r: PRINT AT 19,27;"
": PRINT AT 20,27;" "
6062 LET n=n+1: IF n<5 THEN GO T
O 6060
6100 GO TO 160
6200 FOR z=6 TO 20: PRINT AT z,q
+3;c$: BEEP .01,z: PRINT AT z,q+
3;" ": NEXT z: PRINT AT 20,q+3;"
"
6210 IF q+3=r OR q+3=r+1 OR q+3=
r+2 THEN GO TO 6230
6220 PRINT AT 19,r;" ": PRINT
AT 20,r;" ": LET n=n+1: IF n<5
THEN GO TO 6060: GO TO 160
6230 FOR x=1 TO 5: PRINT AT 19,r
;b$: BEEP .5,x: PRINT AT
19,r;" ": NEXT x: LET pu=pu+50
6240 LET n=n+1: IF n<5 THEN GO T
O 6060: GO TO 160
6500 CLS : FOR f=1 TO 2: BEEP 0.
6,-24: BEEP 0.4,-24: BEEP 0.2,-2
4: BEEP 0.6,-24: BEEP 0.4,-21: B
EEP 0.2,-22: BEEP 0.4,-22: BEEP
0.2,-24: BEEP 0.4,-24: BEEP 0.2,
-25: BEEP 0.6,-24: NEXT f
6510 PRINT ""PUNTEGGIO FINALE "
:pu
6520 IF ld=1 AND pu>=rec1 THEN L
ET rec1=pu: PRINT FLASH 1;"NUOU
O RECORD": INPUT "NOME PREGO ";
n$: GO TO 7000
6530 IF ld=2 AND pu>=rec2 THEN L
ET rec2=pu: PRINT FLASH 1;"NUOU
O RECORD": INPUT "NOME PREGO ";
d$: GO TO 7000
6540 IF ld=3 AND pu>=rec3 THEN L

```

```

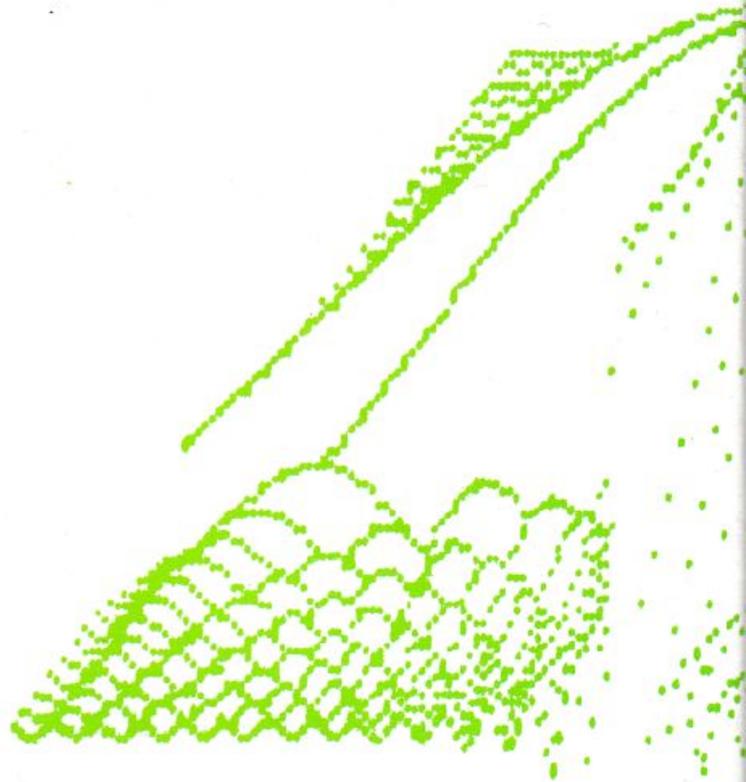
ET rec3=pu: PRINT FLASH 1;"NUOU
O RECORD": INPUT "NOME PREGO ";
u$: GO TO 7000
6550 PAUSE 100
7000 CLS : BORDER 7: PAPER 7: CL
S: PRINT "000000000000000000000000
000000000000": FOR i=1 TO 20: PRI
NT "0";TAB 31;"0": NEXT i: PRINT
"00000000000000000000000000000000
00"
7010 PRINT AT 5,6; INK 0; FLASH
1;"THE CHAMPIONS": PRINT : PRINT
7020 PRINT INK 1;AT 7,3;"CATEGOR
IA 1 ": PRINT AT 8,3;" ";rec1
;" ";n$
7030 PRINT INK 2;AT 10,3;"CATEGO
RIA 2": PRINT AT 11,3;" ";rec2
;" ";d$
7040 PRINT INK 3;AT 13,3;"CATEGO
RIA 3": PRINT AT 14,3;" ";rec
3;" ";u$
7050 PRINT AT 18,1;"PREMERE UN T
ASTO PER RIGIOCARE": PAUSE 0
7060 CLS : LET vite=3: LET pu=0:
GO TO 110
7100 LET n1=0: LET n2=0: LET x=L
EN i$
7110 FOR k=1 TO x
7120 IF n2<32 THEN GO TO 7140
7130 LET n1=n1+1: LET n2=0
7140 PRINT AT n1,n2; INK 0;"0"
7150 BEEP .010,RND*50
7160 PRINT AT n1,n2;i$(k): LET n
2=n2+1
7170 NEXT k
7180 RETURN
8000 FOR u=USR "a" TO USR "q"+7:
READ v: POKE u,v: NEXT u: RETUR
N
8010 DATA 0,0,3,206,166,166,202,
15
8020 DATA 0,0,0,204,54,54,76,192
8030 DATA 4,3,3,3,6,26,20,36
8040 DATA 126,0,0,0,128,96,32,16
8050 DATA 36,36,36,36,36,36,36,4
7
8060 DATA 16,16,16,16,16,16,16,2
08
8070 DATA 127,112,46,95,40,36,34
,35
8080 DATA 248,56,16,208,63,0,127
,128
8090 DATA 0,0,0,0,0,224,220,34
8100 DATA 36,43,44,48,31,12,127,
220
8110 DATA 127,255,0,31,224,64,24
8,12
8120 DATA 145,201,229,229,127,12
7,126,0
8130 DATA 3,7,3,1,227,159,135,31
8140 DATA 192,224,192,126,199,24
9,225,246
8150 DATA 103,143,147,39,71,131,
129,0
8160 DATA 230,241,201,226,226,19
3,126,0
8170 DATA 8,24,28,44,76,94,126,6
0

```



Equazioni di 1° e 2° grado

Questo semplice programma è dedicato ai più piccoli fra i nostri lettori che oltre ad imparare le semplici equazioni di primo e secondo grado, potranno scoprire l'uso del proprio ORIC I nell'ambito matematico oltre a quello consueto dei videogiochi.



```
1 REM#####
2 REM# #
3 REM# Risoluzione di equazioni #
4 REM# di primo e secondo #
5 REM# grado. #
6 REM# #
7 REM# di F. RUSSO list-1985 #
8 REM#####

10 CLS:PRINT:PRINTCHR$(17);:PAPER0:INK1
80 FORT=0T07
90 PLOT0,T,17:PRINT
100 NEXTT:PRINT:PRINT
110 PRINTCHR$(4);CHR$(27)+"J EQUAZIO
NI DI PRIMO E SECONDO
114 PRINT:PRINT:PRINT
120 PRINTSPC(14);CHR$(27)+"JGRADO"
130 FORT=19T026
140 PLOT0,T,18:NEXT
150 WAIT200:PRINTCHR$(4)
180 CLS
190 PLOT2,10,"Equazioni di primo grado..
.....(1)
200 PLOT2,12,"Equazioni di secondo grado
.....(2)
```

```
210 PLOT10,25,"Immetti la scelta"
215 GETA$
220 IFA$="1"THEN500
230 IFA$="2"THEN1000
250 GOTO215
500 CLS:A$="EQUAZIONI DI PRIMO GRADO
510 GOSUB5000
520 CLS:PRINT:PRINTSPC(14);
530 PRINTCHR$(4);CHR$(27)"JAX+B=0"CHR$(4)
)CHR$(17)
540 PRINT:PRINT"-----
-----":PRINT
550 PRINT:PRINT:INPUT"A=";A
560 PRINT:INPUT"B=";B
570 H=520:GOSUB5100
580 GOSUB5040
590 X=-B/A
600 PRINT:PRINTCHR$(4)CHR$(27)"J( "A")X+
( "B")=0"CHR$(4)
610 PRINT:PRINT:PRINT:PRINT"La soluzione
e' : "
620 PRINT:PRINT
630 PRINT"X="X
640 GOTO5150
997 REM
998 REM
999 REM
1000 CLS:A$="EQUAZIONI DI SECONDO GRADO"
:GOSUB5000
1010 CLS:PRINT:PRINTSPC(12);
1020 PRINTCHR$(4);CHR$(27)"JAX^2+BX+C=0"
CHR$(4)CHR$(17)
```

ORIC 1



```
1025 PRINT:PRINT"-----  
-----":PRINT  
1030 PRINT:PRINT:INPUT"A=";A  
1040 PRINT:INPUT"B=";B  
1050 PRINT:INPUT"C=";C  
1060 H=1510  
1070 GOSUB5100  
1090 GOSUB5040  
1130 Z=B^2-(4*A*C)  
1140 IFZ<0THEN1300  
1150 X1=(-B+SQR(Z))/(2*A)  
1160 X2=(-B-SQR(Z))/(2*A)  
1165 PRINT:PRINTCHR$(4)CHR$(27)"J( "A")X  
^2+( "B")X+( "C")=0"CHR$(4)  
1167 PRINT:PRINT  
1170 PRINT:PRINT:PRINT:PRINT"Le due solu  
zioni sono:"  
1175 PRINT:PRINT  
1180 PRINT"X(1)=";X1  
1185 PRINT  
1190 PRINT"X(2)=";X2  
1195 PRINT:PRINT  
1200 PRINT:PRINT:PRINT"Il discriminante  
e' uguale a:"  
1210 PRINT:PRINTZ  
1250 GOSUB5150  
1300 PRINT  
1310 PRINT"L'EQUAZIONE HA SOLUZIONI IMMA  
GINARIE"  
1315 PRINT:PRINT"IL DISCRIMINANTE E' UGU  
ALE A:";PRINT:PRINTZ:PRINT  
1320 PRINT:PRINTSPC(12)"RIPROVA!:"
```

```
1330 WAIT300:GOTO180  
1499 REM  
5000 PLOT9,13,19  
5010 PLOT10,13,A$  
5020 WAIT200  
5030 RETURN  
5037 REM  
5038 REM  
5039 REM  
5040 CLS:PLOT0,13,17:PRINTCHR$(17)  
5050 PLOT15,13,32  
5060 PLOT1,13,"ADESSO CALCOLO"  
5070 PLOT16,13,16  
5080 WAIT 100:CLS  
5090 RETURN  
5097 REM  
5098 REM  
5099 REM  
5100 PRINT:PRINT"SONO ESATTI SI O NO (S/  
N)? ";  
5110 GETA$  
5120 IFA$="S"THENRETURN  
5130 PRINTCHR$(17):POP:GOTOH  
5147 REM  
5148 REM  
5149 REM  
5150 PLOT7,25,"Premi RETURN per continua  
re"  
5160 GETA$  
5170 IFASC(A$)=13THEN180  
5180 PRINTCHR$(17):CLS:END  
5190 RETURN
```



Semina

Il gioco si svolge su una tavola con 14 buchi disposti in forma ellittica, 6 per ogni metà della figura e 2 che serviranno a contenere i semi raccolti dai giocatori. Il computer rappresenta l'avversario.

All'inizio le file opposte di buchi contengono 3 semi ciascuna e lo scopo del gioco è quello di prendere i semi da uno dei propri buchi e porli, distribuendoli uno ad uno in senso antiorario, nei buchi successivi.

Ripetendo più volte l'operazione si deve riuscire a depositare il maggior numero di semi possibile nel proprio buco-magazzino.

Le regole non terminano qui: se al proprio turno si riesce a guadagnare un seme, si guadagna la possibilità di ripetere la mossa; se l'ultimo seme spostato ricade in un buco vuoto, allora si guadagnano il seme e tutti quelli corrispondenti, nella fila opposta, all'avversario.

Il gioco termina quando una delle 2 file di buchi è completamente vuota.

Probabilmente il gioco potrà sembrare un po' complicato, ma in realtà risulterà semplice e divertente.

Buon divertimento.

