



VIC

da zero

di Tommaso Pantuso

+64

C64: note per la lettura del generatore di caratteri

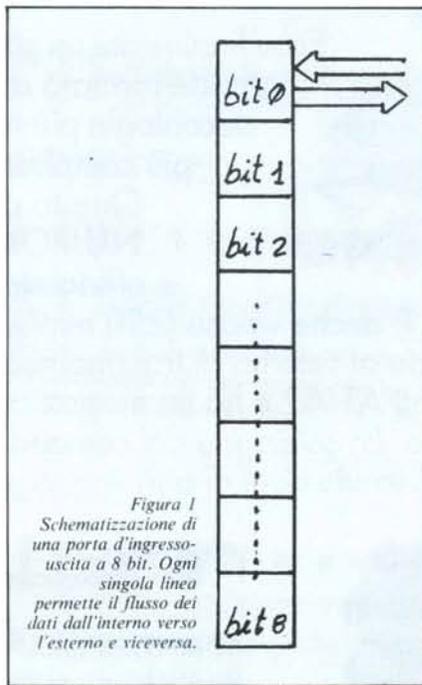
Riprendiamo il discorso sul generatore di caratteri del Commodore 64 e vediamo questa volta tutte le operazioni da compiere per leggere il contenuto al fine di produrne una copia in Ram. Rimarrà poi solo da esaminare il modo per mandare il sistema a leggere un eventuale nuovo set di caratteri lì dove abbiamo deciso di porlo.

Le porte d'Ingresso/Uscita

Naturalmente non ci riferiamo a quelle di casa nostra. Di porte d'ingresso/uscita abbiamo parlato parecchio tempo fa in questa stessa rubrica (n. 25, dic. 83) ma visto che da quel di molta acqua è passata sotto i ponti, è forse il caso di illustrare in proposito qualche concetto fondamentale proponendoci di parlare tra un po' di tempo delle porte d'I/O del 64 in maniera più consistente.

Intanto diciamo che questi brevi cenni sulle porte d'I/O ci sono utili per capire bene il modo in cui vengono effettuate alcune manipolazioni del sistema al fine di produrre un certo tipo di interazione con la memoria Ram o Rom. Non spaventatevi per come ci siamo espressi: tra breve non avrete più dubbi.

Una porta è semplicemente un oggetto da cui entra o esce qualcosa. Se il concetto sembra facile, è perché in effetti lo è veramente! Nei casi che ci interessano più da vicino, in un computer, attraverso una porta passano dei dati sotto forma di infor-



mazioni binarie (insiemi di 0 e di 1). Nella sua forma più generale, una porta preleva informazioni dall'esterno o manda informazioni verso l'esterno, intendendo per esterno tutto il resto dell'ambiente ad esclusione della porta. In altre parole, il flusso di dati può avvenire in maniera bidirezionale: è in questo caso che si parla di porte d'I/O, cioè d'Input/Output. Potete osservare la schematizzazione di una porta a otto bit nella figura 1. Il numero di bit da cui è composta la porta è rappresentato dal numero di informazioni elementari che possono transitare contemporaneamente attraverso esse. Naturalmente, attraverso ciascuna delle linee elementari della porta, l'informazione può transitare, nello stesso istante, in una sola direzione. In altre parole, una singola linea, non può essere nello stesso istante sia un'ingresso che un'uscita. Pertanto deve necessariamente esistere un sistema che permetta all'utente di portare ogni singola linea in una delle due situazioni desiderate.

Il sistema impiegato normalmente è molto semplice. Praticamente una porta d'I/O viene completamente manipolata utilizzando due registri dedicati allo scopo. Facciamo un esempio. Nella figura 2 è schematizzata una porta d'ingresso/uscita e i due registri attraverso i quali vengono comandate le operazioni che essa deve svolgere. Il registro D prende il nome di Registro di Direzione dei Dati mentre il registro R è il registro d'ingresso/uscita. La funzione di ciascun registro è facilmente intuibile: tramite D stabiliremo il ruolo di ciascuna linea mentre in R avverrà il vero e proprio transito dei dati: in altre parole, in R porremo un'informazione da mandare verso l'uscita o preleveremo da R quella proveniente dall'esterno. Per essere più chiari riferiamoci ad un'altra figura, la 3. Per prima cosa definiamo bene il modo in cui agisce il registro D: ciascun bit di tale registro può essere 0 oppure 1. In corrispondenza ad uno 0 la corrispondente linea di R sarà predisposta come ingresso mentre in presenza di un 1 la stessa linea sarà un'uscita. Una situazione abbastanza chiara è presente nella figura precedentemente citata dove vengono ben stabiliti gli ingressi e le uscite a seconda del contenuto di D.

Nota. Naturalmente la condizione discriminante potrebbe essere anche quella opposta, cioè uno 0 stabilisce un'uscita e un 1 un ingresso, ma ciò è irrilevante ai fini del nostro discorso.

Definiamo ora la funzione di R. Essa è ancora più semplice di quella affidata a D. Se proviamo a configurare le linee della porta tutte come uscita, ponendo in D tutti 1 e ponendo in R ad esempio la parola 10010101 (fig. 4a), questa stessa parola la ritroveremo sulle linee fisiche di uscita della porta — ad esempio, come normalmente avviene, i pin di un circuito integrato —. Lo stesso per il caso opposto cioè, configurando le linee come ingressi, una parola posta sulle linee fisiche d'ingresso provoca una immagine in R (fig. 4b). Naturalmente sono possibili combinazioni miste della porta, cioè alcune linee possono essere configurate come ingresso ed altre come uscita.

Sulle porte non abbiamo bisogno di dire altro. Andiamo allora a vedere dove viene applicato ciò che abbiamo appreso in questa prima parte.

La porta I/O del 6510

Una porta d'I/O molto importante appartiene al cuore del C 64, cioè al microprocessore 6510. Se date uno sguardo alla figura 5, dove riportiamo una parte dello schema elettrico del computer in questione, potrete presto individuarla essendo contrassegnata con la lettera P, numerata da 0 a 5; si tratta infatti di una porta a 6 bit. *Errata Corrige.* Visto che ci troviamo in argomento, ne approfittiamo per correggere un'impresione presente nello scorso articolo dove in chiusura abbiamo accennato sommariamente a tale porta illustrando brevemente le funzioni svolte da ciascuna delle sue linee.

L'impresione consiste nell'aver detto che si trattava di una porta a 5 bit. Nonostante questo, i lettori attenti avranno notato che le funzioni descritte erano illustrate a partire dal bit di posto 0 fino a quello di posto 5, cioè in totale sono stati presi in considerazione 6 bit quindi, ai fini pratici, l'errore non sussiste.

Tale porta è contraddistinta, come di consueto, da un Registro di Direzione dei Dati (RDD) e da un Registro d'Ingresso/Uscita (RIO). L'RDD è mappato in memoria in pagina zero, precisamente nella locazione 0, mentre il RIO compare nella locazione 1. Quindi, in linea teorica, possono essere effettuate delle letture dei contenuti di tali registri e delle scritture in essi mediante delle semplici Peek e Poke. In pratica, per particolari caratteristiche di

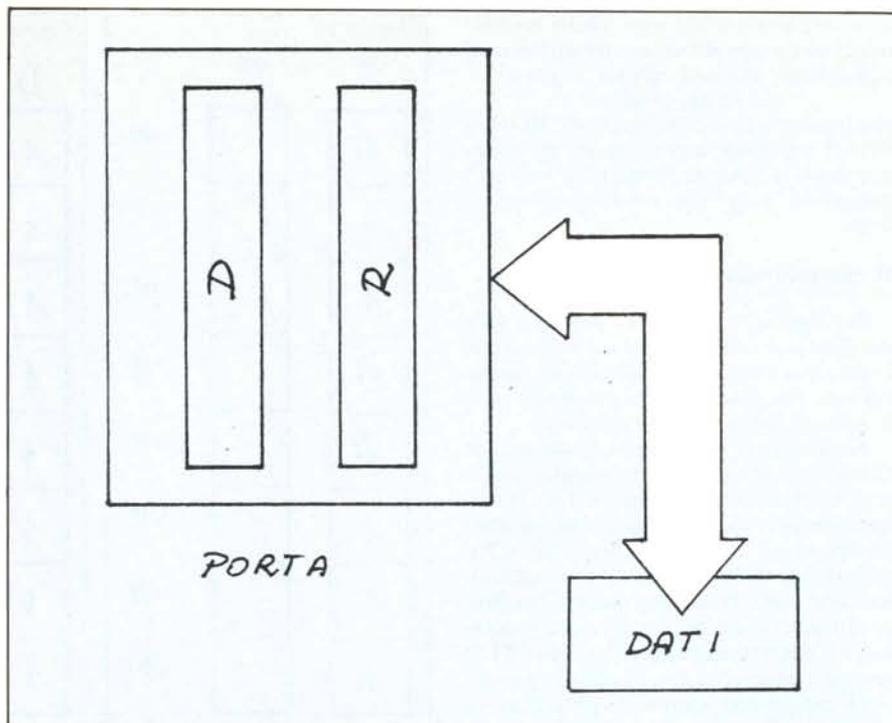


Figura 2 - I due registri attraverso i quali viene gestita una porta d'I/O.

progetto, alcune configurazioni non sono raggiungibili e per la lettura di RDD e RIO bisognerebbe fare alcune considerazioni che in questa sede riteniamo poco opportune. Aggiungiamo che la porta è naturalmente mappata in una locazione ad 8 bit della quale dovremo ignorare lo stato dei bit più significativi.

Dello scopo di questa porta abbiamo già parlato la volta scorsa quindi su di esso non ci soffermeremo molto. Ricordiamo solo che mediante essa, il 64 può gestire più dei 64K di memoria permessi dal suo Bus d'indirizzo poiché, mediante la porta d'I/O, possono essere selezionati (in diversi istanti) blocchi di memoria (Ram o Rom) diversi mappati agli stessi indirizzi (escludendo l'uno o l'altro a seconda delle esigenze). Per sapere qualcosa di più, fate riferimento all'articolo comparso nello scorso numero.

Qualcosa in più vogliamo invece dire sulla funzione svolta da ciascuna linea della porta perché riteniamo che ciò sia abbastanza utile al di fuori degli scopi che ci siamo prefissi all'inizio dell'articolo.

In RDD, ciascun bit a zero indica che il bit corrispondente in RIO è configurato come ingresso mentre un bit a 1 sempre in RDD indica un'uscita in RIO. Lo stato di default in RDD — mappato come detto nella locazione zero — è 00101111 (47 decimale) ma, riferendoci a tale locazione, ai fini pratici, potremo considerarla come un registro a 6 bit e non indicheremo quindi i due zeri di posto più significativo. Diremo allora che, all'accensione, il contenuto della locazione 0 è 101111, cioè cinque linee sono configurate come uscita e una come ingresso.

Le funzioni della porta d'I/O 6510

Bit 0 (1) - Contrassegnato sullo schema anche dalla scritta Loram, controlla la Rom del Basic e gli 8K di Ram che si trovano "sotto" di essa. Tale linea è normalmente alta: quando cambia stato, il suo posto viene occupato da 8K di Ram da A000 a BFFF.

Bit 1 (1) - Contrassegnato dalla scritta Hiram, quando è a livello alto, tiene inseriti gli 8K di Kernal che scompaiono lasciando il posto agli 8K di Ram sottostante (E000-FFFF) quando la linea commuta a livello basso.

Bit 2 (1) - La linea controllata da questo bit è chiamata Charen e viene usata praticamente per la selezione della Rom di 4K del generatore di caratteri (D000-DFFF). Normalmente Charen è in condizione 1 e nell'area indirizzabile del microprocessore sono presenti i dispositivi d'I/O (Vic, Sid, Cia ecc.) e la Rom dei caratteri è inaccessibile per la lettura. Tale condizione si inverte quando Charen cambia stato.

Bit 3 (1) - È la linea di scrittura del registratore a cassette.

Bit 4 (0) - Verifica la pressione di uno dei tasti del registratore.

Bit 5 (1) - Abilita il motore del registratore.

Per fare un rapido esempio sull'uso della porta, soffermiamoci un attimo sul bit 4 che configura come ingresso la corrispondente linea del registro d'I/O: essa è quindi pronta a ricevere un segnale dall'esterno, in questo caso particolare quello relativo alla pressione di uno dei tasti del Datasette. Se annulliamo questo effetto, configurando ad esempio la linea come uscita,

verrà annullato anche quello della pressione dei tasti e non potremo più controllare il registratore. Per una verifica, digitate:

Poke 0,63 <Return>
che trasformerà il contenuto dell' RDD in 11111 e provate a caricare un programma: dopo la richiesta "press play on tape", premendo "play" non otterrete nessun effetto.

Il controllo dell'Interrupt

Sul significato di questa parola e sulle implicazioni dell'interrupt sul sistema abbiamo già ampiamente parlato in questa rubrica. Pertanto diamo per scontate tutte le nozioni legate a tale argomento.