```
1 REM *****
2 REM **
3 REM * SEMINA *
4 REM **
5 REM *****
6 REM
7 REM ##### INIZIALIZZAZIONE #####

8 REM
9 PAPER4:INK3
10 DEFFNA(X)=X+14*(X>14)
15 DEFFNE(X)=(X<8)OR(X>13)
17 DEFFNX(X)=A(1)+A(2)+A(3)+A(4)+A(5)+A(
6)
18 DEFFNY(X)=A(8)+A(9)+A(10)+A(11)+A(12)
+A(13)
19 DEFFNR(X)=RND(9)*5-2+X
20 DIMA(14),B(4,14)
30 FORI=1TO13:A(I)=3:NEXT:A(7)=0
40 PRINTCHR$(12)CHR$(17):PLOT13,10,"S E
M I N A":PLOT12,10,12
60 WAIT400
65 GOSUB4005:PRINT:PRINT:PRINTCHR$(17);
70 PRINT"Vuoi cominciare a giocare? ";
80 GETZ$:IFZ$<>"S"THEN360
300 FORU=1TO2
303 GOSUB4000:PRINT
```

```
305 PRINT"Immetti la tua mossa":INPUTA$:
U=VAL(A$):IFU=0THEN300
310 IFFNE(U)THENPRINT"Tu puoi solo muove
re da 8 a 13":GOTO300
320 IFA(U)=0THENPRINT"Non ci sono semi l
i' ?":GOTO300
330 X=U:GOSUB2000
335 IFFNX(0)=0ORFNY(0)=0THEN5000
340 IFY=14THENIFU=1THENPRINT"Seconda mos
sa .....":NEXT
345 GOSUB4000:PRINT:PRINT
360 PRINTCHR$(14);
365 PLOT1,14,"Un momento prego.":J=1:GOS
UB510
370 IFFNX(0)=0ORFNY(0)=0THEN5000
380 GOTO300
495 REM
500 REM ##### MOSSA COMPUTER #####
505 REM
507 PRINT
510 B1=0:B2=0:B3=0:B4=0:S1=-91:S2=-91:S3
=-91:S4=-91
518 PRINTCHR$(17);
519 POKE#269,18:PRINTCHR$(17);
520 FORA1=1TO6:IFA(A1)=0THEN900
525 PRINT". ";
530 GOSUB1000:X=A1:GOSUB2000:F=0:IFY<>7T
```



```

HENF=1:GOTO600
540 FORA2=1TO6:IFA(A2)=0THEN800
550 GOSUB1000:X=A2:GOSUB2000
600 FORA3=8TO13:IFA(A3)=0THEN700
610 GOSUB1000:X=A3:GOSUB2000:IFY<>14THEN
680
620 FORA4=8TO13:IFA(A4)=0THEN660
630 GOSUB1000:X=A4:GOSUB2000:IFA(14)-A(7
)>FNR(S4)THENS4=A(14)-A(7):
B4=A4
640 GOSUB1010
660 NEXTA4
670 X=B4:GOSUB2000
680 IFA(14)-A(7)>FNR(S3)THENS3=A(14)-A(7
):B3=A3
685 GOSUB1010
700 NEXTA3:X=B3:GOSUB2000
705 IFF=1THEN810
720 IFA(7)-A(14)>FNR(S2)THENS2=A(7)-A(14
):B2=A2
730 GOSUB1010
800 NEXTA2
805 X=B2:GOSUB2000
810 IFA(7)-A(14)>FNR(S1)THENS1=A(7)-A(14
):B1=A1
820 GOSUB1010
900 NEXTA1
910 X=B1:GOSUB2000:PRINT"La mia mossa e'
";B1;
911 IFFNX(0)=0ORFNY(0)=0THEN5000
912 IFY=7THENX=B2:GOSUB2000:PRINT"e "B2
915 FORI=1TO3000:NEXT
920 PRINT:RETURN
1000 T=T+1:FORZ=1TO14:B(T,Z)=A(Z):NEXT:R
ETURN
1010 FORZ=1TO14:A(Z)=B(T,Z):NEXT:T=T-1:R
ETURN
1999 REM
2000 REM #### MOSSA POSIZIONE X ####
2001 REM
2010 FORZ=X+1TOX+A(X):Y=FNA(Z):A(Y)=A(Y)
+1:NEXT
2025 A(X)=0:Q=-1
2026 IFY=7ORY=14THEN2060
2030 IFA(Y)=1AND(14-Y)>0THENQ=A(14-Y):A(
14-Y)=0:A(Y)=0
2040 IFX<7THENA(7)=A(7)+Q+1:GOTO2060
2050 A(14)=A(14)+Q+1
2060 RETURN
3997 REM
3998 REM #### TABELLA SCHERMO ####
3999 REM
~000, WAIT100
4005 PRINTCHR$(12):PLOT0,1,2:PLOT0,3,6
4006 PLOT0,9,6:PLOT0,11,2:PLOT0,6,2

```

```

4010 PRINT"      13 12 11 10 9 8
4020 PRINT" Tu ----- Io
4030 PRINT"      ";FORI=13TO8STEP-1:A$=S
TR$(A(I))
4040 IFLEN(A$)>2THENA$=RIGHT$(A$,2):GOTO
4055
4050 PRINT" ";
4055 PRINTA$":";:NEXT:PRINT" "
4060 PRINT" <----->
4070 PRINT"-----
4080 A$=STR$(A(14)):IFLEN(A$)>2THENA$=RI
GHT$(A$,2)
4090 PRINTA$":      ";A$
=STR$(A(7))
4100 IFLEN(A$)>2THENA$=RIGHT$(A$,2)
4105 PRINTA$":
4110 PRINT"-----
4120 PRINT" <-#----->
4130 PRINT" ";
4135 FORI=1TO6:A$=STR$(A(I))
4137 IFLEN(A$)>2THENA$=RIGHT$(A$,2):GOTO
4143
4140 PRINT" ";
4143 PRINTA$":";:NEXT
4145 PRINT
4150 PRINT" -----
4160 PRINT"      1 2 3 4 5 6
4200 RETURN
4997 REM
4998 REM #### FINE GIOCO ####
4999 REM
5000 PRINTCHR$(12)
5005 IFA(14)=A(7)THENS100
5010 IFA(14)>A(7)THENS060
5020 PRINT"Io ho vinto per "A(7)-A(14)
5030 GOTO5200
5060 PRINT"Hai vinto!!! Ma solo per "A
(14)-A(7)
5080 GOTO5200
5100 PRINT"Abbiamo fatto "A(7)
5200 PRINT"Vuoi giocare ancora?"
5210 GETZ$:IFZ$="S"THENRUN
5220 END
507 PRINT
520 FORA1=1TO6:IFA(A1)=0THEN900
630 GOSUB1000:X=A4:GOSUB2000:IFA(14)-A(7
)>FNR(S4)THENS4=A(14)-A(7):
B4=A4
4090 PRINTA$":      ";A$
=STR$(A(7))
4120 PRINT" <----->

```



Arredatore elettronico

Questo programma costituisce un esempio di come sfruttare le elevate capacità grafiche e di calcolo del Commodore 64 in modo semplice ed efficace. Si è scelto di operare con il SIMON'S BASIC per la possibilità che questo TOOL offre di gestire con facilità schermi grafici ed alfanumerici contemporaneamente e per la realizzazione diretta di figure geometriche senza complicate routines in linguaggio macchina, che il Simon's permette mediante semplici istruzioni basic (di cui il CBM 64 in configurazione base è carente).

Il programma serve a determinare la migliore disposizione dei mobili di una stanza lavorando sulla pianta di questa, riprodotta sullo schermo, ottenendo i vari spostamenti mediante joystick.

Esaminiamo in dettaglio il funzionamento del programma.

Dopo una presentazione grafica (subroutine 5000-5140) e la richiesta di istruzioni, riportate sommariamente nella subroutine 6000-6210, le linee 58-84 richiedono le dimensioni della stanza in esame. La linea 30 è molto importante poiché definisce la scala utilizzata per tracciare la pianta, permettendo la riproduzione sullo schermo sia di ambienti ristretti che di ambienti più ampi, senza alcuna limitazione particolare. Le linee da 100 a 342 definiscono lo schermo grafico dividendolo in due aree: a sinistra viene limitata l'area di lavoro entro cui realizzare la pianta della stanza, a destra vengono riprodotti i quattro tasti funzione presenti sulla tastiera del CBM 64, debitamente abilitati. Inizialmente la pianta della stanza è rappresentata da un rettangolo chiuso; l'utente deve completarla inserendo porte e finestre, come discontinuità, nelle linee che definiscono le pareti. È possibile, inoltre, modificare la pianta data inserendo angoli e/o rientranze, per rappresentare ambienti dalla geometria più irregolare (ad es. corridoi con sviluppo a L). Per fare tutto ciò occorre disegnare o cancellare linee; nel primo caso si deve premere il tasto F1 e, mantenendo

questo premuto, spostare il joystick nella direzione voluta, mentre per cancellare si dovrà premere il tasto F3 e, manovrando il joystick, portare il puntino cursore, inizialmente alle coordinate 100, 100; sopra la linea da cancellare.

A questo punto ripassando sopra detta linea per il tratto corrispondente alla lunghezza della porta o della finestra che si vuole disegnare, si ottiene l'apertura della linea. Per distinguere la porta, o le porte, dalle finestre, è possibile stampare una P davanti a questa/e, premendo F7, posizionando il cursore davanti alla relativa apertura e premendo il tasto di fuoco, (subroutine 4000-4120). Al termine del disegno della pianta, premendo F5, il computer chiede se il disegno è soddisfacente: in caso di risposta negativa si ricomincia daccapo, altrimenti il programma procede ridefinendo tutti i tasti funzione. Si può ora iniziare ad inserire i mobili. Le linee da 350 a 530 riportano sullo schermo le nuove funzioni dei tasti, mentre da 600 a 694 il programma procede nell'acquisizione della forma e delle dimensioni del mobile. Le linee da 700 a 840 posizionano il mobile; una volta inserite forma e dimensioni, si torna allo schermo grafico, per cui premendo F3 è possibile porre il cursore nel punto in cui si vuole posizionare il mobile; raggiunta la posizione è sufficiente schiacciare il tasto di fuoco sul joystick per visualizzarlo. Nel caso di mobili circolari, la posizione del cursore rappresenta il centro, negli altri casi l'angolo superiore sinistro. Nell'introduzione dei dati relativi al mobile, per una rappresentazione ottimale è bene introdurre la dimensione maggiore come lunghezza e la dimensione minore come larghezza. Premendo F5 è possibile ruotare di 90 gradi verso il basso l'ultimo mobile inserito, mentre con F7 è possibile spostare con il joystick un qualsiasi mobile già posizionato (linee 2000-2340). Premendo la barra spaziatrice si accede ad altre 'opzioni' ottenute ridefinendo ulteriormente i tasti funzione.