Nel paragrafo precedente, illustrando le funzioni svolte dal bit 2 del registro di Direzione dei dati del microprocessore abbiamo osservato che il generatore dei caratteri occupa lo stesso spazio dei dispositivi d'Ingresso/Uscita del sistema. Deduciamo allora che non è possibile accedere contemporaneamente a tali dispositivi e al generatore. Il sistema accede ai dispositivi d'I/O sessanta volte al secondo quindi, quando noi li escludiamo per porre il generatore di caratteri nell'area indirizzabile del microprocessore al fine di leggerlo, dovremo fare in modo che il sistema non acceda più ad essi. Il perchè è semplice: trovando la Rom dei caratteri al posto dei dispositivi di I/O il sistema reagirà in modo anomalo. Dato che fisicamente l'interrupt è tenuto in vita da un segnale alla frequenza di circa 60 hertz, generato da uno dei due timer interni al Cia 1 del C 64, basterà escludere l'effetto di tale timer per ottenere l'effetto desiderato. Ciò si risolve nell'impedire che il segnale a 60 Hz giunga sull'ingresso Irq del microprocessore, scollegando, per così dire, l'uscita da cui proviene tale segnale dal resto del mondo. Questo è possibile agendo su un certo bit di un apposito registro, posto nella locazione 56334, che controlla

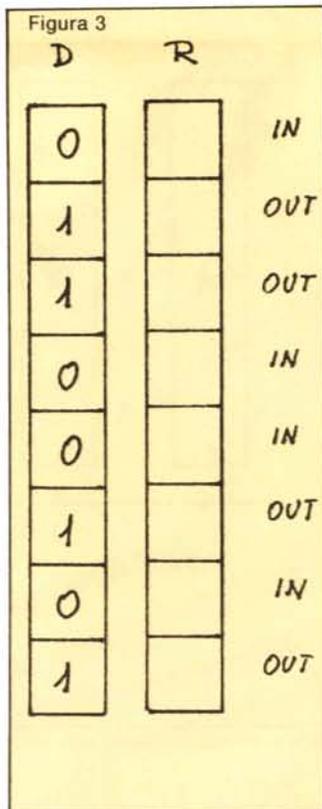


Figura 3 - Il Registro di Direzione dei Dati D permette di configurare ogni singola linea come ingresso o come uscita.

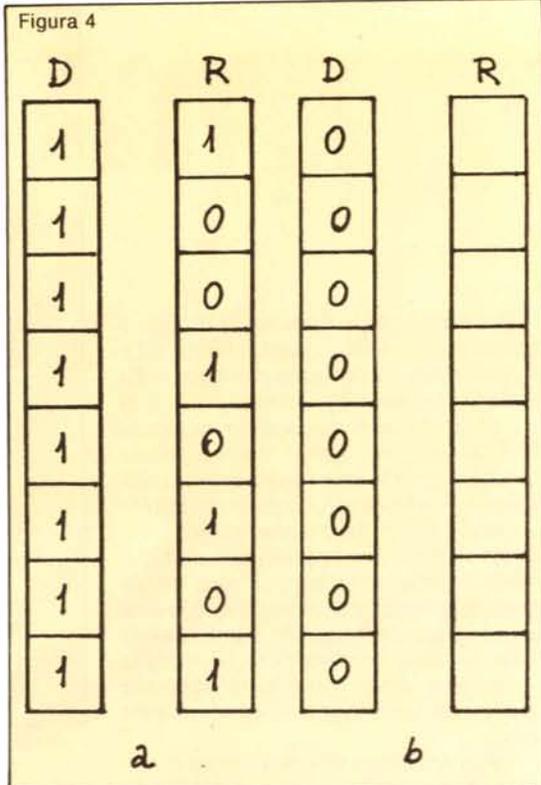


Figura 4 - a) - Se configuriamo tutte le linee come uscite ponendo tutti "1" in D, una parola binaria posta in R la ritroveremo in uscita. b) Se in D poniamo tutti "0" R sarà pronto a ricevere tutte le informazioni che arrivano dall'esterno.

appunto l'effetto del timer di cui abbiamo parlato. Su quest'ultimo argomento spenderemo qualche parola in più la prossima volta. Per il momento, riassumiamo i punti fondamentali che ci permettono la lettura del generatore di caratteri.

La prima cosa da fare è neutralizzare il ciclo d'interrupt. Ciò si ottiene con
POKE 56334,PEEK(56334)AND254
A questo punto potremo mettere il gene-

ratore di caratteri in condizione di essere letto dal microprocessore e, da quanto visto, ciò è possibile con

POKE 1,PEEK(1)AND251.
Effettuata la lettura, potremo ristabilire le condizioni di partenza con
POKE 56334,PEEK(56334)OR1
POKE 1,PEEK(1)OR4
Eventuali dubbi su tali operazioni li chiariremo la prossima volta.

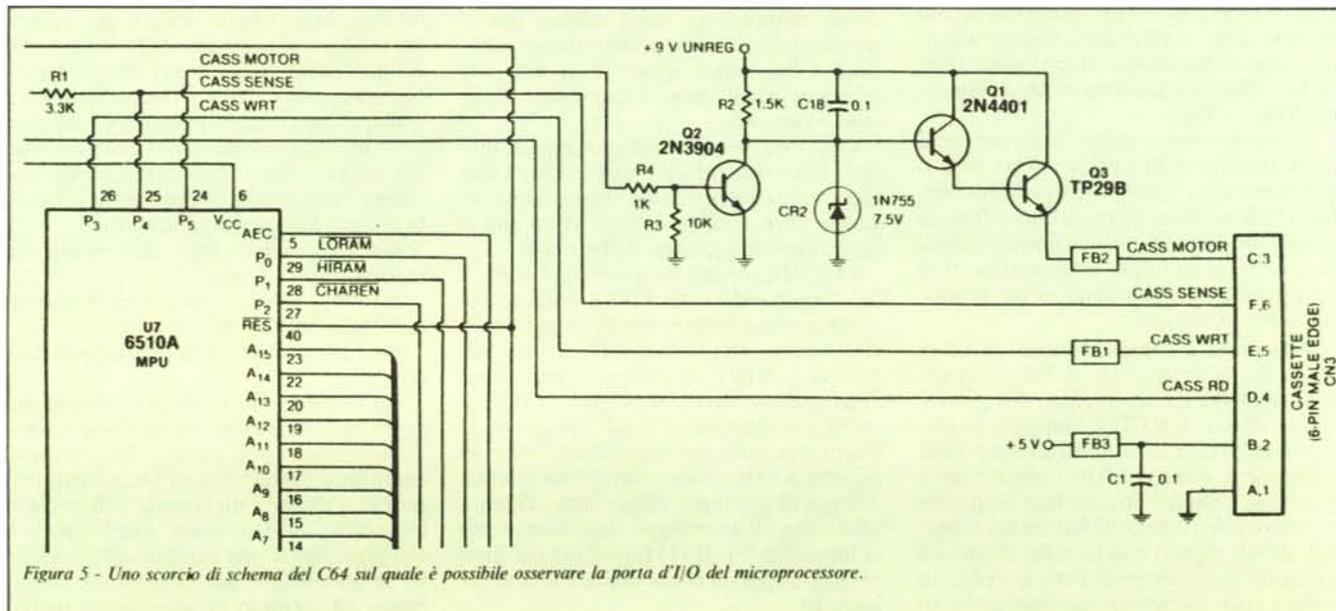


Figura 5 - Uno scorcio di schema del C64 sul quale è possibile osservare la porta d'I/O del microprocessore.

EdiSoftec:
offre programmi
più utili e più facili
per gli utenti italiani.



SUPER WINDOW

- Data base e foglio elettronico fusi in un concetto innovativo.
- Un vero "centro informativo" personale, potente, flessibile, di uso facilissimo.
- Una soluzione immediata ai problemi di gestione dati per: professionisti, impiegati, dirigenti, ecc....
- L'unico in grado di manipolare facilmente archivi di dati provenienti dal centro EDP e da altre applicazioni.

Facile da imparare e da usare.

Il foglio Super Window gestisce 1000 righe di 26 colonne. Ogni colonna può contenere dati di qualsiasi tipo lunghi fino a 60 caratteri.

Con Super Window è possibile mischiare liberamente i tipi di informazioni: voci, numeri e date, anche nella stessa colonna.

Le date possono essere scritte nel modo più comodo. Il Super Window riconosce i vari formati e ne tiene conto in fase di ordinamento.

© EDI-SOFTEC 1985 * Super Window * Memoria 95K + 93K
Archivio: FATTURE.SW Versione 1.1 (Vedi righe sele./ordin.)

Colonna	A	B	C	D	E	F	G	
Nome	FATT	DATA	EMI	CLIENTE	SCADENZA	IMP. NETTO	IVA	
Formato	4	8	10	10	8	9	3	
Riga 950	186	24	1985	ALBERTI	30/1/85	1200000	18	1770000
Riga 952	185	85	3/24	ALBERTI	85/3/24	1200000	18	1534000
Riga 948	181	23	2-85	BERTONI	20/3-85	1200000	18	1475000
Riga 948	182	24	02-85	BERTONI	20/4/85	1500000	9	1630000
Riga 950	56	15	03/85	BERTONI	15/5/85	5733000	9	624897
Riga 947	187	35	02/85	CITILIANI	15/03/85	250000	18	295000
Riga 948	187	10	7/85	MARTINI	15/5/85	1725000	9	1880795
Riga 948	183	03	1985	NARDO*	15/3/85	2750000	18	3245000
Riga 951	57	3	19	NARDO*	30/03/85	5675000	9	618575
Riga 949	184	10	3/85	NARDO*	1985/5/1	5300000	9	5727000

950 A (Numero) 186

F1 Quadro/Visione F2 Impara/Ripeti F3 Leggi F4 Scrivi
F5 Selezione F6 Riordina F7 Nomi/Formati F8 Appella
F9 Stampa F10 Copie/Calcola End Fine lavoro Sec Ripetizione

Il Super Window può ordinare in tutti i modi possibili (in questo caso, ad esempio, per nome cliente e per data scadenza).

Il Super Window può effettuare calcoli aritmetici fra colonne o fra una colonna ed un numero.

Le funzioni del Super Window disponibili in ogni momento sono sempre mostrate nelle 3 righe più basse dello schermo. Le funzioni sono richiamabili premendo il tasto funzione corrispondente.

Nessun comando da ricordare.

Le funzioni possibili sono sempre chiaramente spiegate nelle ultime 3 righe dello schermo. Sono tutte richiamabili premendo i tasti funzione. Alcune delle funzioni possibili sono qui illustrate.

SELEZIONA Consente di selezionare tutte le righe che soddisfano una o più condizioni. Ogni selezione può contemporaneamente verificare fino a 10 diverse condizioni alternative. Non ci sono limiti al numero di selezioni successive eseguibili; ogni selezione successiva opera nell'insieme di righe risultanti dalla selezione precedente.

RIORDINA Consente di riordinare le righe secondo la gerarchia numerica, alfabetica o di data di una determinata colonna. L'ordinamento può essere fatto in modo ascendente o discendente ed è possibile riordinare su più colonne contemporaneamente fino ad un massimo di 10.

CALCOLA Si possono effettuare calcoli aritmetici fra colonne e fra colonne e numeri. È anche possibile ottenere i valori della sommatoria progressiva dei dati di una colonna.

LEGGI/SCRIVI I dati introdotti possono essere registrati in due formati diversi. Il primo formato (formato Super Window) è compatibile con il formato SYLK del Multiplan e permette di leggere o scrivere archivi Multiplan o della famiglia ASSISTANT IBM. Il secondo formato (formato diretto) consente di leggere o scrivere gli archivi delle applicazioni gestionali più diffuse (anagrafiche clienti/fornitori, articoli a magazzino, ecc.) o provenienti dal centro EDP.

IMPARA/RIPETI È possibile memorizzare le sequenze operative più usate per ogni specifica applicazione del Super Window, per poi rieseguirle quante volte si vuole.

STAMPA È possibile stampare tutti o parte dei dati disponendo le colonne che si vuole stampare nell'ordine preferito. È possibile richiedere totali generali e parziali in una o più colonne. È possibile effettuare "stampe" su video oppure stampare in "spool" cioè in sovrapposizione di tempo con le elaborazioni successive.

POTENZA E FLESSIBILITÀ NON COMUNI.

Il Super Window è anche una vera "centralina di scambio" di informazioni fra archivi e ambienti normalmente non compatibili. Con Super Window puoi leggere ad esempio gli archivi clienti della contabilità generale; selezionare i clienti che ti interessano ad esempio quelli con fatturato superiore a ...; riordinarli per zona e per agente; estrarre i dati di vendita ed inviarti al Multiplan o ad un programma di presentazione grafica.



In vendita presso i concessionari di Personal e Professional Computers MS-DOS, ad esempio IBM e Olivetti.

EdiSoftec S.p.a. - Corso S. Maurizio, 79
10124 Torino - Tel. (011) 839.64.45-6-7

Multiplan e MS-DOS sono marchi registrati dalla Microsoft. ASSISTANT è un marchio registrato dalla IBM.

Desidero ricevere ulteriori informazioni sui prodotti EDI-SOFTEC e l'indirizzo del rivenditore più vicino.