```
130 LINE 245,10,245,10+M,1:TEXT 235,20,"
M",1,1,8
135 TEXT 235,30,"M",1,1,8
140 BLOCK 260,0,318,20,1:BLOCK 260,50,31
8,70,1
150 BLOCK 260,100,318,120,1:BLOCK 260,15
0,318,170,1
160 TEXT 262,2,"INIZIO",2,1,8
165 TEXT 262,11,"TRACCIA",2,1,8
170 TEXT 262,52,"FINE",2,1,8
175 TEXT 262,61,"TRACCIA",2,1,8
180 TEXT 262,102,"FINE",2,1,8
190 TEXT 262,111,"PIANTA",2,1,8
193 TEXT 262,152,"PORTA",2,1,8
195 REC 5,5,LU1,LA1,1
200 X=100:Y=100:PLOT X,Y,1
203 A#="":GETA#:IFA#=""THEN203
205 IFA#=""THEN1
206 IFA#=""THENGOSUB4000:GOTO203
207 IFA#=""THEN2
208 IFA#=""THEN310
209 PLOT X,Y,T
210 IFJOY=1THENY=Y-1:GOSUB1000:GOTO203
220 IFJOY=2THENY=Y-1:X=X+1:GOSUB1000:GOT
0203
230 IFJOY=3THENX=X+1:GOSUB1000:GOTO203
240 IFJOY=4THENX=X+1:Y=Y+1:GOSUB1000:GOT
0203
250 IFJOY=5THENY=Y+1:GOSUB1000:GOTO203
260 IFJOY=6THENY=Y+1:X=X-1:GOSUB1000:GOT
0203
270 IFJOY=7THENX=X-1:GOSUB1000:GOTO203
280 IFJOY=8THENX=X-1:Y=Y-1:GOSUB1000:GOT
0203
300 GOTO210
310 TEXT 5,191,"OK? (S/N)",1,1,8
320 GETA#:IFA#=""THEN320
330 IFA#=""THEN100
340 IFA#<>"S"THEN320
342 BLOCK 5,191,80,199,0
345 :
347 REM ***INSERIMENTO MOBILI***
348 :
350 BLOCK 260,0,318,20,0:BLOCK 260,50,31
8,70,0
360 BLOCK 260,100,318,120,0:BLOCK 260,15
0,318,170,0
370 BLOCK 260,0,318,20,1:BLOCK 260,50,31
8,70,1
380 BLOCK 260,100,318,120,1:BLOCK 260,15
0,318,170,1
390 TEXT 262,2,"INSER.",2,1,8
400 TEXT 262,11,"MOBILE",2,1,8
410 TEXT 262,52,"POSIZ.",2,1,8
420 TEXT 262,61,"MOBILE",2,1,8
430 TEXT 262,102,"RUOTA",2,1,8
440 TEXT 262,111,"ULTIMO",2,1,8
450 TEXT 262,152,"SPOSTA",2,1,8
460 TEXT 262,161,"MOBILE",2,1,8
462 TEXT 262,173,"SPAZIO",1,1,8
464 TEXT 262,181,"* MENU",1,1,8
470 B#="":GETB#:IFB#=""THEN470
480 IFB#=""THENGOSUB600:GOTO470
490 IFB#=""THENGOSUB700:GOTO470
500 IFB#=""THENGOSUB900:GOTO470
510 IFB#=""THENGOSUB2000:GOTO470
520 IFB#=""THEN3000
530 IFB#<>"ANDB#<>"ANDB#<>"ANDB#<>"
ANDB#<>"ANDB#<>"THEN470
597 :
598 REM ***INPUT DIM. MOBILE***
599 :
600 CSET 0:I=I+1:N=N+1
602 PRINT"DENOMINAZIONE DEL MOBILE?"
604 INPUT$(I)
610 PRINT"FORMA DEL MOBILE:"
614 PRINT"R] RETTANGOLARE"
620 PRINT"C] CIRCOLARE"
625 PRINT"E] ELLITTICA"
630 PRINT"PREMI IL TASTO CORRISPONDEN
TE"
640 GETC#:IFC#=""THEN640
650 IFC#="R"THEN670
655 IFC#="E"THEN665
660 IFC#<>"C"THEN640
662 F(I)=2:INPUT"RAGGIO (IN M.)=":R:
R1(I)=INT(R*M):R2(I)=INT(R*M)
664 GOTO680
665 F(I)=2:INPUT"RAGGIO 1 (IN M.)=":
RA:R1(I)=INT(RA*M)
666 INPUT"RAGGIO 2 (IN M.)=":RB:R2(I)=I
NT(RB*M):GOTO680
670 F(I)=1:INPUT"R LUNGHEZZA (IN M.)=":
H:H1(I)=INT(H*M)
672 INPUT"R LARGHEZZA (IN M.)=":K:K1(I)
=INT(K*M)
680 PRINT"MI DATI SONO OK? (S/N)"
685 GETD#:IFD#=""THEN685
690 IFD#=""THEN610
692 IFD#<>"S"THEN685
694 CSET 2:RETURN
697 :
698 REM ***POSIZIONE MOBILE***
699 :
700 X=100:Y=100:PLOT X,Y,1
709 PLOT X,Y,2
710 IFJOY=1THENY=Y-1:GOSUB1000:GOTO709
720 IFJOY=2THENY=Y-1:X=X+1:GOSUB1000:GOT
0709
730 IFJOY=3THENX=X+1:GOSUB1000:GOTO709
740 IFJOY=4THENX=X+1:Y=Y+1:GOSUB1000:GOT
0709
750 IFJOY=5THENY=Y+1:GOSUB1000:GOTO709
760 IFJOY=6THENY=Y+1:X=X-1:GOSUB1000:GOT
0709
770 IFJOY=7THENX=X-1:GOSUB1000:GOTO709
780 IFJOY=8THENX=X-1:Y=Y-1:GOSUB1000:GOT
0709
785 IFJOY=128THEN790
787 GOTO710
790 ON F(I)GOTO792,793
792 X(I)=X:Y(I)=Y:REC X(I),Y(I),H1(I),K1
(I),1:GOTO794
793 X(I)=X:Y(I)=Y:CIRCLE X(I),Y(I),R1(I)
,R2(I),1
794 TEXT 5,191,"OK? (S/N)",1,1,8
795 GETA#:IFA#=""THEN795
```



```

796 IFA#="N"THEN800
797 IFA#<>"S"THEN795
798 BLOCK 5,191,80,199,0:RETURN
800 BLOCK 5,191,80,199,0
810 ON F(I)GOTO820,830
820 REC X(I),Y(I),H1(I),K1(I),0
830 CIRCLE X(I),Y(I),R1(I),R2(I),0
840 GOTO700
897 :
898 REM ***ROTAZIONE MOBILE***
899 :
900 ONF(I)GOTO910,950
910 IFFL(I)THEN940
930 REC X(I),Y(I),H1(I),K1(I),0:REC X(I)
,Y(I),K1(I),H1(I),1:FL(I)=1:RETURN
940 REC X(I),Y(I),K1(I),H1(I),0:REC X(I)
,Y(I),H1(I),K1(I),1:FL(I)=0:RETURN
950 IFFL(I)THEN970
960 CIRCLE X(I),Y(I),R1(I),R2(I),0:CIRCL
E X(I),Y(I),R2(I),R1(I),1:FL(I)=1:RETURN
970 CIRCLE X(I),Y(I),R2(I),R1(I),0:CIRCL
E X(I),Y(I),R1(I),R2(I),1:FL(I)=0:RETURN
997 :
998 REM ***CONTROLLO DEL JOYSTICK***
999 :
1000 IFX<STHENX=5:PLOT X,Y,1:RETURN
1010 IFX>250THENX=250:PLOT X,Y,1:RETURN
1020 IFY<STHENY=5:PLOT X,Y,1:RETURN
1030 IFY>187THENY=187:PLOT X,Y,1:RETURN
1040 PLOT X,Y,1:RETURN
1997 :
1998 REM ***SPOSTAMENTO MOBILI***
1999 :
2000 CSET 0:PRINT"MOBILI FINORA DISPOST
I:"
2010 FORI=1TON
2020 PRINTI,T$(I):PRINT
2030 NEXT
2040 PRINT"MOBATTI IL NUMERO CORRISPONDE
NTE AL MO-"
2050 PRINT"BILE CHE VUOI SPOSTARE"
2060 NU=0:INPUTNU:IFNU=0THENCSET 2:RETUR
N
2070 CSET 2:ONF(NU)GOTO2075,2085
2075 IFFL(NU)THEN2082
2080 REC X(NU),Y(NU),H1(NU),K1(NU),0:GOT
O2100
2082 REC X(NU),Y(NU),K1(NU),H1(NU),0:GOT
O2100
2085 IFFL(NU)THEN2095
2090 CIRCLE X(NU),Y(NU),R1(NU),R2(NU),0
2095 CIRCLE X(NU),Y(NU),R2(NU),R1(NU),0
2100 X=X(NU):Y=Y(NU):PLOT X,Y,1
2110 PLOT X,Y,2
2120 IFJOY=1THENY=Y-1:GOSUB1000:GOTO2110
2130 IFJOY=2THENY=Y-1:X=X+1:GOSUB1000:GO
T02110
2140 IFJOY=3THENX=X+1:GOSUB1000:GOTO2110
2150 IFJOY=4THENX=X+1:Y=Y+1:GOSUB1000:GO

```

```

T02110
2160 IFJOY=5THENY=Y+1:GOSUB1000:GOTO2110
2170 IFJOY=6THENY=Y+1:X=X-1:GOSUB1000:GO
T02110
2180 IFJOY=7THENX=X-1:GOSUB1000:GOTO2110
2190 IFJOY=8THENX=X-1:Y=Y-1:GOSUB1000:GO
T02110
2200 IFJOY=128THEN2220
2210 GOTO2120
2220 ON F(NU)GOTO2230,2240
2230 X(NU)=X:Y(NU)=Y:IFFL(NU)THEN2235
2232 REC X(NU),Y(NU),H1(NU),K1(NU),1:GOT
O2250
2235 REC X(NU),Y(NU),K1(NU),H1(NU),1:GOT
O2250
2240 X(NU)=X:Y(NU)=Y:IFFL(NU)THEN2245
2242 CIRCLE X(NU),Y(NU),R1(NU),R2(NU),1:
GOTO2250
2245 CIRCLE X(NU),Y(NU),R2(NU),R1(NU),1
2250 TEXT 5,191,"OK? (S/N)",1,1,8
2260 GETA#:IFA#=""THEN2260
2270 IFA#="N"THEN2300
2280 IFA#<>"S"THEN2260
2290 BLOCK 5,191,80,199,0:RETURN
2300 BLOCK 5,191,80,199,0
2310 ON F(NU)GOTO2315,2320
2315 IFFL(NU)THEN2325
2320 REC X(NU),Y(NU),H1(NU),K1(NU),0:GOT
O2340
2325 REC X(NU),Y(NU),K1(NU),H1(NU),0:GOT
O2340
2328 ILFL(NU)THEN2335
2330 CIRCLE X(NU),Y(NU),R1(NU),R2(NU),0:
GOTO2340
2335 CIRCLE X(NU),Y(NU),R2(NU),R1(NU),0
2340 GOTO2100
2997 :
2998 REM ***ULTIME OPZIONI***
2999 :
3000 BLOCK 260,0,318,20,0:BLOCK 260,50,3
18,70,0
3010 BLOCK 260,100,318,120,0:BLOCK 260,1
50,318,170,0
3020 BLOCK 260,0,318,20,1:BLOCK 260,50,3
18,70,1
3030 BLOCK 260,100,318,120,1
3040 TEXT 262,2,"ALTRA",2,1,8
3050 TEXT 262,11,"PIANTA",2,1,8
3060 TEXT 262,52,"FINE",2,1,8
3070 TEXT 262,61,"LAVORO",2,1,8
3075 TEXT 262,102,"RUOTA",2,1,8
3080 TEXT 262,111,"MOBILE",2,1,8
3090 B#=""GETB#:IFB#=""THEN3090
3100 IFB#="■"THEN NRM:RUN60
3110 IFB#="□"THEN NRM:PRINT"J":END
3115 IFB#="▣"THEN GOSUB7000:GOTO3090
3117 IFB#=" "THEN350
3120 IFB#<>"■"ANDB#<>"□"ANDB#<>"▣"ANDB#<>
" "THEN3090
3997 :
3998 REM ***POSIZIONE PORTA***
3999 :

```




Caccia alla base

Cerca le 20 basi invisibili che il computer ha celato nella griglia di destra. L'apparire di un quadratino rosso ti informerà del piazzamento completato.

Ora tocca a te disporre altrettante fasi sulla griglia di sinistra digitando prima una lettera dalla A alla J e poi un numero dallo 0 al 9 per vederle sullo schermo (inutile contarle poiché il computer farà apparire un quadratino giallo al raggiungimento del numero 20).

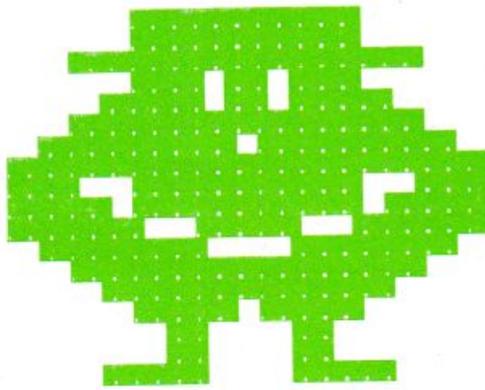
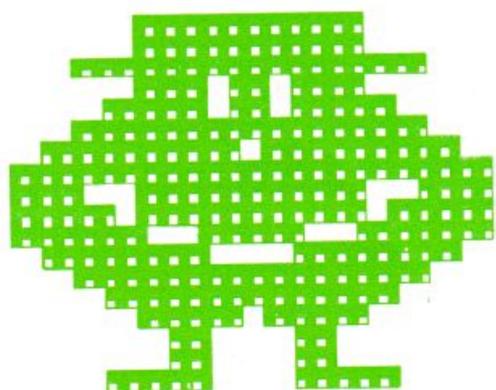
Seguendo la stessa procedura (lettera-numero) devi cercare di «scovare» le basi nemiche che, se ne avrai individuata la posizione, verranno visualizzate in rosso mentre nel caso di colpo fallito verrà visualizzata una X.

Il primo colpo è sempre a favore del computer che farà diventare viola una tua base scoperta e visualizzerà la solita X in caso contrario.

Vince, ovviamente, chi per primo scopre le 20 basi avversarie. Digitare Z per una nuova partita.

VARIABILI PRINCIPALI

A	Basi computer (20)
Z=1	Disposizione basi del giocatore
Z=2	Ricerca basi del computer
H	Basi del giocatore (20)
T	Calcolo vincita del giocatore
V	Calcolo vincita del computer
L	Lettere griglia sinistra
M	Colonna lettere griglia sinistra
LL	Lettere griglia destra
MM	Colonna lettere griglia destra
P	Numeri griglia sinistra
Q	Riga numeri griglia sinistra
PP	Numeri griglia destra
QQ	Riga numeri griglia destra
R	Riga RND basi computer
C	Colonna RND basi computer
N	Controllo spazio vuoto per immissione basi computer
F	Coordinata lettera
G	Coordinata numero
U	Ricerca colpo giocatore
CC	Colonna RND colpo computer
RR	Riga RND colpo computer
O	Ricerca colpo computer



```

10 CALL CLEAR
20 CALL SCREEN(11)
30 PRINT "CACCIA ALLA BASE"
40 PRINT " "
50 PRINT "DADO 1984"
60 FOR YOU=1 TO 2000
70 NEXT YOU
80 CALL CLEAR
90 CALL SCREEN(5)
100 CALL CHAR(96,"007E7E7E7E7E")
110 CALL CHAR(104,"007E7E7E7E7E")

```

```

120 CALL CHAR(112,"007E7E7E7E7E")
130 CALL CHAR(120,"007E7E7E7E7E")
140 CALL CHAR(128,"00666618186666")
150 CALL CHAR(136,"007E7E7E7E7E")
160 CALL COLOR(9,5,16)
170 CALL COLOR(7,16,5)
180 CALL COLOR(10,1,16)
190 CALL COLOR(11,10,16)
200 CALL COLOR(12,11,16)
210 CALL COLOR(13,2,16)
220 CALL COLOR(14,14,16)
230 CALL COLOR(3,16,5)
240 CALL COLOR(4,16,5)
250 CALL COLOR(8,16,5)
260 CALL COLOR(5,16,5)
270 CALL COLOR(6,16,5)
280 A=0

```

```

290 Z=1
300 H=0
310 T=0
320 V=0
330 REM
340 REM
350 REM
360 FOR X=8 TO 17
370 CALL HCHAR(X,5,96,10)
380 NEXT X
390 FOR Y=8 TO 17
400 CALL HCHAR(Y,19,96,10)
410 NEXT Y
420 L=65
430 M=5
440 GOTO 480
450 L=L+1

```

```

460 M=M+1
470 IF M>14 THEN 500
480 CALL HCHAR(7,M,L)
490 GOTO 450
500 LL=65
510 MM=19
520 GOTO 560
530 LL=LL+1
540 MM=MM+1
550 IF MM>28 THEN 580
560 CALL HCHAR(7,MM,LL)
570 GOTO 530
580 P=48
590 Q=8
600 GOTO 640
610 P=P+1
620 Q=Q+1
630 IF Q>17 THEN 660
640 CALL HCHAR(Q,4,P)
650 GOTO 610
660 PP=48
670 QQ=8
680 GOTO 720
690 PP=PP+1
700 QQ=QQ+1
710 IF QQ>17 THEN 740
720 CALL HCHAR(QQ,18,PP)
730 GOTO 690
740 RANDOMIZE
750 R=INT(10*RND)+8
760 RANDOMIZE
770 C=INT(10*RND)+19
780 CALL GCHAR(R,C,N)
790 IF N=104 THEN 740
800 A=A+1
810 IF A>20 THEN 850
820 CALL HCHAR(R,C,104)
830 CALL SOUND(100,1760,10)
840 GOTO 740
850 CALL HCHAR(5,5,112)
860 GOTO 870
870 CALL KEY(0,K,S)
880 IF S=0 THEN 870
890 IF S=-1 THEN 870
900 IF K=65 THEN 1010
910 IF K=66 THEN 1060
920 IF K=67 THEN 1110
930 IF K=68 THEN 1160
940 IF K=69 THEN 1210
950 IF K=70 THEN 1260
960 IF K=71 THEN 1310
970 IF K=72 THEN 1360
980 IF K=73 THEN 1410
990 IF K=74 THEN 1460
1000 GOTO 870
1010 ON Z GOTO 1020,1040
1020 F=5
1030 GOTO 1510
1040 F=19
1050 GOTO 1510
1060 ON Z GOTO 1070,1090
1070 F=6
1080 GOTO 1510
1090 F=20
1100 GOTO 1510
1110 ON Z GOTO 1120,1140
1120 F=7
1130 GOTO 1510
1140 F=21
1150 GOTO 1510
1160 ON Z GOTO 1170,1190
1170 F=8
1180 GOTO 1510

```

```

1190 F=22
1200 GOTO 1510
1210 ON Z GOTO 1220,1240
1220 F=9
1230 GOTO 1510
1240 F=23
1250 GOTO 1510
1260 ON Z GOTO 1270,1290
1270 F=10
1280 GOTO 1510
1290 F=24
1300 GOTO 1510
1310 ON Z GOTO 1320,1340
1320 F=11
1330 GOTO 1510
1340 F=25
1350 GOTO 1510
1360 ON Z GOTO 1370,1390
1370 F=12
1380 GOTO 1510
1390 F=26
1400 GOTO 1510
1410 ON Z GOTO 1420,1440
1420 F=13
1430 GOTO 1510
1440 F=27
1450 GOTO 1510
1460 ON Z GOTO 1470,1490
1470 F=14
1480 GOTO 1510
1490 F=28
1500 GOTO 1510
1510 CALL KEY(0,K,S)
1520 IF S=0 THEN 1510
1530 IF S=-1 THEN 1510
1540 IF K=48 THEN 1650
1550 IF K=49 THEN 1670
1560 IF K=50 THEN 1690
1570 IF K=51 THEN 1710
1580 IF K=52 THEN 1730
1590 IF K=53 THEN 1750
1600 IF K=54 THEN 1770
1610 IF K=55 THEN 1790
1620 IF K=56 THEN 1810
1630 IF K=57 THEN 1830
1640 GOTO 1510
1650 G=8
1660 ON Z GOTO 1850,1960
1670 G=9
1680 ON Z GOTO 1850,1960
1690 G=10
1700 ON Z GOTO 1850,1960
1710 G=11
1720 ON Z GOTO 1850,1960
1730 G=12
1740 ON Z GOTO 1850,1960
1750 G=13
1760 ON Z GOTO 1850,1960
1770 G=14
1780 ON Z GOTO 1850,1960
1790 G=15
1800 ON Z GOTO 1850,1960
1810 G=16
1820 ON Z GOTO 1850,1960
1830 G=17
1840 ON Z GOTO 1850,1960
1850 CALL GCHAR(G,F,W)
1860 IF W=120 THEN 870
1870 H=H+1
1880 IF H>20 THEN 1920
1890 CALL HCHAR(G,F,120)
1900 CALL SOUND(100,277,10)
1910 GOTO 870

```

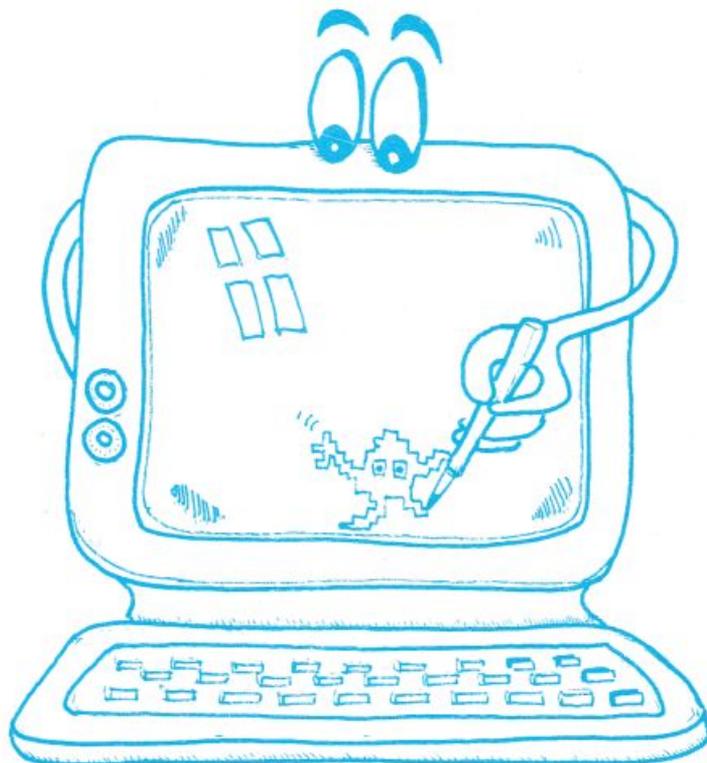
```

1920 CALL HCHAR(20,5,120)
1930 CALL SOUND(100,330,10,494,10,784,10)
1940 Z=2
1950 GOTO 2110
1960 CALL GCHAR(G,F,U)
1970 IF U=96 THEN 2030
1980 IF U=104 THEN 2060
1990 IF U=112 THEN 2010
2000 IF U=128 THEN 2010
2010 CALL SOUND(100,110,10)
2020 GOTO 870
2030 CALL HCHAR(G,F,128)
2040 CALL SOUND(100,-5,10)
2050 GOTO 2110
2060 CALL HCHAR(G,F,112)
2070 CALL SOUND(500,523,10,740,10,831,10)
2080 T=T+1
2090 IF T=20 THEN 2280
2100 GOTO 2110
2110 RANDOMIZE
2120 CC=INT(10*RND)+5
2130 RANDOMIZE
2140 RR=INT(10*RND)+8
2150 CALL GCHAR(RR,CC,0)
2160 IF 0=96 THEN 2200
2170 IF 0=120 THEN 2230
2180 IF 0=128 THEN 2110
2190 IF 0=136 THEN 2110
2200 CALL HCHAR(RR,CC,128)
2210 CALL SOUND(100,-7,10)
2220 GOTO 870
2230 CALL HCHAR(RR,CC,136)
2240 CALL SOUND(500,110,10,117,10,123,10)
2250 V=V+1
2260 IF V=20 THEN 2550
2270 GOTO 870
2280 PRINT TAB(8);"E VA BENE HAI VINTO"
2290 CALL SOUND(200,233,10)
2300 CALL SOUND(200,262,10)
2310 CALL SOUND(400,277,10)
2320 CALL SOUND(200,262,10)
2330 CALL SOUND(200,233,10)
2340 CALL SOUND(400,220,10)
2350 CALL SOUND(200,233,10)
2360 CALL SOUND(200,262,10)
2370 CALL SOUND(400,175,10)
2380 CALL SOUND(200,196,10)
2390 CALL SOUND(200,220,10)
2400 CALL SOUND(400,233,10)
2410 CALL SOUND(200,208,10)
2420 CALL SOUND(200,185,10)
2430 CALL SOUND(400,175,10)
2440 CALL SOUND(200,156,10)
2450 CALL SOUND(200,139,10)
2460 CALL SOUND(400,131,10)
2470 CALL SOUND(200,139,10)
2480 CALL SOUND(200,156,10)
2490 CALL SOUND(200,175,10)
2500 CALL SOUND(200,156,10)
2510 CALL SOUND(200,139,10)
2520 CALL SOUND(200,131,10)
2530 CALL SOUND(400,117,10)
2540 GOTO 2570
2550 PRINT TAB(8);"TI HO VINTO UMAND"
2560 CALL SOUND(1000,-7,10)
2570 PRINT TAB(8);"Z FOR PLAY"
2580 CALL KEY(0,K,S)
2590 IF S=0 THEN 2580
2600 IF S=-1 THEN 2580
2610 IF K=90 THEN 80
2620 GOTO 2580

```


ATTENZIONE!

Per un diverso interfacciamento con la stampante il segno ' (apostrofo) corrisponde agli apici ".
Fate attenzione nel digitare il programma.
10 PRINT '□' DIVENTA: 10 PRINT"□"



```

2767 REM *****
2768 REM * SUBROUTINE DATO ERRATO *
2769 REM *****
2770 PRINT' : DATO ERRATO. RIPETERE PREG
0... : FORX=1TO1000: NEXT: RETURN
2997 REM *****
2998 REM * SUBROUTINE MUSICA 1 *
2999 REM *****
3000 POKE54277,40:POKE54284,40:POKE54291
,40:REM SET ATTACK/DECAY VOCE 1,2,3
3010 POKE54278,40:POKE54285,40:POKE54292
,40:REM SET SUSTAIN/RELEASE VOCE 1,2,3
3023 POKE51283,33:REM WAVEFORM V1
3024 POKE51230,PEEK(51283)-1
3025 POKE51357,33:REM WAVEFORM V2
3026 POKE51304,PEEK(51357)-1
3027 POKE51431,33:REM WAVEFORM V3
3028 POKE51378,PEEK(51431)-1
3029 POKE54274,0:REM LO-PULSE V1
3030 POKE54275,0:REM HI-PULSE V1
3031 POKE54281,0:REM LO-PULSE V2
3032 POKE54282,0:REM HI-PULSE V2
3033 POKE54288,0:REM LO-PULSE V3
3034 POKE54289,0:REM HI-PULSE V3
3040 REM * INIZIALIZZ.PARAM. MUSICA 1 *
3045 POKE56333,3:POKE49913,182:POKE49914
,206
3050 POKE49918,178:POKE49919,204:POKE499
10,178:POKE49911,204:GOTO3300
3097 REM *****
3098 REM * SUBROUTINE MUSICA 2 *
3099 REM *****

```