Nome Azienda
Attività
Indirizzo
CAP Città
Personal Computer usato
Programmi più utilizzati

QUOTAZIONI

Materiale nuovo imballato

CENTRO
ASSISTENZA
SPECTRUM

SUMUS

SUMUS s.r.l.
Via S. Gallo 16/r
50129 Firenze
tel. 055/29.53.61

IPEROFFERTE MAGIA SUMUS (QUANTITÀ LIMITATA)

Spectrum 48K con 6 games pack	279.000
Spectrum 48K plus con 6 games pack	339.000
Apple compatibile con tastiera separata di tipo professionale, 64K, doppio processore (6502 + Z-80)	799.000
PC IBM compatibile, 128K, doppio drive da 360K cad., clock calendario con batteria in tampone, interfaccia parallela e seriale	3.150.000

COMPATIBILE APPLE

LEMON II modelli vari	telefonare
MOUSE IC 64K biprocessore	679.000
MOUSE IIC biprocessore con tastiera separata ecc. ..	799.000

ACCESSORI PER APPLE O COMPATIBILI

Floppy disk controller	79.000
Floppy disk drive (slim o standard)	349.000
Interfaccia stampante EPSON (grafica)	94.000
Interfaccia stampante EPSON con buffer 16K (espandibile on board a 64K con aggiunta integr.)	199.000
Interfaccia stampante CENTRONICS (non grafica)	73.000
Interfaccia stampante GRAPPLER (grafica)	94.000
Scheda CP/M (con Z-80), senza software	69.000
Scheda 80 colonne con soft switch	139.000
Scheda interfaccia seriale RS-232 (no buffer)	79.000
Scheda interfaccia Super Seriale (buffer)	180.000
Scheda espansione memoria + 128K	349.000
Scheda convertitore A/D 16 ingressi	125.000
Scheda musicale	109.000
Scheda sintesi vocale	69.000
Scheda orologio calendario con accumulatori	99.000
Scheda interfaccia monitor RGB	99.000
Scheda PAL (non raccomandata per il colore)	99.000
Scheda programmatore EPROM (2716/32/64)	99.000
Joystick plastico di precisione	42.000
Joystick metallico	37.000
Mouse con software	125.000
Modem con accoppiatore acustico ed interfaccia	259.000
Penna ottica con software	335.000
Language card (espande i vecchi 48K a 64K)	89.000

MONITORS

Monocromatici, vari tipi, primarie marche, da lire	152.000
A colori, vari tipi, primarie marche, da lire	455.000

STAMPANTI

Mannesmann Tally MT-80 (80 cps, 80/132 col., grafica, Epson compatibile, foglio singolo e modulo continuo)	telefonare
Epson RX 80 F/T (stesse caratteristiche ma 100 cps)	737.000
Stampante Welco (stesse caratteristiche ma 130 cps)	699.000
Idem con interfaccia seriale anziché parall.	730.000
Mitsui 2100, 120 cps, 80/132 colonne, near letter quality	999.000
Margherita, 18 cps	699.000
Idem con tastiera, usabile come macchina per scrivere intelligente o come stampante, completa di display multilinea a cristalli liquidi, correzione automatica	899.000

PLOTTERS

Plotter intelligente Mannesmann Tally Pixy 3, 3 penne formato A4	999.000
Plotterino/stampante Sony, 4 colori, veloce, possibilità di rotolo, larghezza 21 cm (A4), 80 colonne se usato in modo stampante	534.000

ACCESSORI PER PC/IBM E COMPATIBILI

Cavo stampante PC/stampante parallela	50.000
Unità a disco 5" 1/4 aggiuntiva	399.000
Espansione di memoria +64K da montare sulla scheda già esistente	115.000

COMPUTER SANYO

MBC-550 - 16 bit - 128K RAM espandibili a 256 con incrementi da 64K - parzialmente IBM compatibile - grafica alta risoluzione 640 x 200 punti in 8 colori indipendenti - tastiera professionale - interfaccia stampante (senza cavo) - una unità a disco da 160K - compreso MS-DOS, Wordstar, Calcstar, BASIC	2.099.000
MBC-555 - come il precedente ma con due drive ed in più Datastar, Formsort, Reportstar, Spellstar, Mailm. ..	2.699.000
MBC-550/2 - come 550 ma con disco da 360K - comprende programmi "usa Sanyo PC", "programma in BASIC", "disegno con il CAD", "contabilità", e manuali in italiano	2.450.000
Disk drive aggiuntivo (trasforma 550 in 555)	399.000
Disk drive aggiuntivo (trasforma 550/2 in 555/2)	450.000
Cavo stampante MBC/stampante parallela	59.000
Espansione di memoria, installata, 64K RAM	99.000
Interfaccia RS-232 per serie MBC	118.000

COMPUTERS PORTATILI

BONDWELL 12 - a valigia - 64K RAM - video incorporato da 9" - due unità a disco da 256K (non formattati) cadauno - secondo drive compatibile Spectravideo, Kaypro od Osborne con comando software - interfaccia parallela per stampante - due interfacce seriali RS-232 - SINTETIZZATORE VOCALE INCORPORATO - uscita monitor supplementare - comprende CP/M, Wordstar, Calcstar, Datastar, Reportstar, Mailmerge ..	2.721.000
---	-----------

TAVOLI PER COMPUTERS

Ciatti mod. Memory (cm 60 x 82 x 115, piano scorrevole, disponibile bianco, nero, noce)	179.000
Ciatti mod. Logic, (ripiegabile, con ruote e supporto monitor, colori bianco e nero)	289.000
Eledra, tipo piccolo (circa 70 x 80 x 50), colore bianco, progettato per C64, adattissimo a Apple & C. e per stampanti ..	58.500
Socored, super professionale	346.000
Supporto in plexiglass per stampanti	61.016

PORTADISCHETTI E VARIE

In plexiglass, da 10 dischi	4.237
Da 40 dischi a vaschetta con serratura	24.576
Da 80 dischi a vaschetta con serratura	33.050
Pinza bucatrice per floppy. Consente di usare entrambi i lati del dischetto (Apple, Commodore)	5.932

PREZZI INCREDIBILI SU:

APPLE - MACINTOSH - OLIVETTI M 24

SUMUS - LA PIÙ GRANDE ORGANIZZAZIONE DI VENDITA IN TOSCANA DI HOME & PERSONAL COMPUTERS - NON POSSIAMO ELENCARE TUTTO - VENITE A TROVARCI DI PERSONA - SIAMO APERTI ANCHE IL SABATO (fino a estate).

I prezzi qui indicati sono da intendersi franco negozio IVA esclusa. I prezzi e le disponibilità variano - telefonateci prima dell'ordine.

ATTREZZATISSIMO
CENTRO ASSISTENZA
SPECTRUM.

TUTTI I RICAMBI
A MAGAZZINO.

SCONTO 50%
AI NOSTRI CLIENTI!



IL
NEGOZIO
DI
SUPER
SUMUS!



Il linguaggio macchina sullo Spectrum

Terza parte

Nei precedenti numeri di TuttoSpectrum abbiamo visto diverse tecniche mediante le quali è possibile passare parametri tra un programma Basic chiamante ed un programma in linguaggio macchina e viceversa.

Abbiamo visto come questo compito possa venire assolto in maniera banale mediante l'uso delle istruzioni PEEK e POKE, e come però esistano anche tecniche più "eleganti", in cui al programmatore viene risparmiato l'onere di occuparsi del passaggio dei parametri. Questo avviene implicitamente ogni volta che il controllo viene trasferito dal programma Basic a quello in linguaggio macchina o viceversa.

In particolare al termine della puntata precedente avevamo visto un metodo per passare un qualsiasi numero di valori numerici interi da un programma Basic ad uno in linguaggio macchina facendo uso della istruzione DEF FN. Ora esamineremo una tecnica che ci permetta di svolgere la stessa operazione ed in più quella inversa ovvero il trasferimento di dati da un programma in linguaggio macchina ad uno in Basic. Vedremo come sia possibile in linguaggio macchina leggere o scrivere dei valori direttamente all'interno delle variabili Basic.

Oltre che per trasferire interi di uno o due byte, questi due metodi si adattano altrettanto bene anche al trasferimento di numeri in virgola mobile, nel formato a cinque byte utilizzato dallo Spectrum. Su

tale argomento però preferiamo non soffermarci almeno per il momento.

Non ci fermeremo comunque al passaggio di semplici valori numerici o di coppie di byte, ma esamineremo anche il caso di stringhe. Tuttavia, prima di addentrarci in questi argomenti e per poter meglio comprendere quanto segue, conviene approfondire un attimo la nostra conoscenza su come vengano gestite le variabili nello Spectrum.

Il capitolo 24 del manuale dello Spectrum, è dedicato alla gestione della memoria. Osserviamo la mappa della memoria riportata a pagina 121 (edizione inglese). Si nota come l'area destinata alle variabili del Basic, indicata con la dicitura "variables", segue immediatamente l'area in cui è memorizzato il programma Basic, "Basic program" nella figura. L'indirizzo iniziale dell'area variabili è contenuto nella variabile di sistema VARS, che si trova alle locazioni di memoria 23627 (il byte meno significativo) e 23628 (il byte più significativo). Il valore contenuto in VARS non è costante, ma dipende da (e nel caso di Spectrum senza microdrive solo da) la dimensione del programma Basic che precede.

Ogni volta che viene dato il RUN l'area delle variabili di sistema viene reinizializzata e tutte le variabili presenti in memoria vengono cancellate. Provate subito dopo aver acceso lo Spectrum a dare questa se-

quenza di comandi (senza numeri di linea)

```
PRINT a
```

il computer risponderà con il messaggio

```
2 VARIABLE NOT FOUND, 0:1
```

in quanto alla variabile "a" non è ancora stato assegnato alcun valore. Infatti inserendo

```
LET a=0
```

e poi

```
PRINT a
```

viene regolarmente stampato 0 in cima allo schermo. Se a questo punto diamo il comando

```
RUN
```

apparentemente non otteniamo alcun risultato in quanto in memoria non è presente alcun programma, tuttavia se inseriamo di nuovo il comando

```
PRINT a
```

otteniamo daccapo il messaggio

```
2 VARIABLE NOT FOUND, 0:1
```

segno che il comando RUN oltre a lanciare un eventuale programma e cancellare lo schermo cancella anche tutte le variabili presenti in memoria.

Limitandoci per il momento a considerare variabili numeriche semplici (non matrici), esaminiamo cosa succede ogni volta che si fa riferimento ad una variabile. Per prima cosa il computer verifica se la variabile sia o meno già presente in memoria. L'area variabili viene scandita sequenzialmente ed il nome di ciascuna variabile viene confrontato con quello presente nell'istruzione o nel comando Basic in esecuzione. Se la ricerca termina con successo allora il valore della variabile viene letto o aggiornato a seconda dell'operazione da eseguire. In particolare se si tratta di un'operazione di aggiornamento di una variabile numerica, il valore viene riscritto sopra il vecchio e la posizione della variabile nella memoria non cambia.

Se la ricerca del nome della variabile in memoria non ha successo, nel caso si trattasse di un'operazione di lettura il sistema risponderrebbe con il messaggio

```
2 VARIABLE NOT FOUND.
```

Nel caso si trattasse di un'operazione di assegnazione di un valore ad una variabile menzionata per la prima volta, il sistema provvede innanzitutto a riservare spazio sufficiente per contenere il nome ed il valore della nuova variabile e poi trascrive entrambi. La nuova variabile viene accodata in fondo all'area variabili, facendo slittare in avanti tutto quanto segue.