```

3100 POKE54277,40:POKE54284,40:POKE54291
,40:REM SET ATTACK/DECAY VOCE 1,2,3
3110 POKE54278,40:POKE54285,40:POKE54292
,40:REM SET SUSTAIN/RELEASE VOCE 1,2,3
3123 POKE51283,33:REM WAVEFORM V1
3124 POKE51230,PEEK(51283)-1
3125 POKE51357,33:REM WAVEFORM V2
3126 POKE51304,PEEK(51357)-1
3127 POKE51431,33:REM WAVEFORM V3
3128 POKE51378,PEEK(51431)-1
3129 POKE54274,0:REM LO-PULSE V1
3130 POKE54275,0:REM HI-PULSE V1
3131 POKE54281,0:REM LO-PULSE V2
3132 POKE54282,0:REM HI-PULSE V2
3133 POKE54288,0:REM LO-PULSE V3
3134 POKE54289,0:REM HI-PULSE V3
3140 REM * INIZIALIZZ.PARAM. MUSICA 2 *
3145 POKE56333,3:POKE49910,0:POKE49911,2
02:POKE49918,0:POKE49919,202
3150 POKE49913,8:POKE49914,203:GOTO3300
3197 REM *****
3198 REM * SUBROUTINE MUSICA 3 *
3199 REM *****
3200 POKE54277,40:POKE54284,40:POKE54291
,40:REM SET ATTACK/DECAY VOCE 1,2,3
3210 POKE54278,40:POKE54285,40:POKE54292
,40:REM SET SUSTAIN/RELEASE VOCE 1,2,3
3223 POKE51283,33:REM WAVEFORM V1
3224 POKE51230,PEEK(51283)-1
3225 POKE51357,33:REM WAVEFORM V2
3226 POKE51304,PEEK(51357)-1
3227 POKE51431,33:REM WAVEFORM V3

```



```
3228 POKE51378,PEEK(51431)-1
3229 POKE54274,0:REM LO-PULSE V1
3230 POKE54275,0:REM HI-PULSE V1
3231 POKE54281,0:REM LO-PULSE V2
3232 POKE54282,0:REM HI-PULSE V2
3233 POKE54288,0:REM LO-PULSE V3
3234 POKE54289,0:REM HI-PULSE V3
3240 REM * INIZIALIZZ.PARAM. MUSICA 3 *
3245 POKE56333,3:POKE49910,11:POKE49911,
203
3250 POKE49918,11:POKE49919,203:POKE4991
3,175:POKE49914,204
3299 REM * START MUSICA *
3300 POKE49915,1:POKE49916,1:POKE49917,1
:POKE49912,1
3305 POKE54296,15:REM SET VOLUME
3310 POKE788,0:POKE789,200:POKE56333,131
:RETURN
3997 REM *****
3998 REM * ANIMAZIONE SPRITE EGIZIANO *
3999 REM *****
4000 POKEVR+1,112:POKEVR+3,133:REM COORD
.Y
4005 POKEVR+28,3:REM ABILITAZ. MULTIC.
4010 POKE2040,13:POKE2041,14:REM PUNTATO
RI BLOCCHI SPRITE
4015 POKEVR+37,1:POKEVR+38,8:POKEVR+39,2
:POKEVR+40,0:REM COLORI
4019 REM INIZ. ROUTINE L.M.
4020 POKE253,126:POKE254,158:POKE52991,2
52:POKE52990,0:POKE52988,126
4025 POKE52989,159:POKE52986,126:POKE529
87,158:POKE52983,254:POKE52984,158
4050 PRINT' TAB(9)'
4060 PRINTTAB(9)'
4070 PRINT'
4080 PRINT'
4090 PRINT'
4100 PRINT'
4110 PRINT'
4120 PRINT' TAB(28)'
4130 PRINTTAB(27)'
4140 PRINTTAB(26)'
4150 PRINTTAB(25)'
4160 PRINTTAB(24)'
4170 PRINTTAB(23)'
4180 PRINTTAB(22)'
4190 PRINTTAB(21)'
4200 PRINTTAB(20)'
4210 PRINT' MORETTO
4220 PRINTTAB(25)'
4230 PRINT'
4240 PRINT'
4250 PRINT'
4260 PRINT'
4270 PRINT'
4280 PRINT'
4290 PRINT'
4300 PRINT'
```

```
PRESENTA:
4310 PRINT'
4320 PRINT'
4390 POKE2023,241:POKE56295,2:POKEVR+16,
0:POKEVR+23,0:POKEVR+29,0
4400 POKEVR+27,3:POKEVR+21,3:RETURN
20000 REM *****
20001 REM * PROCEDURA PRINCIPALE *
20002 REM *****
20003 PRINT' :POKE53281,14:GOSUB44
20005 PRINT' *****
*****';
20010 PRINT'*****
*****';
20015 PRINT'***** MENU' PRINCIPALE
*****';
20020 PRINT'*****
*****';
20025 PRINT'*****
*****';
20030 PRINT' [1] SPRITE NORMALE
20035 PRINT' [2] SPRITE MULTICOLOR
20037 PRINT' [3] MUSICA ON/OFF'
20039 PRINT' [4] MUSICA 1/2/3'
20040 PRINT' [5] END'
20045 INPUT' [0] QUALE OPZIONE
[ ]';M
20050 IFM<10RM>5THENGOSUB2768:GOTO20045
20055 INPUT' TUTTO OK [S/N] '$
20060 IFLEFT$(X$,1)='N'THENPRINT'TTTTT'
':GOTO20045
20065 ONMGOTO20100,20200,20067,20070
20066 POKE49912,0:PRINT' OK! PADRONE
':END
20067 IFPEEK(49912)<0THENPOKE49912,0:GO
T20005:REM POKE49912,0 TERMINA MUSICA
20068 POKE56333,3:POKE54296,15:POKE788,0
:POKE789,200:POKE49912,1
20069 POKE56333,131:GOTO20005
20070 MU=MU+1:IFMU=4THENMU=1
20075 ONMUGOSUB2998,3098,3198
20080 GOTO20005
20100 PRINT' TAB(12)'SPRITE NORMALE':
GOSUB2708
20165 ONXGOSUB180,365,399,411,418,438,69
8,598,925,998,918
20170 IFM=2THEN20200
20175 IFX=12THEN20005
20180 GOTO20100
20200 PRINT' TAB(11)'SPRITE MULTICOLO
R':GOSUB2708
20205 ONX GOSUB2198,365,399,2698,418,199
8,698,598,925,998,918
20210 IFM=1THEN20100
20215 IFX=12THEN20005
20220 GOTO20200
34997 REM *****
34998 REM * L.M. MUSICA (INTERRUPT) *
34999 REM *****
40000 DATA165,251,72,165,252,72,173,254,
194,133,251,173,255,194,133,252,104,141
40010 DATA255,194,104,141,254,194,206,25
3,194,208,69,169,32,141,4,212,165,251
40020 DATA205,249,194,208,10,165,252,205
,250,194,208,3,76,17,201,160,0,177
```



```

40030 DATA251,208,10,200,200,177,251,141
,253,194,76,87,200,141,0,212,200,177
40040 DATA251,141,1,212,200,177,251,141,
253,194,169,33,141,4,212,24,165,251
40050 DATA105,3,133,251,144,2,230,252,20
6,252,194,208,69,169,32,141,11,212
40060 DATA165,251,205,249,194,208,10,165
,252,205,250,194,208,3,76,17,201,160
40070 DATA0,177,251,208,10,200,200,177,2
51,141,252,194,76,161,200,141,7,212
40080 DATA200,177,251,141,8,212,200,177,
251,141,252,194,169,33,141,11,212,24
40090 DATA165,251,105,3,133,251,144,2,23
0,252,206,251,194,208,69,169,32,141
40100 DATA18,212,165,251,205,249,194,208
,10,165,252,205,250,194,208,3,76,17
40110 DATA201,160,0,177,251,208,10,200,2
00,177,251,141,251,194,76,235,200,141
40120 DATA14,212,200,177,251,141,15,212,
200,177,251,141,251,194,169,33,141,18
40130 DATA212,24,165,251,105,3,133,251,1
44,2,230,252,165,251,72,165,252,72
40140 DATA173,254,194,133,251,173,255,19
4,133,252,104,141,255,194,104,141,254
40150 DATA194,76,64,201,173,248,194,240,
24,169,1,141,253,194,141,252,194,141,251
40160 DATA194,173,246,194,133,251,173,24
7,194,133,252,76,246,200,169,49,141
40170 DATA20,3,169,234,141,21,3,169,0,14
1,24,212,76,246,200
40190 REM*****
40191 REM* L.M. SPOSTAMENTO'MOB'EGIZ. *
40192 REM*****
40200 DATA173,255,206,240,118,173,254,20
6,240,6,206,254,206,76,49,234,169,8
40210 DATA141,254,206,165,253,205,250,20
6,208,7,165,254,205,251,206,240,20,165
40220 DATA253,205,247,206,208,7,165,254,
205,248,206,240,59,206,255,206,206,255
40230 DATA206,160,128,24,165,253,105,128
,133,253,144,2,230,254,201,254,208,16
40240 DATA165,254,201,159,208,10,173,250
,206,133,253,173,251,206,133,254,177,253
40250 DATA153,64,3,136,16,248,173,255,20
6,141,0,208,141,2,208,76,49,234
40260 DATA206,255,206,206,255,206,206,25
5,206,206,255,206,76,119,201,169,49,141
40270 DATA15,201,169,234,141,16,201,169,
0,141,21,208,76,49,234
40490 REM*****
40491 REM* NOTE MUSICA 1 (INTERRUPT) *
40492 REM*****
40500 DATA30,25,30,0,0,30,0,0,30,135,33,
30,31,21,30,97,8,60
40510 DATA135,33,15,31,21,15,162,37,15,9
6,22,15,62,42,30,30,25,30
40520 DATA97,8,60,135,33,30,31,21,30,60,
50,60,209,18,30,233,7,30
40530 DATA30,25,30,71,6,30,62,42,30,30,2
5,60,97,8,60,0,0,15
40540 DATA62,42,15,193,44,30,49,28,60,15
2,5,60,60,50,15,193,44,15
40550 DATA62,42,15,30,25,60,97,8,60,193,
44,15,60,50,30,162,37,15
40560 DATA30,25,60,71,6,60,135,33,15,162

```

```

,37,15,62,42,15,162,37,30
40570 DATA165,31,15,0,0,60,49,28,15,30,2
5,30,30,25,15,96,22,15
40580 DATA135,33,30,31,21,30,97,8,60,135
,33,15,31,21,15,162,37,15
40590 DATA96,22,15,62,42,30,30,25,30,97,
8,60,135,33,30,31,21,30
40600 DATA60,50,60,209,18,30,233,7,30,30
,25,30,71,6,30,62,42,30
40610 DATA30,25,60,97,8,30,0,0,15,0,0,30
,62,42,15,193,44,15
40620 DATA49,28,30,152,5,60,60,50,15,62,
42,15,30,25,30,193,44,15
40630 DATA162,37,30,96,22,60,71,6,60,0,0
,15,135,33,15,135,33,60
40640 DATA31,21,60,97,8,60,0,0,60,0,0,60
,0,0,60
40690 REM*****
40691 REM* NOTE MUSICA 2 (INTERRUPT) *
40692 REM*****
40700 DATA30,25,30,31,21,60,97,8,60,30,2
5,15,49,28,15,30,25,60,31,21,60,97,8
40710 DATA60,62,42,30,31,21,60,97,8,60,6
2,42,15,193,44,15,62,42,60,31,21,60,97,8
40720 DATA60,62,42,30,30,25,60,97,8,60,6
0,50,15,193,44,15,62,42,60,30,25,60,97
40730 DATA8,60,162,37,30,30,25,30,233,7,
30,62,42,15,96,22,30,71,6,30,162,37,15
40740 DATA135,33,60,31,21,60,97,8,60
40750 DATA30,25,30,31,21,60,97,8,60,30,2
5,15,49,28,15,30,25,60,31,21,60,97,8
40760 DATA60,62,42,30,31,21,60,97,8,60,6
2,42,15,193,44,15,62,42,60,31,21,60,97,8
40770 DATA60,62,42,30,30,25,60,97,8,60,6
0,50,15,193,44,15,62,42,60,30,25,60,97
40780 DATA8,60,162,37,30,30,25,30,233,7,
30,62,42,15,96,22,30,71,6,30,162,37,15
40790 DATA135,33,60,31,21,60,97,8,60
40800 DATA62,42,15,156,26,60,233,7,60,62
,42,15,62,42,30,165,31,60,156,26,60,71
40810 DATA5,60,135,33,15,49,28,60,12,7,6
0,162,37,15,135,33,15,165,31,15,49,28
40820 DATA60,31,21,60,12,7,60,135,33,15,
49,28,60,12,7,60,135,33,15,165,31,15
40830 DATA135,33,15,162,37,60,165,31,60,
71,6,60,30,25,15,31,21,60,97,8,60,30
40840 DATA25,15,62,42,30,193,44,60,209,1
8,60,233,7,60,30,25,30,31,21,60,97,8
40850 DATA60,30,25,15,49,28,15,30,25,60,
31,21,60,97,8,60,62,42,30,31,21,60,97
40860 DATA8,60,62,42,15,193,44,15,62,42,
60,30,25,60,97,8,60,62,42,30,30,25,60
40870 DATA97,8,60,60,50,15,193,44,15,62,
42,30,30,25,60,97,8,60,0,0,15,60,50,15
40880 DATA162,37,15,96,22,30,233,7,15,60
,50,15,71,6,15,135,33,15,31,21,30,97,8
40890 DATA15,0,0,15,0,0,15,0,0,60,0,0,60
,0,0,60
40897 REM *****
40898 REM * DATA MUSICA 3 (INTERRUPT) *
40899 REM *****
40900 DATA223,39,8,0,0,32,0,0,32,134,35,
8,181,23,48,107,47,8,223,29,8,0,0
40910 DATA8,0,0,8,107,47,16,223,29,16,22
3,39,8,0,0,32,0,0,32,134,35,8,181

```