Tutta questa dissertazione non è fine a sé stessa, ma ci suggerisce un metodo per andare a leggere il contenuto di una o più variabili Basic direttamente in linguaggio macchina. Abbiamo detto che il RUN pro-

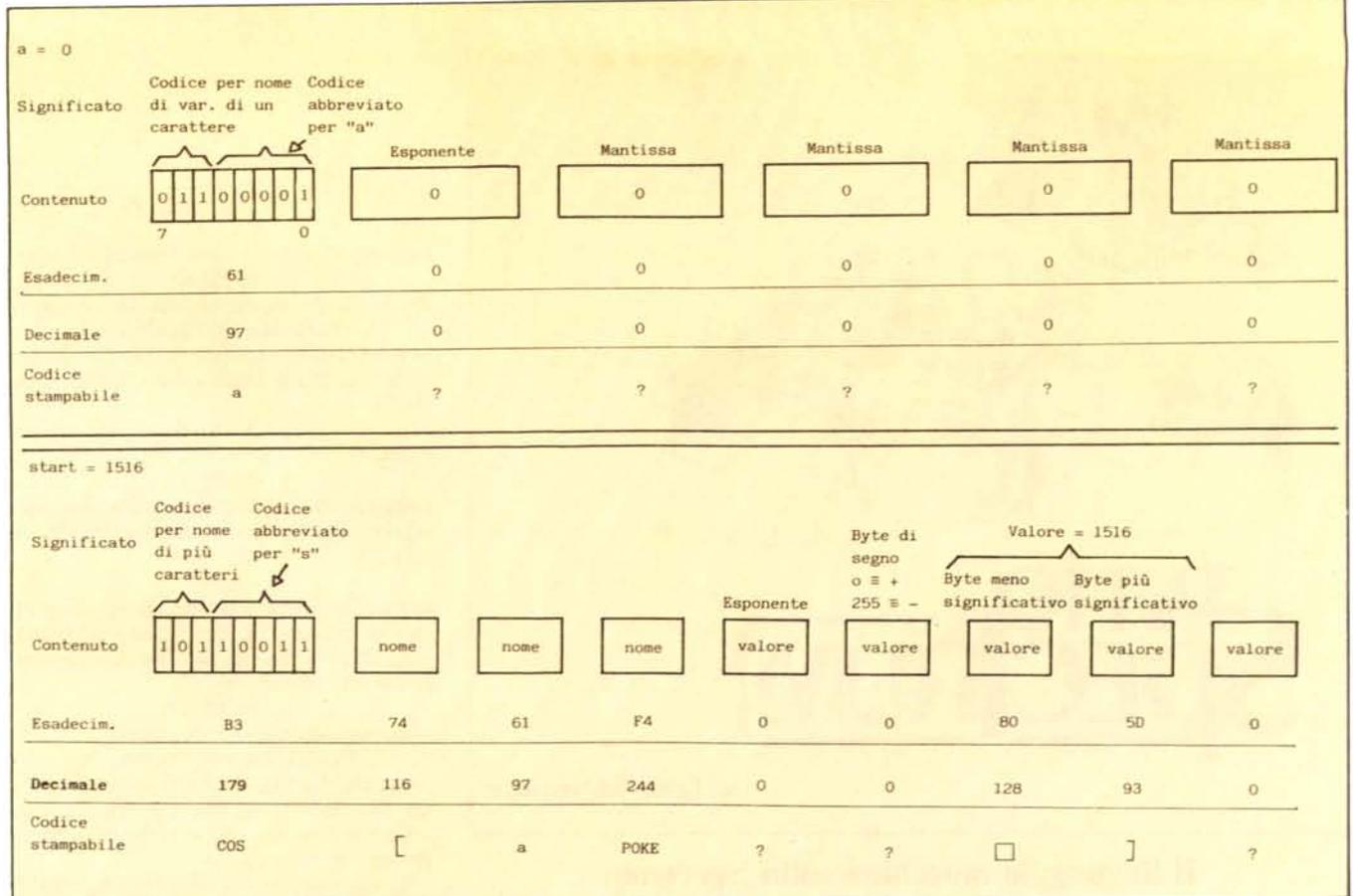


Figura 1 - Rappresentazione in memoria del nome e del valore di variabili numeriche.

duce come primo effetto quello di cancellare tutte le variabili. Durante l'esecuzione del programma via via che l'interprete incontra nuove assegnazioni di variabili provvede ad accodare ogni nuova variabile alle altre nell'ordine. La prima variabile contenuta nell'area è puntata dalla variabile di sistema VARS, di cui abbiamo parlato sopra.

Se la prima istruzione del programma è:

```
1 LET a = 0
```

siamo sicuri di avere un puntatore (VARS) alla zona di memoria in cui è conservata la variabile "a".

A questo punto per capire come poter ricavare un valore di qualche interesse dobbiamo studiare il modo in cui vengono memorizzate le variabili nello Spectrum. Per adesso continueremo ad occuparci solo di variabili numeriche semplici.

Alle pagine 122 (in fondo) e 123 (in testa) del manuale sono illustrati in maniera piuttosto schematica i formati di memorizzazione delle variabili numeriche semplici, sia nel caso di nome costituito da un unico carattere che da più caratteri. Notate come lo schema generale sia:

NOME VARIABILE VALORE VARIABILE

Il valore della variabile viene memorizzato nel consueto formato a cinque byte utilizzato nello Spectrum per rappresentare i valori numerici. Il primo byte contiene l'esponente, i restanti quattro il numero vero

e proprio. Ricorderete anche che se il numero è contenuto dell'intervallo -65535, 65535, viene adottata una forma particolare in cui il terzo e quarto dei cinque byte contengono il modulo (il valore senza segno) del numero stesso (il terzo la parte meno significativa e il quarto quella più significativa).

Per quanto riguarda la memorizzazione del nome della variabile abbiamo già notato come esista qualche differenza di notazione tra i nomi costituiti da una sola lette-

ra e quelli costituiti da più caratteri. Osserviamo che il primo byte della codifica del nome di una variabile deve contenere il codice di un carattere alfabetico. Poiché i caratteri alfabetici sono solo 26 (dato che non si fa distinzione tra lettere maiuscole e minuscole), i 256 valori diversi rappresentabili con un byte sono decisamente sovrabbondanti. I progettisti della Sinclair hanno pensato allora di sfruttare questa sovrabbondanza per inserire nel primo byte un'informazione aggiuntiva sul tipo di variabile rappresentata. Poiché 2 alla quinta potenza è uguale a 32, cinque bit sono sufficienti a rappresentare il primo carattere alfabetico del nome.

I restanti tre bit (nella codifica i tre bit più significativi) servono ad indicare di che tipo di variabile si tratta. La codifica è la seguente

- 000 NON USATO
- 001 NON USATO
- 010 VARIABILE DI STRINGA
- 011 VARIABILE NUMERICA
- 100 VARIABILE NUMERICA MULTIPLA (MATRICE)
- 101 VARIABILE NUMERICA
- 110 VARIABILE DI STRINGA MULTIPLA (MATRICE)
- 111 VARIABILE DI CONTROLLO DI UN CICLO FOR ... NEXT

```

10 PRINT "INSERIRE LA VARIABILE NUMERICA"
20 PRINT "NELLA FORMA"
30 PRINT "50 LET <nome> = <valore>"
40 PRINT "POI DIGITARE RUN 50 " : STOP
50 LET q=65535
100 LET VARS=PEEK 23627+PEEK 23628*256
110 PRINT "VARS=" ;VARS
120 REM stampa della codifica del nome
130 LET cont=VARS
140 PRINT "codifica del nome": PRINT
"codice","car. stampabile"
150 PRINT PEEK cont,CHR$(PEEK cont)
160 IF (PEEK cont)>=128 THEN GO SUB 300
170 REM stampa dei bites del valore
180 PRINT "byte del valore"
190 FOR i=1 TO 5
200 LET cont=cont+1
210 PRINT i,PEEK cont
220 NEXT i
230 STOP
300 REM stampa il resto di un nome
310 LET cont=cont+1
320 PRINT PEEK cont,CHR$(PEEK cont)
330 IF PEEK cont<128 THEN GO TO 310
340 RETURN

```

Figura 2 - Stampa della codifica del nome e del contenuto della prima variabile numerica inserita in un programma Basic.

```

1 REM fa la somma di due numeri leggendo i valori direttamente dalle variabili
10 LET A=0
20 LET B=0
30 LET SOMMA=0
40 FOR I=29000 TO 29823
50 READ A
60 POKE I,A
70 NEXT I
80 DATA 221,42,75,92,221,110,3,221,102,4,221,94,9,221,86,10,25,221,117,19,221,
116,20,201
90 INPUT "PRIMO ADDENDO: ";A
95 INPUT "SECONDO ADDENDO: ";B
100 RANDOMIZE USR 29000
110 PRINT "SOMMA=" ;SOMMA
130 GO TO 90
    
```

Figura 3a

#HISOFT GEN2 ASSEMBLER#
Copyright HISOFT 1983
All rights reserved

```

Pass 1 errors: 00

7148 DD2A4B5C 10 ORG 29000
7148 DD6E83 20 LD IX,(#5C4B) ; CARICA VARS
714C DD6684 30 LD L,(IX+3) ; PRELEVA IL PRIMO ADDENDO
714F DD6684 40 LD H,(IX+4)
7152 DD5E89 50 LD E,(IX+9) ; PRELEVA IL SECONDO ADDENDO
7155 DD568A 60 LD D,(IX+10)
7158 19 70 ADD HL,DE ; FA LA SOMMA
7159 DD7513 80 LD (IX+19),L ; COPIA IL RISULTATO IN SOMMA
715C ED7414 90 LD (IX+20),H
715F C9 100 RET

Pass 2 errors: 00

Table used: 13 from 132
    
```

Figura 3b

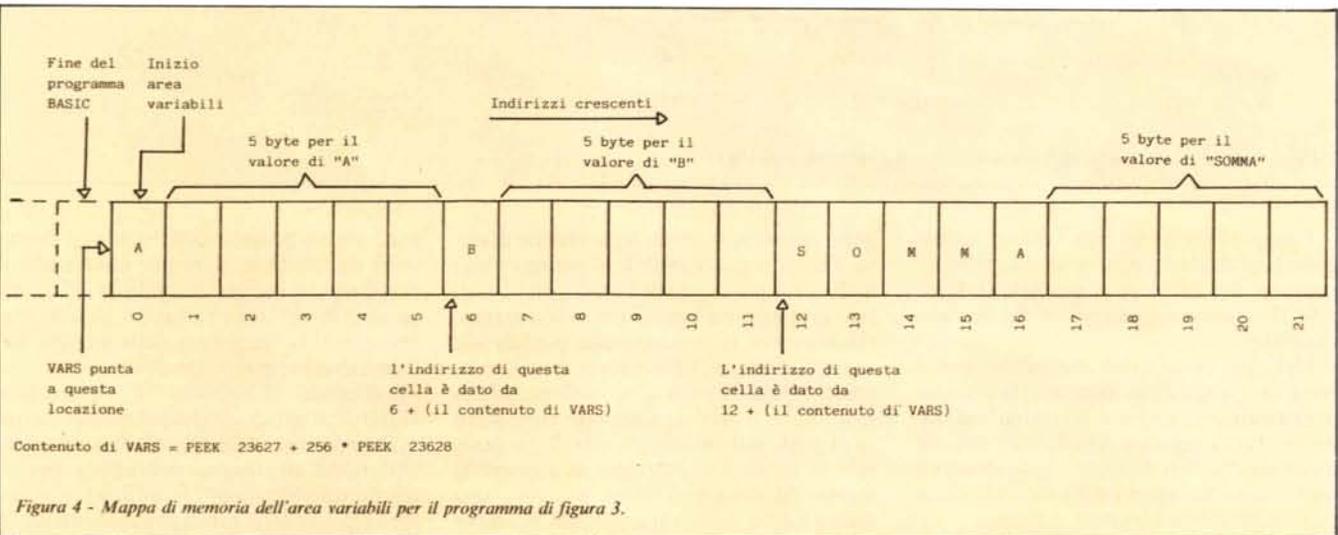


Figura 4 - Mappa di memoria dell'area variabili per il programma di figura 3.

```

1 REM INVERSIONE DI UNA STRINGA MEDIANTE UNA VARIABILE DI STRINGA SEMPLICE
2 REM OCCORRE DARE IL RUN OGNI VOLTA
10 INPUT "STRINGA: ";A$: PRINT A$
30 FOR I=29000 TO 29826
40 READ A
50 POKE I,A
60 NEXT I
70 DATA 42,75,92,38,6,22,0,25,78,35,89,22,0,229,25,209,19,203,57,26,70,119,120,18,43,19,
13,194,87,113,201
90 RANDOMIZE USR 29000
100 LET start=PEEK 23627+PEEK 23628*256
120 LET end=PEEK (start+1)+256*PEEK (start+2)
140 FOR I=start+3 TO start+end-2: PRINT CHR$(PEEK I): NEXT I: PRINT
    
```

Figura 5a

```

#HISOFT GEN2 ASSEMBLER#
Copyright HISOFT 1983
All rights reserved

Pass 1 errors: 00

7148 10 ORG 29000
7148 2A4B5C 20 LD HL,(#5C4B) ; CARICA IL CONTENUTO DI VARS
714B 23 25 INC HL ; PUNTA ALLA LUNGHEZZA DELLA LISTA
714C 4E 30 LD C,(HL) ; LA COPIA IN C
714D 23 35 INC HL
714E 59 40 LD E,C ; ED ANCHE IN DE
714F 1600 50 LD D,#00
7151 E5 55 PUSH HL
7152 19 60 ADD HL,DE ; ORA HL PUNTA ALLA FINE DELLA LISTA
7153 D1 65 POP DE
7154 13 67 INC DE ; E DE ALL'INIZIO
7155 CB39 70 SRL C ; DIVIDE LA LUNGHEZZA PER 2
7157 A 80 LOOP LD A,(DE) ; ESEGUE IL CICLO DI SCAMBIO DEI CARATTE
RI
7158 46 90 LD B,(HL)
7159 77 100 LD (HL),A
715A 78 110 LD A,B
715B 12 120 LD (DE),A
715C 26 130 DEC HL ; SPOSTA I DUE PUNTORI VERSO
715D 13 140 INC DE ; IL CENTRO DELLA LISTA DI UN CARATTERE
715E 80 150 DEC C
715F C25771 160 JP NZ,LOOP
7162 C9 170 RET

Pass 2 errors: 00

Table used: 24 from 168
    
```

Figura 5b

```

1 REM INVERSIONE DI UNA STRINGA CON IL METODO DEL VETTORE DI STRINGA
10 DIM A$(1,20)
20 FOR I=29000 TO 29830
30 READ A
40 POKE I,A
50 NEXT I
60 DATA 42,75,92,38,6,22,0,25,78,35,89,22,0,229,25,209,19,203,57,26,70,119,120,
18,43,19,13,194,91,113,201
90 INPUT "INSERIRE UNA STRINGA, MAX 20 CAR":A$(1)
100 GO SUB 200
110 RANDOMIZE USR 29000
115 PRINT "LA STRINGA VIENE STAMPATA CON TUTTI GLI SPAZI BIANCHI CHE LA PRE
CEDONO IN MEMORIA:"
120 GO SUB 200
130 GO TO 90
200 FOR I=1 TO 20
210 PRINT A$(I,1)
220 NEXT I
230 PRINT
240 RETURN
    
```

Figura 6a

```

#HISOFT GEN2 ASSEMBLER#
Copyright HISOFT 1983
All rights reserved

Pass 1 errors: 00

7148 10 ORG 29000
7148 2A4B5C 20 LD HL,(#5C4B) ; LOCAZIONE PUNTATA DA VARS
714B 1E06 30 LD E,#06 ; HL PUNTA ALLA
714D 1600 35 LD D,#00 ; LUNGHEZZA
714F 19 37 ADD HL,DE ; DELLA STRINGA
7150 AE 40 LD C,(HL) ; CHE VIENE COPIATA IN C
7151 2E 50 INC HL
7152 59 60 LD E,C
7153 1600 70 LD D,#00
7155 E5 80 PUSH HL
7157 D1 90 ADD HL,DE ; HL PUNTA ALLA FINE DEL TESTO
7158 13 100 POP DE ; DE PUNTA ALL' INIZIO
7159 CB39 110 INL DE ; DEL TESTO
715B 1A 120 SRL C ; DIVIDE LA LUNGHEZZA PER 2
715D 1A 150 LOOP LD A,(DE) ; CICLO DI SCAMBIO DEI CARATTERI
715E 46 160 LD B,(HL)
715F 77 170 LD (HL),A
7158 78 180 LD A,B
715F 12 190 LD (DE),A
7160 2B 200 DEC HL
7161 15 210 INC DE
7162 0D 220 DEC C
7163 C25B71 230 JP NZ,LOOP
7166 C9 240 RET

Pass 2 errors: 00

Table used: 24 from 168
    
```

Figura 6b

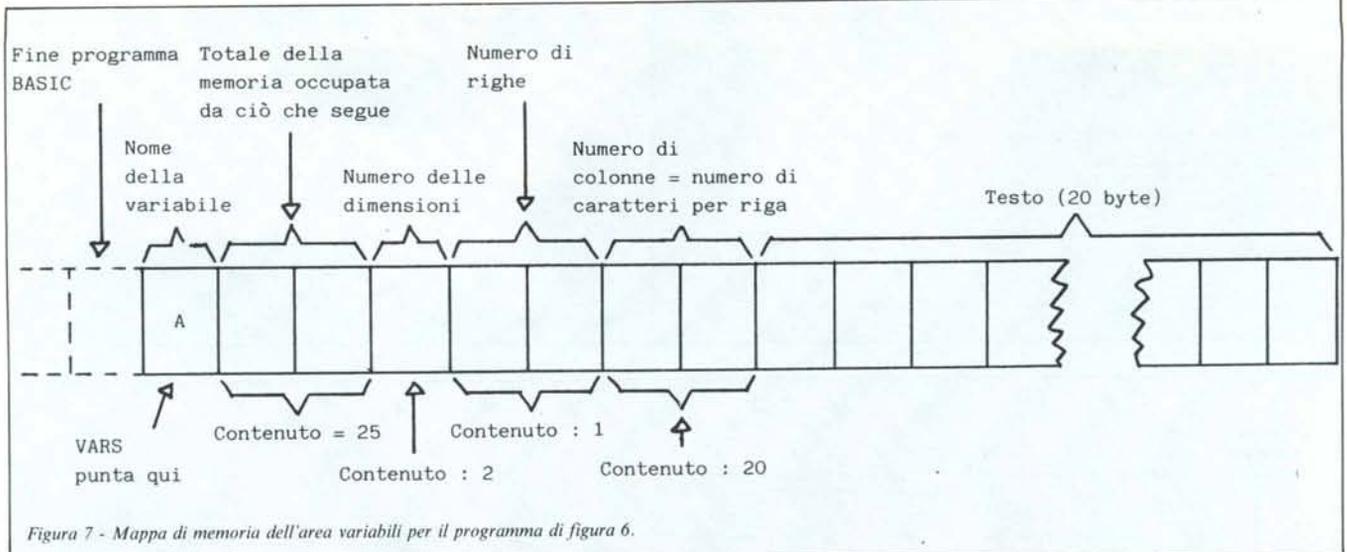


Figura 7 - Mappa di memoria dell'area variabili per il programma di figura 6.

I cinque byte meno significativi contengono la codifica compatta del carattere alfabetico. Per ottenere la normale codifica ASCII occorre sommarli 96 (in esadecimale 60).

Nel caso di variabili numeriche con il nome più lungo di un carattere, la codifica per i caratteri che vanno dal secondo all'ultimo è quella regolare ASCII per i sette bit meno significativi di ciascun byte, mentre il bit più significativo è 1 o 0 a seconda che si tratti dell'ultimo carattere o meno.

Vediamo alcuni esempi. In figura 1 sono rappresentate le codifiche delle variabili numeriche "a" con valore 0 e "start" con valore 1516.

Giunti a questo punto dovreste essere in grado di predichiarare delle variabili e quindi di scandire l'area variabili riconoscendo la loro codifica. Qualcosa di simile viene fatto nel programma di figura 2.

Tornando ora al nostro argomento vediamo un esempio di come tutto ciò possa essere sfruttato per passare dati ad un programma in linguaggio macchina e viceversa. Il programma mostrato in figura 3, esegue la somma di due numeri compresi tra 0 e 32767. I valori degli addendi vengono prelevati direttamente dentro alle variabili "a" e "b", il risultato viene posto direttamente dentro la variabile "somma". Notate come le prime tre istruzioni del programma siano

```
LET a=0
LET b=0
LET somma=0
```

Questo ci assicura che le tre variabili siano memorizzate consecutivamente a partire dalla locazione puntata da VARS. Una mappa del segmento di memoria di interesse è mostrato in figura 4.

Consideriamo ora le variabili di stringa. Possiamo pensare di implementare un sistema per il passaggio dei parametri analogo a quello che abbiamo appena descritto per le variabili numeriche. Tuttavia, poiché la lunghezza delle variabili di stringa non è costante, queste nello Spectrum ven-

gono gestite in maniera leggermente diversa. Quando una variabile di stringa viene dichiarata per la prima volta, viene calcolato lo spazio necessario per la sua memorizzazione ed essa viene posta in coda alle variabili presenti in memoria in quel momento. Quando però se ne vuole modificare il valore, il nuovo valore non viene scritto al posto del vecchio, in quanto in generale la lunghezza potrebbe non essere la stessa. Al contrario viene generata una nuova copia della variabile con il nuovo valore e questa viene accodata in fondo all'area. Immediatamente dopo la vecchia istanza della variabile viene cancellata e lo spazio da essa occupato viene recuperato facendo slittare in giù nella memoria tutto ciò che segue.

È chiaro quindi che anche dichiarando per prima una variabile di stringa, in modo che essa venga puntata dalla coppia VARS, ne perderemmo ogni traccia nel momento in cui gli assegniamo un secondo valore. Applicare il nostro metodo ad un tale tipo di variabile può risultare utile solo se il parametro da passare al sottoprogramma in linguaggio macchina è costante e non varia per ogni esecuzione del programma Basic.

La figura 5 contiene un esempio di applicazione. Il programma in linguaggio macchina esegue l'inversione di una stringa, ovvero data una stringa in ingresso ne scambia il primo carattere con l'ultimo, il secondo con il penultimo, e così via. Il programma riceve la stringa da invertire nella variabile "a\$" nella quale pone anche il risultato finale. La variabile "a\$" viene dichiarata per prima nel programma Basic, affinché venga direttamente puntata dalla coppia VARS. Per i motivi che abbiamo appena esposti, per far funzionare correttamente il programma con un'altra stringa è necessario dare il RUN una seconda volta.

A pagina 124 del manuale viene descritto il formato di memorizzazione delle variabili di stringa. Come potete vedere il

testo vero e proprio della stringa è preceduto da tre byte, il primo contenente il carattere che insieme al simbolo "\$" costituisce il nome della variabile, gli altri due contenenti la lunghezza della stringa, nel formato byte meno significativo-byte più significativo. Il simbolo "\$" tipico delle variabili di stringa non viene memorizzato.

Il problema della riassegnazione della variabile di stringa può essere però superato con un sotterfugio. È sufficiente utilizzare il posto di una stringa avente un unico elemento. In questo caso però siamo vincolati a predichiarare, e quindi a rispettare, la lunghezza massima della stringa. La differenza sta proprio qui: essendo la dimensione massima della stringa predichiarata, questa occupa in memoria una quantità costante di locazioni di memoria, e qualsiasi variazione di valore produce una riscrittura del nuovo valore sopra il vecchio, senza condurre ad alcun spostamento della variabile nella memoria.

Il programma di figura 6 è analogo a quello utilizzato nell'esempio precedente, ma utilizza la tecnica ora descritta. Il formato di memorizzazione delle matrici di stringhe è illustrato nella seconda figura a pagina 124 del manuale. In figura 7 lo stesso disegno viene particolarizzato al caso della variabile "a\$" dell'esempio di figura 6.

Abbiamo visto come sia possibile in condizioni particolari andare a leggere e scrivere nelle variabili di un programma Basic. Per far ciò abbiamo dovuto a lungo parlare di come vengano gestite le variabili nello Spectrum. Speriamo di non avervi troppo annoiato con lunghe descrizioni teoriche. Del resto anche questo argomento ci sembra almeno altrettanto interessante quanto il nostro obiettivo primario. La prossima volta termineremo questa breve rassegna di tecniche per il passaggio di parametri, introducendo qualcosa che al livello di linguaggio macchina possa riprodurre la funzione svolta dalle linee DATA nel Basic.

MC



electronic devices srl

Via Ubaldo Comandini, 49 (Romanina, Il Università) 00173 Roma
Tel. 06/6132394-6132619-2562757 Tx 616248 Eldev-I

"FRIENDLY" MAIL SERVICE
VENDITA DIRETTA E PER CORRISPONDENZA

DIVISIONE INFORMATICA

NOVITÀ NOVITÀ:

LINEA LITHIUS 286 PC/AT COMPATIBILI

**SISTEMI OPERATIVI: XENIX MULTI UTENTE
MS-DOS 3.0 MS-DOS 2.0/2.11**

Hardware (Versione Base): Piastra madre con microprocessore INTEL 80286 6 MHz (8 MHz), 8 slot AT/PC Compatibili, 512 KRam "on board" Ram 41256

1 Disk driver 1,2 MByte formattati
Controller per 2 driver e 2 Winchester da 20 MByte
Alimentatore 192 W

1 Uscita seriale RS 232/1 Uscita parallela Centronics IBM
Tastiera PC/AT Compatibile (3 led)
Scheda RGB o Monocromatica alta risoluzione
Monitor 12" (opzionale 14") monocromatico alta risoluzione
Ingresso TTL o RGB.

Manualistica (5 manuali), sistemi operativi DOS 3.0 (opzionale XENIX)

Prezzo Lit. 6.000.000

Versione "Enhanced": come versione base, ma con WINCHESTER SLIM da 20 MByte e scheda multifunzione con orologio, n. 2 seriali, n. 1 parallela, 2 MByte possibile espansione.

Prezzo Lit. 7.700.000

LINEA LITHIUS PC/XT (COMPATIBILI MS DOS CPM 86 E CCPM 86)

Modello PC/1 256 KRam, 1 driver 360 K TEAC, controller per driver, alimentatore 130 W, scheda multifunzione, scheda RGB, monitor 14" alta risoluzione fosfori verdi o ambr.

Prezzo Lit. 2.400.000

Modello PC/2 come PC/1, ma

con 2 Driver Teac 360 K Lit. 2.720.000

Modello PC/XT come PC/1, ma con Hard-Disk 10 MByte

Slim Teac SD-510 e controller per Hard Disk

Prezzo Lit. 4.700.000

— **Terminali Ampex** per AT e compatibili

— **Kit di espansione RAM**, 256K Ram NEC 150 ns per AT e compatibili

Prezzo Lit. 72.000

— **Kit di espansione 64KRAM**, 4164 NEC 150 ns.