```
40920 DATA23,48,107,47,8,223,29,8,0,0,8,
0,0,8,107,47,16,223,29,16,223,39
40930 DATA8,0,0,32,0,0,32,134,35,8,181,2
3,32,107,47,16,223,29,16,239,19,32
40940 DATA0,0,16,0,0,16,107,47,16,223,29
,16,195,17,48,0,0,16,0,0,16,193,44,8
40950 DATA165,31,8,0,0,8,0,0,8,193,44,16
,165,31,16,223,39,8,0,0,32,0,0,32
40960 DATA134,35,8,195,17,48,193,44,8,16
5,31,8,0,0,8,0,0,8,193,44,16
40970 DATA165,31,16,223,39,8,0,0,32,0,0,
32,134,35,8,195,17,48,193,44,8
40980 DATA165,31,8,0,0,8,0,0,8,193,44,16
,165,31,16,223,39,8,0,0,32,0,0,32
40990 DATA134,35,8,195,17,32,193,44,16,1
65,31,16,239,19,32,0,0,16,0,0,16
41000 DATA193,44,16,165,31,16,181,23,48,
0,0,16,0,0,16,107,47,8,223,29,8
41010 DATA0,0,8,0,0,8,107,47,16,223,29,1
6
41030 DATA223,39,8,0,0,32,0,0,32,134,35,
8,190,59,48,107,47,8,223,29,8
41040 DATA0,0,8,0,0,8,107,47,16,223,29,1
6,223,39,8,0,0,32,0,0,32,134,35,8
41050 DATA190,59,48,107,47,8,223,29,8,0,
0,8,0,0,8,107,47,16,223,29,16
41051 DATA223,39,8,0,0,32,0,0,32,134,35,
8
41052 DATA190,59,32,107,47,16,223,29,16,
12,71,32,0,0,16,0,0,16,107,47,16
41060 DATA223,29,16,191,79,48,0,0,16,0,0
,16,193,44,8,165,31,8,0,0,8,0,0,8
41070 DATA193,44,16,165,31,16,223,39,8,0,
0,32,0,0,32,134,35,8,191,79,48
41080 DATA193,44,8,165,31,8,0,0,8,0,0,8,
193,44,16,165,31,16,223,39,8
41090 DATA0,0,32,0,0,32,134,35,8,191,79,
48,193,44,8,165,31,8,0,0,8,0,0,8
41100 DATA193,44,16,165,31,16,223,39,8,0,
0,32,0,0,32,134,35,8,191,79,32
41110 DATA193,44,16,165,31,16,12,71,32,0,
0,16,0,0,16,193,44,16,165,31,16
41120 DATA190,59,48,0,0,16,0,0,16,107,47
,8,223,29,8,0,0,8,0,0,8,107,47,16
41130 DATA223,29,16,0,0,16,0,0,16,0,0,16
```

```
48997 REM *****
48998 REM * DATA SPRITE EGIZIANO *
48999 REM *****
49000 DATA2,21,0,0,170,0,0,21,0,0,62,128
,0,253,64,60
49010 DATA62,128,12,61,80,12,254,160,12,
12,0,12,12,0,13,238
49020 DATA220,13,251,220,0,63,12,0,63,12
,0,63,12,0,63,12
49030 DATA0,63,15,0,63,0,0,63,0,0,63,0,0
,63,0,0:REM DATI DEL 1 SPRITE
49040 DATA0,42,0,0,25,0,0,25,0,0,89,64,0
,89,64,1
49050 DATA85,80,0,12,0,0,60,0,0,60,0,0,2
52,0,0,204
49060 DATA0,0,204,0,3,204,0,0,204,0,0,25
2,0,0,60,0
49070 DATA0,14,0,0,14,0,0,14,0,0,190,0,2
,170,0,0:REM DATI DEL 2 SPRITE
49080 DATA2,21,0,0,170,0,0,21,0,0,62,128
,0,253,64,240
```

```
49090 DATA62,128,48,61,80,48,254,160,48,
12,0,48,60,0,55,187
49100 DATA112,55,239,124,0,252,12,0,252,
12,0,252,12,0,252,12
49110 DATA0,252,15,0,252,0,0,252,0,0,252
,0,0,252,0,0:REM DATI DEL 3 SPRITE
49120 DATA0,168,0,0,100,0,0,100,0,1,101,
0,1,101,0,5
49130 DATA85,64,0,240,0,0,240,0,3,240,0,
3,48,0,15,48
49140 DATA0,12,48,0,12,60,0,12,12,0,12,1
2,0,140,12,0
49150 DATA190,12,0,170,12,0,0,14,0,0,186
,0,2,168,0,0:REM DATI DEL 4 SPRITE
49160 DATA8,84,0,2,168,0,0,84,0,0,250,0,
3,245,0,240
49170 DATA250,0,48,245,64,51,250,128,48,
48,0,48,60,0,61,238
49180 DATA220,13,251,220,0,63,12,0,63,12
,0,63,12,0,63,12
49190 DATA0,63,15,0,63,0,0,63,0,0,63,0,0
,63,0,0:REM DATI DEL 5 SPRITE
49200 DATA0,42,0,0,25,0,0,25,0,0,89,64,0
,89,64,1
49210 DATA85,80,0,51,0,0,51,0,0,51,0,0,5
1,0,0,51
49220 DATA0,0,51,0,0,51,192,0,48,192,0,4
8,192,0,48,192
49230 DATA0,48,192,0,48,224,0,51,224,2,2
50,160,10,170,0,0:REM DATI DEL 6 SPRITE
49997 REM *****
49998 REM * DATA L.M. CORNICE ROTANTE *
49999 REM *****
50000 DATA169,255,133,251,133,253,169,3,
133,252,169,215,133,254,173,39,4,141
50010 DATA0,192,173,39,216,141,1,192,160
,39,177,251,170,177,253,200,145,253
50020 DATA138,145,251,136,136,208,241,16
9,0,133,251,133,253,169,4,133,252,169
50030 DATA216,133,254,162,24,160,40,177,
251,72,177,253,160,0,145,253,104,145
50040 DATA251,24,165,251,105,40,133,251,
144,2,230,252,24,165,253,105,40,133
50050 DATA253,144,2,230,254,202,208,217,
169,192,133,251,133,253,169,7,133,252
50060 DATA169,219,133,254,160,1,177,251,
72,177,253,136,145,253,104,145,251,200
50070 DATA200,192,40,208,239,169,191,133
,251,133,253,169,7,133,252,169,219,133
50080 DATA254,162,23,160,0,177,251,72,17
7,253,160,40,145,253,104,145,251,56
50090 DATA165,251,233,40,133,251,176,2,1
98,252,56,165,253,233,40,133,253,176
50100 DATA2,198,254,202,208,217,173,0,19
2,141,79,4,173,1,192,141,79,216,96
```

READY.

ATTENZIONE!

Per un diverso interfacciamento con la stampante il segno ' (apostrofo) corrisponde agli apici ". Fate attenzione nel digitare il programma.
10 PRINT '□' DIVENTA: 10 PRINT"□"

CONCORSO STRIKE!

AUT. MIN. N. 4/267178 del 25/7/84



Ecco di seguito l'elenco di tutti i programmi pubblicati per il concorso:

ZX-SPECTRUM

- 1) LINEE ZERO di Saverio Cantone - ROMA
- 2) TOMBOLA di Tommaso Calò - ROMA
- 3) IL 13 MINUTO PER MINUTO di Fabio Travi - ROMA
- 4) IL LOTTO di Elio Girone - MARANO (NA)

TI - 99/4A

- 1) LE MINE di Gianni Martini - COLDIROBI (IM)
- 2) VINCI QUATTRO di Davide Inbeni - MODENA
- 3) DATA BASE di Luciano Marchetto - NICHELINO (TO)
- 4) MISSION VENUS di Roldano Zappalà - S. GIUSTINO (PG)
- 5) SNAKE di Lorenzo Sciucca - ROMA
- 6) OLIMPIADI di Filippo Pergola - ROMA
- 7) OMEGA LANDER di Fabio Torti - TORTONA (AL)
- 8) MR. JUMP di Enrico Combi - MODENA

SEGA SC-3000

- 1) PAROLIAMO di Ettore Fruscoloni - ROMA
- 2) ASTRO WAR di Saverio Forestiero - ROMA
- 3) PIANOFORTE di Enrico Grandis - BATTAGLIA T. (PD)
- 4) TIRO A VOLO di Simone Grandis - BATTAGLIA T. (PD)
- 5) UOVA SPAZIALI di Arturo Cucè - MESSINA
- 6) BATTAGLIA AEREA di Enzo Masci - ROMA
- 7) SPIDER di Franco Albanelli - CASTELGOMBERTO (VI)
- 8) ARCHER di Adamo Antonellini - FUSIGNANO (RA)
- 9) REAL GOLF di Mario Frustaci - AREZZO
- 10) SUPER MASTER MIND di Fabio Canta - OSTIA (ROMA)

SHARP MZ-700

- 1) ELECTRONIC MASTERMIND di Luca Bonguerrieri - GENOVA
- 2) SALISCENDI di Stefania Palombi - ROMA
- 3) SLOT MACHINE di Dario Letizia - NAPOLI
- 4) BOXE di Massimo Moras - PORDENONE
- 5) JAZZI di Marco Marzano - ROMA
- 6) IL CASTELLO DI WIZ di Alessandro Arposio - MILANO
- 7) CILIEGIE di Mauro Scarsoglio - ALESSANDRIA
- 8) BIORITMI di Costantino Costanzi - MONZA (MI)
- 9) SCI ALPINO di Alessandro Altemani - CASTEL D'AZZANO (VR)
- 10) DISPERSIONI TERMICHE di Giacomo Monnanni - AREZZO

ORIC 1

- 1) BREAK OUT di Marco Belli - ROMA
- 2) SALTA LA RANA di Fabrizio Cuminetti - ROMA

MPF II

- 1) CONTRAEREA di Alfredo Broussard - DOSSOBUONO (VR)
- 2) MASTERMIND di Massimo Marinelli - ROMA

CBM 64

- 1) SINT CODER di Marco Giusti - PISA
- 2) FLIGHT PLAN di Marco Olivotto - ROVERETO (TN)
- 3) ECONOMIA CASA di Dino Mazzini - CASALECCHIO DI RENO (BO)
- 4) PITAGORA di Antonio Procida - CRECCHIO (CH)
- 5) EASY SPRITE di Moretto - MESTRE (VE)
- 6) WIMBLEDON '84 di Bruno Brunelli - FOLIGNO (PG)
- 7) BILANCIO di Luigi Rusticali - RUSSI (RA)

VIC 20

- 1) IL SERPENTE di Simone Mariani - OSTRA (AN) (*)
- 2) CAPITALI EUROPEE di F.P. De Gisi - ATRIPALDA (AV)
- 3) FISHERMAN di Luca Nardini - VELLETRI (ROMA) (*)
- 4) BIORITMI di Paolo de Maria - VENARIA (TO)
- 5) TASTI FUNZIONE di Stefano Ronconi - NOVATE MILANESE (MI)
- 6) FORMULA 1 di Angelo Ariaudo - MILANO
- 7) GHOST di Giulio D'Urso - FIRENZE
- 8) CRAZY WORD di Giuseppe Burgio - MILANO
- 9) BRUCE di Farinelli/Cervellara - BOLOGNA
- 10) SPACCAMATTONI di Roberto D'Acunto - SCAURI (LT)

ZX - 81

- 1) SWIMMING POOL di Francesco Mondelli - OSTIA (ROMA)
- 2) I POMPIERI di Marco Michieli - ROMA
- 3) POETA SINCLAIR di Mario Caracciolo - TORINO
- 4) U.S.A. di Raffaello Danti - SESTO FIORENTINO (FI)
- 5) CONDOMINIO di Loris Gerini - FALCONARA MARITTIMA (AN)

(*) ATTENZIONE: Avvertiamo i lettori che i programmi «FISHERMAN» e il «SERPENTE» pubblicati sul n. 2/85 di LIST, in quanto ritenuti, in un primo momento, meritevoli di rientrare nella rosa delle nove categorie, sono stati successivamente ESCLUSI dal concorso essendo risultati in violazione al regolamento dello stesso concorso, non originali, in quanto pubblicati con identica stesura, il primo sul n. 11/84 della rivista «PAPERSOFT» e il secondo sul n. 2/84 della rivista «LOAD 'N'RUN».

Pertanto le cartoline voto relative a tali programmi NON SARANNO RITENUTE VALIDE ai fini del sorteggio finale.

La Giuria del
Concorso STRIKE

CARTOLINA VOTO CONCORSO

STRIKE!

AUT. MIN. N. 4/267178 del 25/7/84

TITOLO DEL PROGRAMMA

nome

cognome

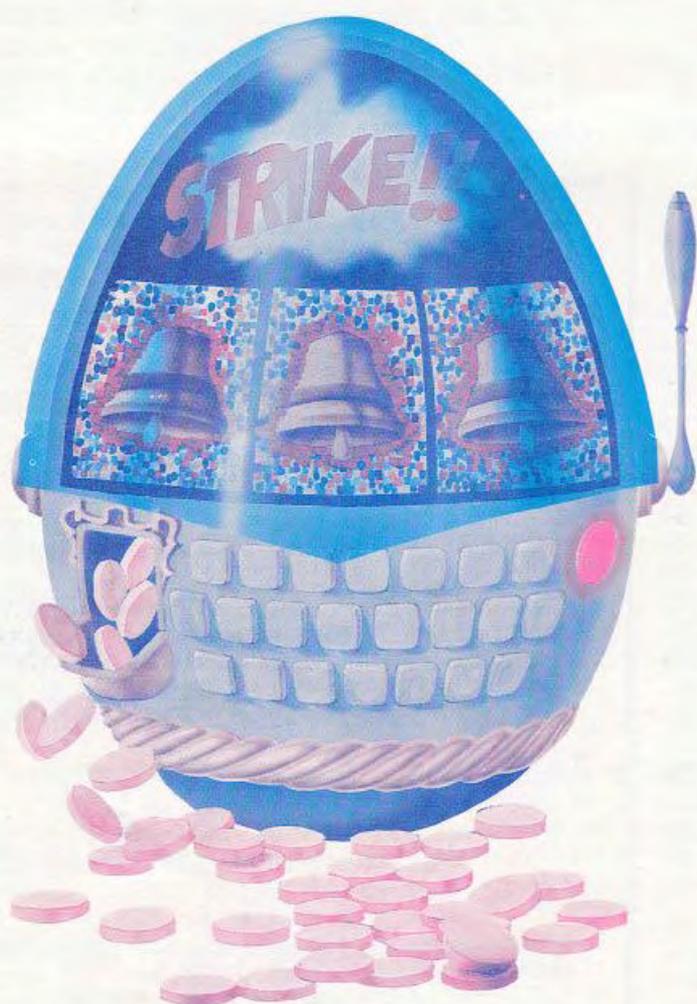
COMPUTER

via

AUTORE

città

cap. età



MILIONARIA

Inserire in busta chiusa

Spett. LIST

programmi per il tuo home computer

Casella Postale 4092

00182 ROMA APPIO

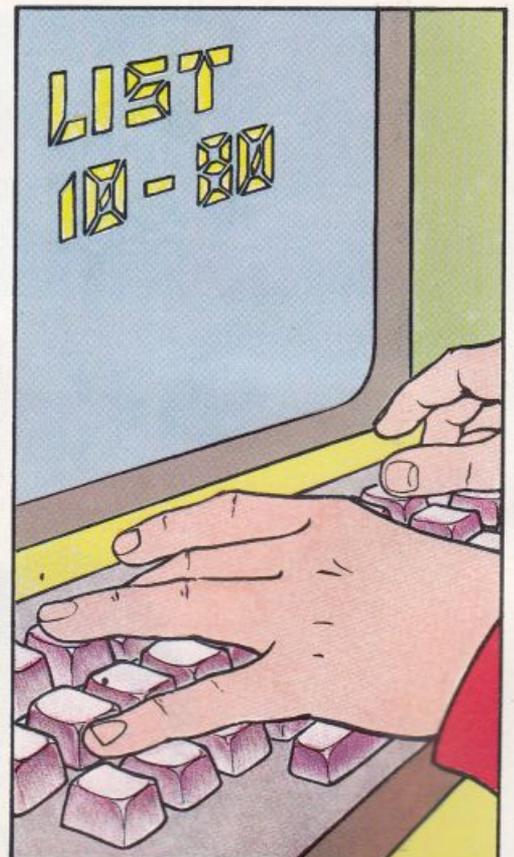
L'AMICO SEGRETO





EH!
MA COSA
FA ?!!
PASSA TUTTO
VELOCISSIMO!

IL LISTATO E' LUNGO, NON
ENTRA TUTTO INTERO NEL
VIDEO. QUINDI DEVI CHIE-
DERGLIELO UNA PARTE
PER VOLTA...
...VEDIAMO, PER ESEMPIO,
LE PRIME OTTO ISTRUZIO-
NI ...



ORA VA MEGLIO.
BISOGNA PROPRIO
SAPERE COME PRENDERLO
QUESTO COMPUTER!