Prezzo Lit. 36.000

EXPANSION CARDS PER LITHIUS PC/XT

Multifunction 256K (0K RAM), n. 1 parallela Centronics, n. 1 RS 232, clock con batteria

..... Lit. 280.000

Multifunction 384K (0K RAM), n. 1 RS 232, n. 1 parallela centronics, clock con batteria

..... Lit. 352.000

Mother Board 8 slot 128K RAM Lit. 527.000

Monochrome grafic, alta risoluzione,

card con printer Lit. 357.000

RGB Card/Monochrome, Grafic Card Lit. 240.000

PC super modem (300-1200 baud. CCITT V21, V23/ Bell-103 e 202) Lit. 450.000

Driver controller per 4 driver Lit. 125.000

Cabinet metallo like IBM, apribile a compasso con pulsanti cromati, completo di parti meccaniche e frontalino

..... Lit. 134.000

Driver slim 360KByte Teac 55D Lit. 345.000

Alimentatore 135W con ventola,

4 connettori uscita Lit. 250.000

Tastiera ergonomica compatibile PC/XT con led Caps Lock/Min. Locks (scritte italiane o americane)

..... Lit. 170.000

Kits Winchester 10 e 20MByte formattati, slim, meccanica

Teac (Host Adapter Xebec, completo di manuali, software,

Installabile da chiunque in pochi minuti e in qualsiasi compatibile

..... Lit. 1.950.000

NOVITÀ!!! Leasing Hardware + software con ulteriori speciali sconti

STAMPANTI

Panasonic KX 1091: F/T, ingresso Centronics, letter quality

80 colonne, 120 CPS Lit. 750.000

KDC-FT 8000, 7 colori, 180cps, 136 colonne, Frizion/trattor,

letter quality 69 caratteri/secondo, matrice 20X18, ingresso

parallelo Centronics, opzionale RGB con dump grafico a

colori dello schermo, compatibile IBM, ottima per office

automation, cad Lit. 1.600.000

PERIFERICHE MONITORS ADI, PHILIPS - STAMPANTI PANASONIC IBM COMPATIBILI, STAMPANTI LASER.

Floppy disk CIS: per APPLE e per PC e AT SS/DD o DS/DD e DS/DD alta densità per PC e AT.

Prezzi a partire da lire 2000 per SS/SD

Importazione diretta/linea assemblaggio e burn in/compontistica.

LINEA LITHIUS A (APPLE DOS E PRO-DOS COMPATIBILI)

ELABORATORI

Modello LP48/TI Tastiera intelligente con elaboratore 48K

RAM, tastierino numerico, tasti funzionali Basic, 8 slot,

alimentatore 5A Lit. 540.000

Modello LP64/TI Tastiera intelligente, con elaboratore 64K

RAM, tastierino numerico, tasti funzioni Basic, CPM, 8 slot,

alimentatore 5A Lit. 580.000

Modello P2-64/TI Elaboratore Dual Processor (6502 + Z80)

compatibile DOS e CP/M, 64K Byte, tastiera intelligente con

tastierino numerico e tasti funzione Basic, 8 slot, alimentatore

5A Lit. 690.000

Modello P2-64K/TS Come P2-64K, ma con tastiera separata

a 84 tasti e tastierino numerico, elegante contenitore in

metallo con possibilità di incorporare 2 driver Slim Line o

Full Size Lit. 890.000

Novità!!!

Modello E-64/TI Elaboratore compatibile PRO-DOS, Apple

Works con 64K RAM, 7 slot di sistema e 1 di espansione 80

colonne con scheda 80 colonne già inserita Lit. 750.000

Note: la compatibilità PRO DOS è totale e ottenuta da pro-

getto e non tramite modifiche casarecce e precarie sulle

Eprom del Firmware!!!

Novità!!!

Modello E-64/TS Tastiera separata: come E-64 ma con elegante ed ergonomica tastiera indipendente con 83 o 96 tasti

e tastierino numerico, cabinet in metallo con possibilità di driver incorporati

..... Lit. 850.000

SISTEMI

Novità!!!

STARTER 1: Elaboratore Lithius P 48/F1 + Driver Slim

Super 5 trazione diretta, meccanica Chinon + driver controller, monitor Philips PCT 1204 + confezioni di 10 dischi

..... Lit. 1.200.000

STARTER 2 come Starter 1, ma con elaboratore Lithius P2-64/TI

..... Lit. 1.300.000

STARTER 3 OFFICE AUTOMATION

Elaboratore, Lithius P2-64/TI + Slim Driver trazione diretta

Super 5 monitor 12" verde o ambrata alta risoluzione +

interfaccia driver + interfaccia stampante + stampante

Panasonic KX-1091 letter quality 120 CPS + corso Word

Processing/ Spread Sheet/ Data Base Lit. 2.700.000

Note: sono possibili altre configurazioni di sistemi a richiesta del cliente. Consulenza sistemistica gratuita!!!

INTERFACCIE E PERIFERICHE

Interfaccia 2 driver Lit. 63.000

Interfaccia grafica Epson + cavo Lit. 80.000

Interfaccia parallela Centronics Lit. 64.000

Interfaccia RS 232 Lit. 75.000

Interfaccia RS232C Lit. 170.000

Interfaccia Via card 6522 Lit. 70.000

Interfaccia 16K Ram Lit. 80.000

Interfaccia Z 80 (CPU 1 MHz) Lit. 70.000

Interfaccia Z 80 (CPU 4 MHz) Lit. 90.000

Interfaccia 80 Colonne Soft Switch Lit. 100.000

Interfaccia Pal card Lit. 80.000

Interfaccia Super serial Lit. 170.000

Interfaccia Modem card

CCITT V21 300 B Lit. 170.000

Interfaccia ICE 6502 card Lit. 232.000

Interfaccia ICE Z 80 card Lit. 270.000

Paddle per Apple (manopole) Lit. 25.000

8088 card + software Lit. 300.000

Accelerator card (6402 a 4 MHz) Lit. 300.000

Driver Slim Super 5 trazione diretta meccanica Chinon

..... Lit. 350.000

CONDIZIONI DI FORNITURA

Tutti i prezzi salvo diversa indicazione si intendono IVA esclusa e F.co ns. Magazzino. Accettiamo ordini solo per iscritto. Specificare Codice Fiscale o Partita IVA.

Pagamenti effettuabili con: vaglia postale, assegno circolare o assegno postale o contrassegno intestandoli a ELECTRONIC DEVICES Srl Via Ubaldo Comandini, 49 - 00173 Roma

PER FAVORE, NON INVIARE DENARO CONTANTE. Incas- seremo gli assegni solo a spedizione effettuata. Le spese di spedizione saranno addebitate alla consegna.

Consegna immediata al ricevimento ordine (per merce disponibile in magazzino).

I prezzi indicati non subiranno variazioni per almeno 30 gg. Visitate i n/s uffici! Per informazioni telefonare tutti i giorni al: 06/6132394 - 6132619 - 2562757 (chiedere della Divisione Informatica).

Chiedere catalogo illustrato gratuito

NOTE: I prezzi relativi a detto listino sono stati stilati in base al cambio del USS = 1.950 e sono quindi soggetti a variazioni.

Sconti particolari ai Sigg. rivenditori, software house, ricercatori, studenti, professionisti.

Super sconti "prima macchina"!!!

Cerchiamo esclusivisti e rappresentanti in tutta Italia.

Pubblicità gratuita per gli esclusivisti di zona!

PER IL TUO COMPUTER

GIOCHI
&
UTILITY

su cassetta!

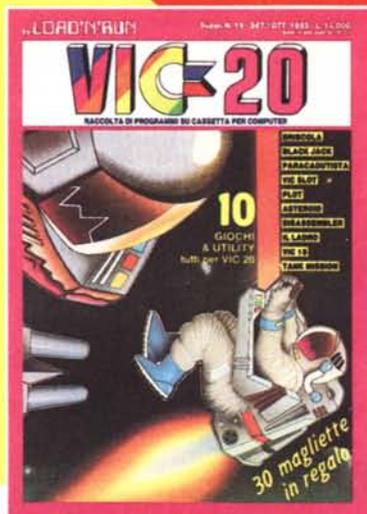
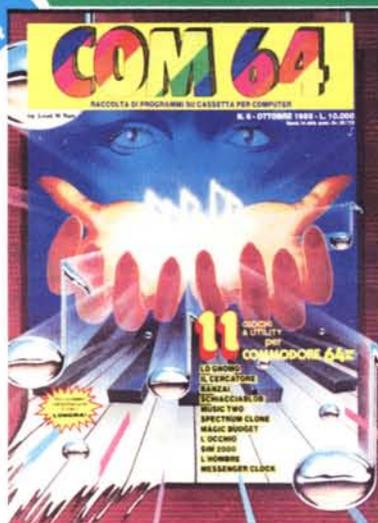
speciale
MSX



per il tuo
Spectrum



con il
COM64



se hai il
VIC 20

in tutte
le edicole

Può essere che nella tua edicola una delle raccolte sia esaurita. In tal caso chiedila direttamente (con un vaglia postale ordinario di lire 10.000 ad Arcadia, C.so Vitt. Emanuele 15, 20122 Milano) specificando naturalmente quale raccolta vuoi. Non dimenticare di indicare il proprio nome e il proprio indirizzo completi!

software MSX

Snow sweet snow

di Gianluca Barfi - Frosinone

Ho da poco acquistato un computer MSX, che trovo molto potente e veloce. La completezza del suo Basic mi ha "spinto" alla programmazione; ho così realizzato questo arcade-sportivo ambientato nel mondo degli sport invernali. Il gioco consiste nel dare velocità a uno sciatore pronto a partire da un trampolino di lancio. Lo sciatore parte al momento della pressione della barra spaziatrice, e la sua velocità dipende dalla posizione assunta in quel momento dall'indicatore visibile in alto sullo schermo. Ad influire sulla lunghezza del salto vi è, oltre alla velocità iniziale, anche la forza del vento, che è sempre a favore e può far guadagnare alcuni metri. La gara parte con una prova di qualificazione, costituita da 5 salti; se la media è superiore alla misura di qualificazione si passa ad una nuova serie di cinque salti.

Le misure ottenute nei salti vengono visualizzate su un tabellone che alla fine renderà nota la media.

Naturalmente vi sono i soliti elogi per i recordmen, incoraggiamenti vari e così via.

L'unico comando da usare è, come già accennato, la barra spaziatrice, che va premuta al momento giusto per fermare l'indicatore di velocità. Il programma occupa circa 4 kbyte di Ram.

Analisi del listato:

60-110 Disegno archi di cerchio per la presentazione

250-260 Stampa le istruzioni

340-360 Disegna montagne

380-450 Disegna trampolino

520-550 Disegna pubblico

560-630 Legge i DATA relativi agli sprite

940-980 Movimento sciatore

1020 Stampa la misura del salto

1030-1050 Uscita sciatore dallo schermo

1100-1170 Movimento dell'indicatore rilevamento della pressione della barra spaziatrice.

1190-1200 Calcola e stampa la media

1210-1220 Elogi per i recordmen

1260 Fine gioco

1280-1330 DATA di definizione degli sprite

Snow sweet snow

```

10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900

```

(continua a pagina 124)


```

430 GOSUB730:GOSUB460:F=VAL("&H"+A$):GOSUB480:GOTO430
440 INPUT I$:IFI$=""THEN I$=""
450 A$=I$
460 IFA$="m"DRAS="M"THEN RETURN100
470 RETURN
480 GOSUB790:IFTTHEN510
490 IF(F<0)OR(F>255)THEN510
500 POKE I,F:PRINTCHR$(30):GOSUB520:GOSUB690:RETURN
510 PRINT"Errore":BEEP:RETURN
520 F=PEEK(I):A$=CHR$(F):GOSUB670
530 PRINTUSING"#####";I:PRINTTAB(7):GOSUB710:PRINTTAB(11):";TAB(14):PRINTUS
ING"#####";F:PRINTTAB(19):GOSUB720:PRINTTAB(23):A$:RETURN
540 CLS:PRINT"Memoria Hex":PRINT:GOSUB280
550 FORJ=1TO23
560 GOSUB710:PRINTTAB(5):";":
570 FORH=1TO8:PRINTTAB(4+H*3):GOSUB720:GOSUB690:NEXT:PRINTCHR$(13):NEXT
580 A$=INKEY$:IFA$="" THEN550
590 GOSUB460:GOTO580
600 CLS:PRINT"Memoria interpretata":PRINT:GOSUB280
610 FORJ=1TO23
620 GOSUB710:PRINTTAB(4):";":
630 FORH=1TO32:F=PEEK(I):GOSUB690:A$=CHR$(F):GOSUB670
640 PRINTA$:NEXT:PRINTCHR$(13):NEXT
650 A$=INKEY$:IFA$="" THEN610
660 GOSUB460:GOTO 650
670 IF(F<32)OR(A$="7F")THEN A$=""
680 RETURN
690 I=I+1:IFI>65535:THEN I=0
700 RETURN
710 PRINTRIGHT$("000"+HEX$(I),4):RETURN
720 PRINTRIGHT$("0"+HEX$(PEEK(I)),2):RETURN
730 PRINTUSING"#####";I:PRINTTAB(7):GOSUB710:PRINTTAB(11):";INPUTA$:IFA$=""
THEN A$=""
740 RETURN
750 PRINTTAB(12):"Premi un tasto";
760 A$=INKEY$:IFA$="m"DRAS="M"THEN RETURN100
770 IFA$=""THEN760
780 RETURN
790 S1$="0123456789abcdef":S2$="0123456789ABCDEF"
800 FORP=LEN(A$)TO1STEP-1
810 FORQ=1TO16:IFMID$(S1$,Q,1)<>MID$(A$,P,1)ANDMID$(S2$,Q,1)<>MID$(A$,P,1)THEN T=
1:NEXTQ:ELSET=0:GOTO16:NEXTQ
820 IFTTHENP=1
830 NEXTP:RETURN
840 T=0:FORP=LEN(I$)TO1STEP-1
850 C$=MID$(I$,P,1):IFC$<"0"ORC$>"9"THEN T=1:P=1
860 NEXTP:RETURN
870 CLS:PRINTTAB(10):"DISASSEMBLER Z80":PRINT:PRINT:GOSUB280
880 FORK=1TO23:GOSUB930
890 PRINTRIGHT$("000"+HEX$(I),4):TAB(7):A$:TAB(18):M$
900 I=I+1:IF I=0:IFI>65535:THEN PRINT"Fine memoria":I=0:K=23:NEXTELSENEXT
910 A$=INKEY$:IFA$="m"DRAS="M"THEN100
920 IFA$<">" THEN910:ELSE930
930 GOSUB1150
940 IFA$="CB"THEN RESTORE1440:I=I+1:GOSUB1150:GOSUB1080:I=I-1:IFB$<">" THENB=B+1:
A$="CB"+A$:RETURNELSEA$="CB":RETURN
950 IFA$="DD"THEN IFHEX$(PEEK(I+1))="CB"THEN1120:ELSE RESTORE1600:I=I+1:GOSUB1150:G
OSUB1080:I=I-1:IFB$<">" THENB=B+1:A$="DD"+A$:RETURNELSEA$="DD":RETURN
960 IFA$="ED"THEN RESTORE1650:I=I+1:GOSUB1150:GOSUB1080:I=I-1:IFB$<">" THENB=B+1:
A$="ED"+A$:RETURNELSEA$="ED":RETURN
970 IFA$="FD"THEN IFHEX$(PEEK(I+1))="CB"THEN1130:ELSE RESTORE1690:I=I+1:GOSUB1150:G
OSUB1080:I=I-1:IFB$<">" THENB=B+1:A$="FD"+A$:RETURNELSEA$="FD":RETURN
980 D$=VAL("&H"+A$)
990 IFD<32THEN RESTORE1280:GOTO1070
1000 IFD<64THEN RESTORE1300:GOTO1070
1010 IFD<96THEN RESTORE1320:GOTO1070
1020 IFD<128THEN RESTORE1340:GOTO1070
1030 IFD<160THEN RESTORE1360:GOTO1070
1040 IFD<192THEN RESTORE1380:GOTO1070
1050 IFD<224THEN RESTORE1400:GOTO1070
1060 RESTORE1420
1070 GOSUB1080:RETURN
1080 READB$:IFLEFT$(B$,2)=A$THEN GOSUB1160:RETURN
1090 IFB$<">" THEN1080
1100 M$="?"
1110 RETURN
1120 RESTORE1630:I=I+3:GOSUB1150:GOSUB1080:I=I-3:IFB$<">" THENGOSUB1140:B=B+2:A$
="DDCB"+RIGHT$("0"+HEX$(PEEK(I+2)),2)+RIGHT$("0"+HEX$(PEEK(I+3)),2):RETURNELSEA$
="DD":RETURN
1130 RESTORE1720:I=I+3:GOSUB1150:GOSUB1080:I=I-3:IFB$<">" THENGOSUB1140:B=B+2:A$
="FDCB"+RIGHT$("0"+HEX$(PEEK(I+2)),2)+RIGHT$("0"+HEX$(PEEK(I+3)),2):RETURNELSEA$
="FD":RETURN
1140 M1$=RIGHT$("0"+HEX$(PEEK(I+2)),2):M$=MID$(B$,3):M$=LEFT$(M$,INSTR(M$,"!"))-1
)+M$+M1$+":":RETURN
1150 A$=RIGHT$("0"+HEX$(PEEK(I)),2):RETURN
1160 B1$="":B2$=""
1170 FORJ=3TO(LEN(B$)):E$=MID$(B$,J,1)
1180 IFF$="#"THENB=B+1:B1$=RIGHT$("0"+HEX$(PEEK(I+1)),2):A$=A$+B1$:C=J-1:J=LEN(B
$):GOTO1230
1190 IFE$="%"THENB=B+1:B1$=RIGHT$("0"+HEX$(PEEK(I+1)),2):A$=A$+B1$:C=J-1:J=LEN(B
$):GOSUB1260:NEXT:M$=MID$(B$,3,C-2)+M$:RETURN
1200 IFE$="@"THENB=B+2:B1$=RIGHT$("0"+HEX$(PEEK(I+1)),2):B2$=RIGHT$("0"+HEX$(PEE
K(I+2)),2):A$=A$+B1$+B2$:C=J-1:J=LEN(B$):GOTO1230
1210 IFE$="!"THENB=B+1:B1$=RIGHT$("0"+HEX$(PEEK(I+1)),2):A$=A$+B1$:E=INSTR(J,B$,
"#!"):IF E THENB=B+1:B2$=RIGHT$("0"+HEX$(PEEK(I+2)),2):A$=A$+B2$:M$=MID$(B$,3,C-2)+
M$+B1$+":":A$+B2$:J=LEN(B$):NEXT:RETURN:ELSEC=J-1:J=LEN(B$):GOTO1230
1220 C=J
1230 NEXT
1240 IFB1$=""ANDB2$=""THEN S$=""ELSE S$=" "
1245 M$=MID$(B$,3,C-2)+S$+B2$+B1$+MID$(B$,C+2)
1250 RETURN
1260 IFE$=">"THEN SR$=RIGHT$("000"+HEX$(PEEK(I+1)+I+2),4):RETURN
1270 SR$=RIGHT$("000"+HEX$(PEEK(I+1)+I-254),4):RETURN
1280 DATA@00DF,01LD BC,0,02LD (BC),A,03INC BC,04INC B,05DEC B,06LD B,07RLCA,08
EX AF,AF,09ADD HL,BC,0ALD A,(BC),0BDEC BC,0CINC C,0DDEC C,0ELD C,0FRRCA
1290 DATA10DJNZ X,11LD DE,0,12LD (DE),A,13INC DE,14INC D,15DEC D,16LD D,0,17RLA,
18JR Z,19ADD HL,DE,1ALD A,(DE),1BDEC DE,1CINC E,1DDEC E,1ELD E,0,1FFRA
1300 DATA20JR NZ,X,21LD HL,0,22LD (H),HL,23INC HL,24INC H,25DEC H,26LD H,0,27DAA
,28JR Z,X,29ADD HL,HL,2ALD HL,(H),2BDEC HL,2CINC L,2DDEC L,2ELD L,0,2FCPL
1310 DATA30JR NC,X,31LD SP,0,32LD (H),A,33INC SP,34INC (HL),35DEC (HL),36LD (HL)

```

(continua a pagina 126)

te che non tutti i caratteri grafici sono rappresentabili.

Questa opzione è utilissima per esaminare la memoria "in modo fine". È possibile individuare scritte all'interno di un programma anche in linguaggio macchina.

Si può esaminare la struttura di una linea Basic e individuarne i TOKEN.

La doppia numerazione (decimale-esadecimale) è utilissima per ricavarci, per esempio degli indirizzi da utilizzare con istruzioni del tipo Peek e Poke senza dover usare le funzioni di conversione.

2) Visualizzazione memoria (input esadecimale)

Opzione identica alla precedente, dalla quale differisce solo perché richiede l'indirizzo di partenza in esadecimale.

3) Scrittura in memoria (input decimale)

Questa opzione può essere usata per scrivere programmi in linguaggio macchina.

Sia l'indirizzo iniziale che i dati devono essere in decimale. La presentazione avviene su cinque colonne come per le opzioni 1 e 2.

Per tornare al menu battere "m" e Return.

4) Scrittura in memoria (input esadecimale)

Opzione identica alla precedente, dalla quale differisce solo perché richiede indirizzo iniziale e dati in esadecimale.

5) Memoria Hex

Con questa opzione la memoria viene visualizzata in esadecimale a pagine di otto byte per riga.

Ovviamente l'indirizzo di partenza bisogna darlo in esadecimale.

6) Memoria interpretata

Anche con questa opzione l'indirizzo di partenza viene dato in esadecimale.

La memoria viene interpretata, cioè vengono mostrati i caratteri ASCII (ed MSX) corrispondenti ai vari byte (come già detto a proposito dell'opzione 1).

Questa opzione serve per individuare scritte all'interno di un programma ed i dati vengono mostrati a paginate di trentadue per riga.

Questo programma è disponibile su cassetta presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 151.