```
10 REM FACCIAMO I CONTI
20 PRINT "CLEAR":PRINT
  "DIMMI UN NUMERO"
30 INPUT X
40 PRINT "NE VOGLIO UN
  ALTRO"
50 INPUT Y
60 PRINT "CLEAR";PRINT
  "E ORA CHE CI FACCIAMO
  CON QUESTI NUMERI?"
70 PRINT "1 LI SOMMIAMO"
80 PRINT "2 LI SOTTRAIA
  MO"
```

COSE' QUEL "CLEAR", DOPO
PRINT, ALLA RIGA 20 ?



IN ALCUNI COMPUTERS SI
USA *CLS* (CLEAR THE SCREEN=
PULISCI IL VIDEO), IN ALTRI
CALL CLEAR, IN ALTRI ANCO-
RA SI DIGITA :*PRINT "{*, SI
PREMONO INSIEME *SHIFT* E
CLEAR/HOME}", NEL LISTATO
COMPARE :*PRINT "☐"*, E,
QUANDO IL PROGRAMMA GIRA,
SCOMPARE TUTTO DALLO
SCHERMO.



NEL DIAGRAM-
MA DI FLUS-
SO GLI ORDINI
START E
END STAN-
NO IN RET-
TANGOLI AR-
ROTONDATI...

PAOLO!

GLI ORDINI NORMALI IN RETTANGOLI... NEI ROMBI QUANDO IL COMPUTER DEVE SCEGLIERE TRA PIU' ALTERNATIVE.

PAOLO, NOI USCIAMO. TORNEREMO TARDI. MI RACCOMANDO, STUDIA LA STORIA. ARRIVEDERCI, RAGAZZI!

BUONA SERA!

VA BENE, CIAO!

```

90 PRINT "3 LI MOLTIPLI
    CHIAMO"
100PRINT "4 LI DIVIDIA
    MO"
110 INPUT Z
120 REM E' URA LA SCELTA
    TRA LE 4 POSSIBILITA'
130 ON Z GOTO 150,160,
    170,180
140 REM CALCOLIAMO
150 PRINT X;"+";Y;"=";
    X+Y:GO TO 190
160 PRINT X;"-";Y;"=";
    X-Y:GO TO 190

```

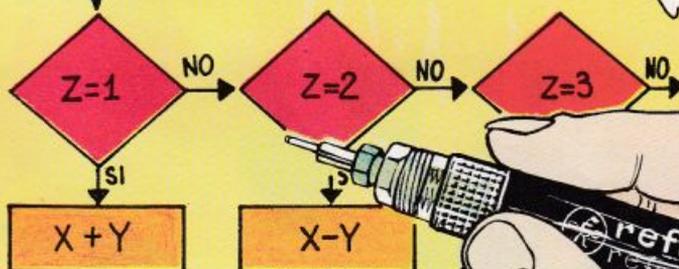
A.M. Mazzi + M. Lorenzi '85

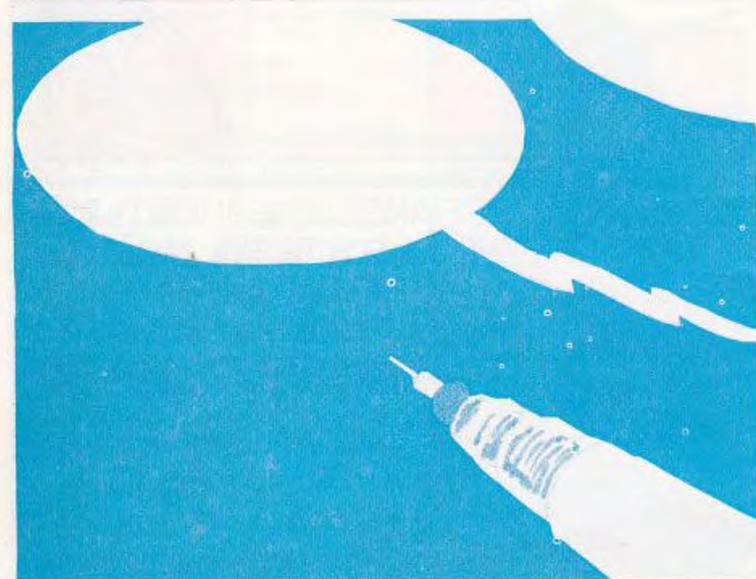
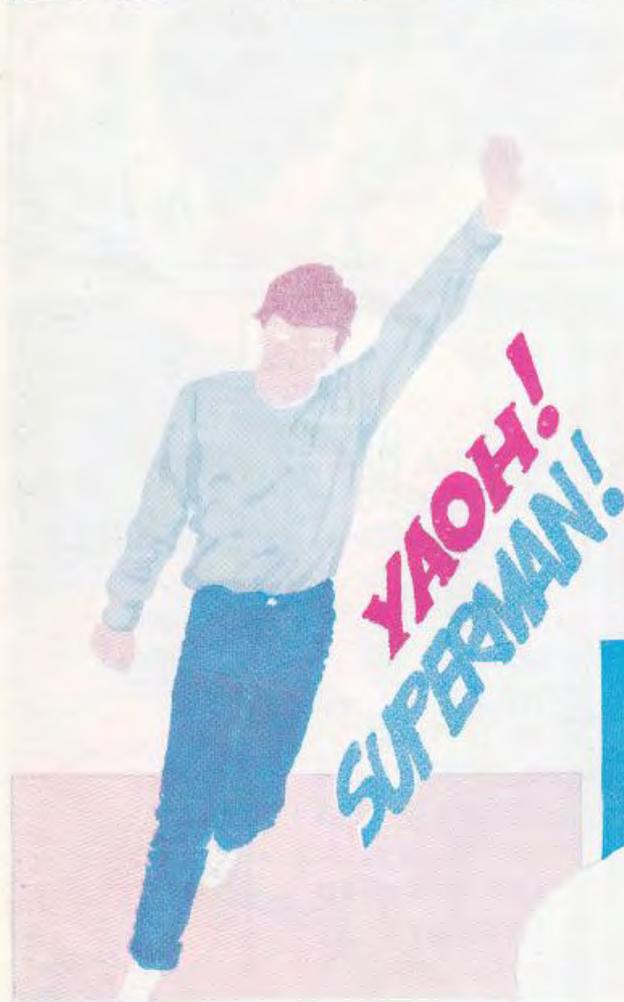
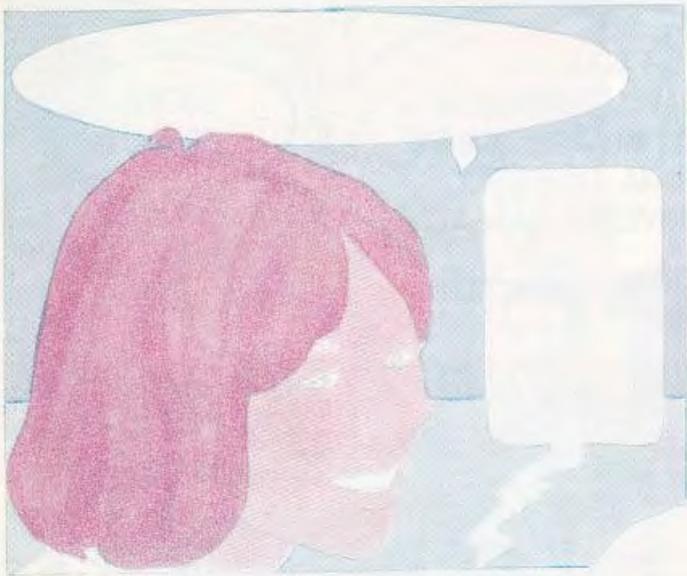
ALLA RIGA 110 VI HA CHIESTO DI DARE UN VALORE ALLA VARIABILE Z, DA SCEGLIERE TRA 1,2,3 O 4. SE DATE 1 VA ALLA RIGA 150, SE DATE 2 ALLA 160, SE 3 ALLA 170, E SE 4 ANDATE ALLA 180.

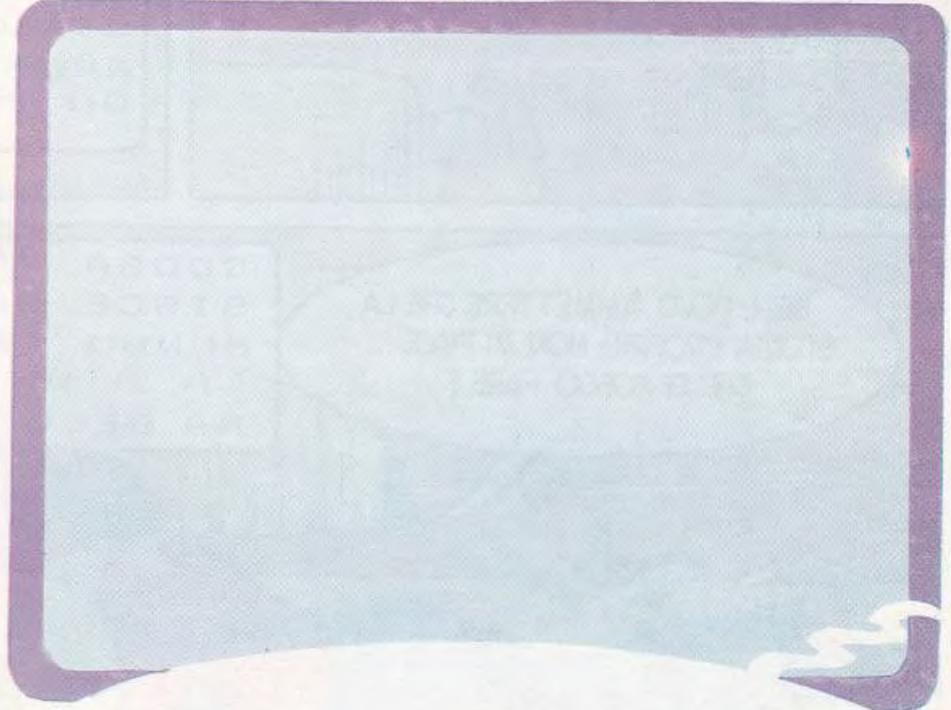
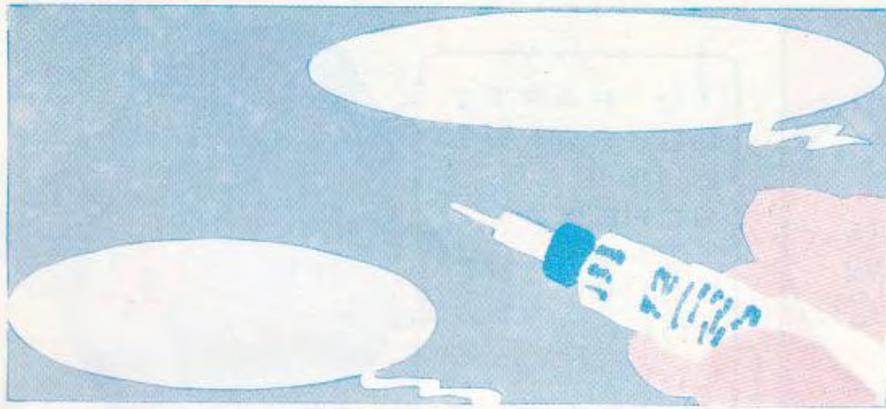
E QUESTA E' LA SECONDA PARTE DEL LISTATO. ALLA RIGA 130 C'E' `ON Z GO TO` CHE ANCORA NON CONOSCETE.

NEL DdF, SOTTO AL RETTANGOLONE CI SONO 4 ROMBI, CHE RAPPRESENTANO LA SCELTA TRA I 4 VALORI DELLA Z. SE VOGLIAMO SOTTRARRE, IL COMPUTER COMINCIA CON Z=1, LA RISPOSTA E' **NO**...

... ALLORA VA A DESTRA, AL ROMBO Z=2 E LA RISPOSTA E' **SI**'.



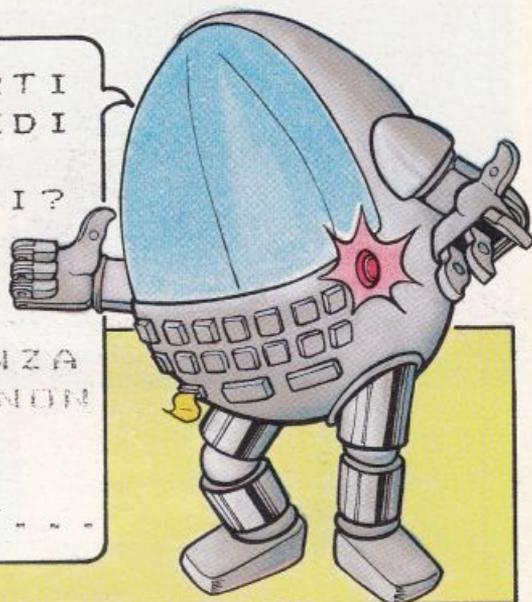






EHI!
ORA COMINCI
ANCHE TU COME LA
MAMMA A FARMI
LE PREDICHE?

IO FARTI
LE PREDI
CHE!
SCHERZI?
TI DI
CEVO
SOLO
CHE SENZA
INPUT NON
SI PUO
AVERE
OUTPUT...

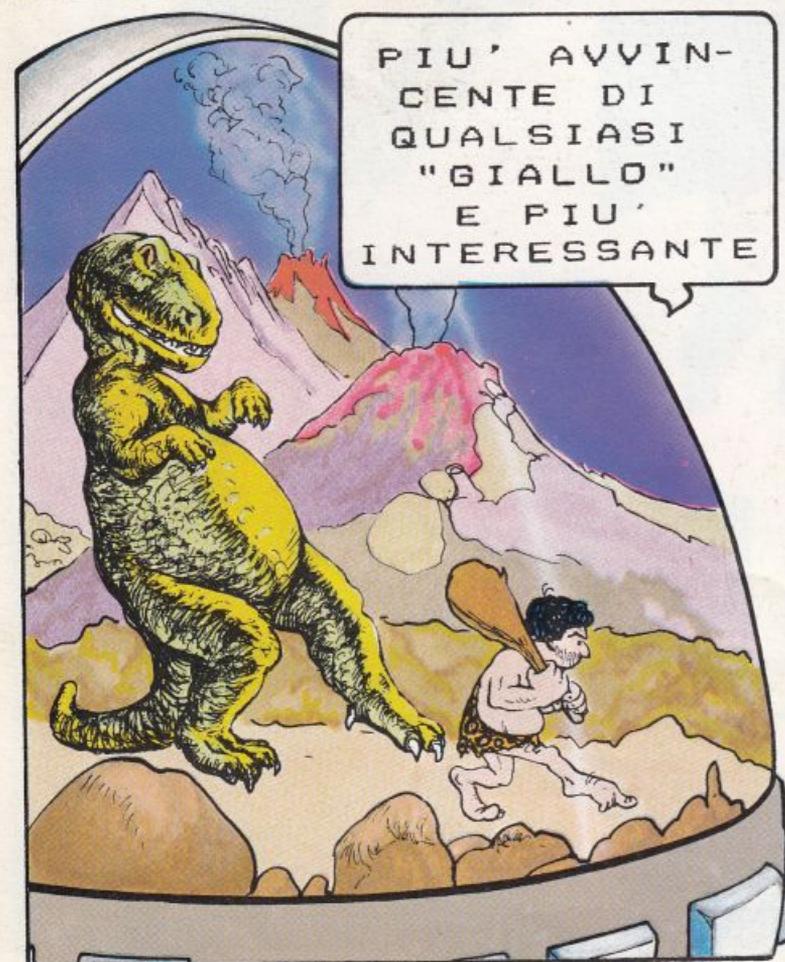


Manni+Comi '85



BEH, DEVO AMMETTERE CHE LA
STORIA PROPRIO NON MI PIACE...
CHE CI POSSO FARE?

COOSA ??? NON TI INCURIO-
SISCE SAPERE COME ERA IL
MONDO TANTI SECOLI FA?
LA STORIA E' L'AVVENTURA
DELL'UOMO,



PIU' AVVIN-
CENTE DI
QUALSIASI
"GIALLO"
E PIU'
INTERESSANTE



DI UN ROMANZO! MA FOR-
SE IL TUO PROFESSORE
NON SA RENDERTELA PIA-
CEVOLE, VOGLIO RACCON-
TARTI IO QUALCOSA...

CONTINUA.

**abbonati
potrai godere del
favoloso sconto del**

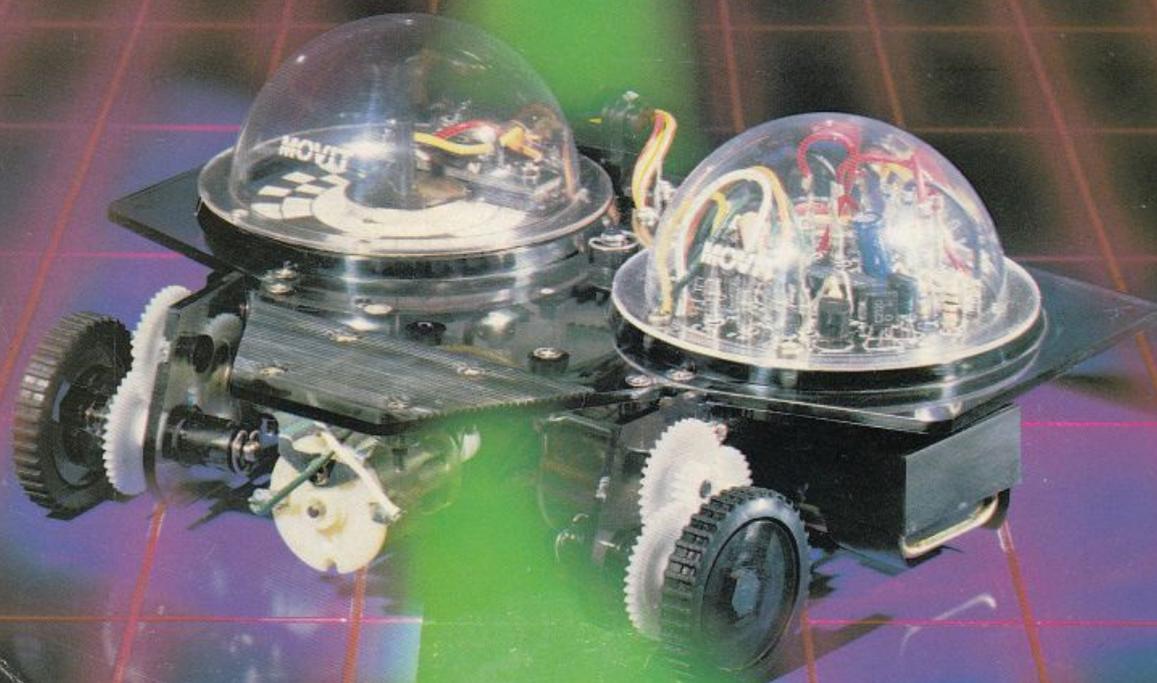
20%

su 11 numeri annui

VALIDO FINO AL 30 GIUGNO 1985

**non perdere
d'occhio il prossimo
numero,
in regalo.....**

MOVIT MOVIT MOVIT



I FANTASTICI "ROBOTS" IN KIT