(segue da pagina 125)

```

.#,37SCF,38JR C.X,39ADD HL.SP,3ALD A.(#),3BDEC SP,3CINC A,3DDEC A,3ELD A.#,3FCFF
1320 DATAA0LD B.B,41LD B.C,42LD B.D,43LD B.E,44LD B.H,45LD B.L,46LD B.(HL),47LD
B.A,48LD C.B,49LD C.C,4ALD C.D,4BLD C.E,4CLD C.H,4DL D.C.L,4ELD C.(HL),4FLD C.A
1330 DATA5LD D.B,51LD D.C,52LD D.D,53LD D.E,54LD D.H,55LD D.L,56LD D.(HL),57LD
D.A,58LD E.B,59LD E.C,5ALD E.D,5BLD E.E,5CLD E.H,5DL D.L,5ELD E.(HL),5FLD E.A
1340 DATA6LD H.B,61LD H.C,62LD H.D,63LD H.E,64LD H.H,65LD H.L,66LD H.(HL),67LD
H.A,68LD L.B,69LD L.C,6ALD L.D,6BLD L.E,6CLD L.H,6DL D.L,6ELD L.(HL),6FLD L.A
1350 DATA7LD (HL).B,71LD (HL).C,72LD (HL).D,73LD (HL).E,74LD (HL).H,75LD (HL).L
,76HALT,77LD (HL).A,78LD A.B,79LD A.C,7ALD A.D,7BLD A.E,7CLD A.H,7DL D.A.L,7ELD A
.(HL),7FLD A.A
1360 DATA8ADD A.B,81ADD A.C,82ADD A.D,83ADD A.E,84ADD A.H,85ADD A.L,86ADD A.(HL)
,87ADD A.A,88ADD A.C,89ADD A.C,8AADD A.C,8BADD A.C,8CADD A.H,8DADD A.L,8EADD A.
(HL),8FADD A.A
1370 DATA9SUB B,91SUB C,92SUB D,93SUB E,94SUB H,95SUB L,96SUB (HL),97SUB A,98SB
C A.B,99SBC A.C,9ASBC A.D,9BSBC A.E,9CSBC A.H,9DSBC A.L,9ESBC A.(HL),9FSBC A.A
1380 DATAAAND B,A1AND C,A2AND D,A3AND E,A4AND H,A5AND L,A6AND (HL),A7AND A,ABXO
R B,AXOR C,AXOR D,ABXOR E,ACXOR H,ADXOR L,AXOR (HL),AFXOR A
1390 DATAAOR B,B1OR C,B2OR D,B3OR E,B4OR H,B5OR L,B6OR (HL),B7OR A,B8CP B,B9CP
C,BACP D,BBCP E,BCCP H,BDCP L,BECP (HL),BFCP A
1400 DATAORRET NZ,C1POP BC,C2JP NZ.@,C3JP @,C4CALL NZ.@,C5PUSH BC,C6ADD A.#,C7RS
T #0,C8RET Z,C9RET,CAJP Z.@,CCALL Z.@,CCALL @,CEADD A.#,CFRST #8
1410 DATADRET NC,D1POP DE,D2JP NC.@,D3JP @,D4CALL NC.@,D5PUSH DE,D6SUB #,D
7RST #10,DBRET C,D9EXE,DAJP C.@,DBIN A.(#),DCCALL C.@,DESB A.#,DFRST #18
1420 DATAEORP PO,E1POP HL,E2JP PO.@,E3EX (SP),HL,E4CALL PO.@,E5PUSH HL,E6AND #,
E7RST #20,E8RET PE,E9JP (HL),EAJP PE.@,EBEX DE.HL,ECCALL PE.@,EEXOR #,EFRST #28
1430 DATAFRET P,FIPOP AF,F2JP P.@,F3DI,F4CALL P.@,F5PUSH AF,F6OR #,F7RST #30,FB
RET M,F9LD SP.HL,FAJP M.@,FBFI,FCCALL M.@,FECF #,FFRST #38,#
1440 DATAORLC B,01RLC B,02RLC C,03RLC D,04RLC H,05RLC L,06RLC (HL),07RLC A,08RR
C B,09RR C,0ARRC D,0BRR C,0CRR C,0DRRC H,0DRRC L,0ERRC (HL),0FRRC A
1450 DATA10RL B,11RL C,12RL D,13RL E,14RL H,15RL L,16RL (HL),17RL A,18RR B,19RR
C,1ARR D,1BRR E,1CRR H,1DRR L,1ERR (HL),1FRR A
1460 DATA20SLA B,21SLA C,22SLA D,23SLA E,24SLA H,25SLA L,26SLA (HL),27SLA A,28SR
A B,29SRA C,2ABRA D,2BSRA E,2CSRA H,2DSRA L,2ESRA (HL),2FSRA A
1470 DATA3SRL B,3PSRL C,3ASRL D,3BSRL E,3CSRL H,3DSRL L,3ESRL (HL),3FSRL A
1480 DATA4ORBIT 0.B,41BIT 0.C,42BIT 0.D,43BIT 0.E,44BIT 0.H,45BIT 0.L,46BIT 0.(
HL),47BIT 0.A,48BIT 1.B,49BIT 1.C,4ABIT 1.D,4BBIT 1.E,4CBIT 1.H,4DBIT 1.L,4EBIT 1.
(HL),4FBIT 1.A
1490 DATA50BIT 2.B,51BIT 2.C,52BIT 2.D,53BIT 2.E,54BIT 2.H,55BIT 2.L,56BIT 2.(HL)
,57BIT 2.A,58BIT 3.B,59BIT 3.C,5ABIT 3.D,5BBIT 3.E,5CBIT 3.H,5DBIT 3.L,5EBIT 3.
(HL),5FBIT 3.A
1500 DATA60BIT 4.B,61BIT 4.C,62BIT 4.D,63BIT 4.E,64BIT 4.H,65BIT 4.L,66BIT 4.(HL)
,67BIT 4.A,68BIT 5.B,69BIT 5.C,6ABIT 5.D,6BBIT 5.E,6CBIT 5.H,6DBIT 5.L,6EBIT 5.
(HL),6FBIT 5.A
1510 DATA70BIT 6.B,71BIT 6.C,72BIT 6.D,73BIT 6.E,74BIT 6.H,75BIT 6.L,76BIT 6.(HL)
,77BIT 6.A,78BIT 7.B,79BIT 7.C,7ABIT 7.D,7BBIT 7.E,7CBIT 7.H,7DBIT 7.L,7EBIT 7.
(HL),7FBIT 7.A
1520 DATA80RES 0.B,81RES 0.C,82RES 0.D,83RES 0.E,84RES 0.H,85RES 0.L,86RES 0.(HL)
,87RES 0.A,88RES 1.B,89RES 1.C,8ARES 1.D,8BRES 1.E,8CRES 1.H,8DRES 1.L,8ERES 1.
(HL),8FRES 1.A
1530 DATA90RES 2.B,91RES 2.C,92RES 2.D,93RES 2.E,94RES 2.H,95RES 2.L,96RES 2.(HL)
,97RES 2.A,98RES 3.B,99RES 3.C,9ARES 3.D,9BRES 3.E,9CRES 3.H,9DRES 3.L,9ERES 3.
(HL),9FRES 3.A
1540 DATAA0RES 4.B,A1RES 4.C,A2RES 4.D,A3RES 4.E,A4RES 4.H,A5RES 4.L,A6RES 4.(HL)
,A7RES 4.A,ABRES 5.B,ABRES 5.C,AARES 5.D,ABRES 5.E,ACRES 5.H,ADRES 5.L,AERES 5.
(HL),AFRES 5.A
1550 DATAB0RES 6.B,B1RES 6.C,B2RES 6.D,B3RES 6.E,B4RES 6.H,B5RES 6.L,B6RES 6.(HL)
,B7RES 6.A,B8RES 7.B,B9RES 7.C,BARES 7.D,BBRES 7.E,BCRES 7.H,BDRES 7.L,BERES 7.
(HL),BFRES 7.A
1560 DATAC0SET 0.B,C1SET 0.C,C2SET 0.D,C3SET 0.E,C4SET 0.H,C5SET 0.L,C6SET 0.(HL)
,C7SET 0.A,C8SET 1.B,C9SET 1.C,CASET 1.D,CBSET 1.E,CCSET 1.H,CDSET 1.L,CESET 1.
(HL),CFSET 1.A
1570 DATAD0SET 2.B,D1SET 2.C,D2SET 2.D,D3SET 2.E,D4SET 2.H,D5SET 2.L,D6SET 2.(HL)
,D7SET 2.A,D8SET 3.B,D9SET 3.C,DASET 3.D,DBSET 3.E,DCSET 3.H,DDSET 3.L,DESET 3.
(HL),DFSET 3.A
1580 DATAE0SET 4.B,E1SET 4.C,E2SET 4.D,E3SET 4.E,E4SET 4.H,E5SET 4.L,E6SET 4.(HL)
,E7SET 4.A,E8SET 5.B,E9SET 5.C,EASET 5.D,EBSET 5.E,ECSET 5.H,EDSET 5.L,EESSET 5.
(HL),EFSET 5.A
1590 DATAF0SET 6.B,F1SET 6.C,F2SET 6.D,F3SET 6.E,F4SET 6.H,F5SET 6.L,F6SET 6.(HL)
,F7SET 6.A,F8SET 7.B,F9SET 7.C,FASET 7.D,FBSET 7.E,FCSET 7.H,FDSET 7.L,FESET 7.
(HL),FFSET 7.A,#
1600 DATA09ADD IX.BC,19ADD IX.DE,21LD IX.@,22LD (@).IX,23INC IX,29ADD IX.IX,2ALD
IX.(#),2BDEC IX,34INC (IX+),35DEC (IX+),36LD (IX+).#,39ADD IX.SP
1610 DATA46LD B.(IX+),4ELD C.(IX+),46LD D.(IX+),46LD E.(IX+),46LD H.(IX+),4
ELD L.(IX+),70LD (IX+).B,71LD (IX+).C,72LD (IX+).D,73LD (IX+).E,74LD (IX+).H
,75LD (IX+).L,77LD (IX+).A
1620 DATA7ELD A.(IX+),86ADD A.(IX+),BEADC A.(IX+),96SUB (IX+),9ESBC A.(IX+)
,A6AND (IX+),AEXOR (IX+),B6OR (IX+),BECP (IX+),E1POP IX,E3EX (SP),IX,E5PUSH
IX,E9JP (IX),F9LD SP.IX.#
1630 DATA06RLC (IX+),0ERRC (IX+),16RL (IX+),1ERR (IX+),26SLA (IX+),2ESRA (I
X+),3ESRL (IX+),46BIT 0.(IX+),4EBIT 1.(IX+),56BIT 2.(IX+),5EBIT 3.(IX+),66
BIT 4.(IX+),6EBIT 5.(IX+),76BIT 6.(IX+),7EBIT 7.(IX+)
1640 DATA86RES 0.(IX+),8ERES 1.(IX+),96RES 2.(IX+),9ERES 3.(IX+),A6RES 4.(IX
+),AERES 5.(IX+),B6RES 6.(IX+),BERES 7.(IX+),C6SET 0.(IX+),CESET 1.(IX+),D
6SET 2.(IX+),DESET 3.(IX+),E6SET 4.(IX+),EESSET 5.(IX+),F6SET 6.(IX+),FESET
7.(IX+),#
1650 DATAA0IN B.(C),41OUT (C).B,42SBC HL.BC,43LD (@).BC,44NEG,45RET,46IM 0,47LD
I.A,48IN C.(C),49OUT (C).C,4AADC HL.BC,4BLD BC.(@),4DRETI,4FLD R.A
1660 DATA50IN D.(C),51OUT (C).D,52SBC HL.DE,53LD (@).DE,56IM 1,57LD A.1,58IN E.(
C),59OUT (C).E,5AADC HL.DE,5BLD DE.(@),5EIM 2,5FLD A.R
1670 DATA60IN H.(C),61OUT (C).H,62SBC HL.HL,67RRD,68IN L.(C),69OUT (C).L,6AADC H
.L,6FLD,72SBC HL.SP,73LD (@).SP,78INA.(C),79OUT (C).A,7AADC HL.SP,7BLD.SP.(@)
1680 DATAA0LDI,A1CPI,A2INI,ASOUTI,ABLDD,APCPD,AAIND,ABOUDT,B0LDIR,B1CPIR,B2INIR,
B3OTIR,BBLDDR,B9CPDR,BAINDR,BBOTDR,#
1690 DATA9ADD IX.BC,19ADD IX.DE,21LD IX.@,22LD (@).IX,23INC IX,29ADD IX.IX,2ALD
IX.(#),2BDEC IX,34INC (IX+),35DEC (IX+),36LD (IX+).#,39ADD IX.SP
1700 DATA46LD B.(IX+),4ELD C.(IX+),46LD D.(IX+),46LD E.(IX+),46LD H.(IX+),4
ELD L.(IX+),70LD (IX+).B,71LD (IX+).C,72LD (IX+).D,73LD (IX+).E,74LD (IX+).H
,75LD (IX+).L,77LD (IX+).A
1710 DATA7ELD A.(IX+),86ADD A.(IX+),BEADC A.(IX+),96SUB (IX+),9ESBC A.(IX+)
,A6AND (IX+),AEXOR (IX+),B6OR (IX+),BECP (IX+),E1POP IX,E3EX (SP),IX,E5PUSH
IX,E9JP (IX),F9LD SP.IX.#
1720 DATA06RLC (IX+),0ERRC (IX+),16RL (IX+),1ERR (IX+),26SLA (IX+),2ESRA (I
X+),3ESRL (IX+),46BIT 0.(IX+),4EBIT 1.(IX+),56BIT 2.(IX+),5EBIT 3.(IX+),66
BIT 4.(IX+),6EBIT 5.(IX+),76BIT 6.(IX+),7EBIT 7.(IX+)
1730 DATA86RES 0.(IX+),8ERES 1.(IX+),96RES 2.(IX+),9ERES 3.(IX+),A6RES 4.(IX
+),AERES 5.(IX+),B6RES 6.(IX+),BERES 7.(IX+),C6SET 0.(IX+),CESET 1.(IX+),D
6SET 2.(IX+),DESET 3.(IX+),E6SET 4.(IX+),EESSET 5.(IX+),F6SET 6.(IX+),FESET
7.(IX+),#

```

Naturalmente non tutti i caratteri sono rappresentabili.

7) Disassembler Z80

Permette di disassemblare, a partire dall'indirizzo dato in esadecimale, con i mnemonici Z80.

Per poterlo sfruttare naturalmente è necessario conoscere il set di istruzioni dello Z80.

Alcune note: al posto della virgola che separa gli operandi ho messo un punto. Spero che nessuno mi odierà per questo. Ho dovuto farlo perché tutti i codici operativi ed i mnemonici sono messi in DATA e la virgola rappresenta appunto il separatore tra i vari dati. Certo, avrei potuto sostituire il punto con la virgola in fase di ricostruzione dei mnemonici, ma ciò avrebbe rallentato ulteriormente il programma, che essendo in Basic non è molto veloce.

A proposito di velocità: per far sì che il disassemblaggio avvenga ad una velocità accettabile ho fatto largo uso della istruzione RESTORE N. Si veda la riga 990 e le seguenti. In pratica ho diviso i codici operativi in blocchi di trentadue diminuendo così i tempi di ricerca. Ho usato questo accorgimento solo per i codici operativi senza prefisso, che sono quello usati più frequentemente.

8) Ritorno al Basic

Con questa opzione si rientra al Basic senza cancellare il programma che resta comunque disponibile.

Intendiamoci: non è che si possa tornare al Basic e scrivere un bel programmino: si andrebbe a sporcare il programma monitor.

Questa opzione è utile piuttosto per registrare routine in linguaggio macchina o per leggerle con le istruzioni del Basic BSAVE e BLOAD.

Infatti mi è sembrato assolutamente inutile inserire nel programma funzioni del genere visto che quelle dell'interprete Basic funzionano egregiamente, tanto che consentono di trasferire blocchi di memoria da una zona all'altra.

Si procede così: si salva l'area da trasferire con:

BSAVE "cas:", indirizzo inizio blocco, indirizzo fine blocco poi la si rilegge con: BLOAD "cas:", valore dello spostamento.

Esempio: se il blocco era stato registrato a partire dall'indirizzo A000 esadecimale all'indirizzo AFFF esadecimale e il valore dello spostamento dato con la BLOAD è di 1000 esadecimale, il blocco verrà caricato a partire da B000 esadecimale per arrivare a BFFF esadecimale.

Per far partire la routine c'è la funzione MSR.



JUKI 6100. Perché non sono tutte così?

Perché ogni utente ha esigenze diverse. Non tutti, per esempio, hanno bisogno di tutte le prestazioni offerte dalla nostra straordinaria 6100. (Anche se, a giudicare dal fatto che è una delle più vendute in Italia, ne hanno bisogno in molti). E non tutti hanno un computer IBM* (ma per costoro abbiamo appena introdotto la nuovissima stampante grafica 6100-I, IBM compatibile).

A noi non piace costringere il cliente a «prendere o lasciare»: noi adeguiamo la nostra gamma per

soddisfare nel miglior modo ogni esigenza. Perciò, qualsiasi prestazione desideriate dalla vostra stampante, troverete la «vostra» JUKI.

Ma a furia di parlarvi delle differenze, rischiamo di trascurare il fatto importante che almeno due cose non cambiano mai. Qualità e rapporto prezzo/prestazioni, per esempio. Sotto questo aspetto tutte le stampanti JUKI sono uguali.

* IBM è un marchio registrato della IBM Corporation.



JUKI 6100: una delle stampanti a margherita qualità lettera più vendute in Italia. Modo grafico e piena capacità di word processing.



JUKI 6300: stampante a margherita ad alta velocità (48 cps) con memoria buffer di 3 Kbyte e piena capacità di word processing.



JUKI 2200: stampante a margherita pienamente portatile con memoria buffer di 2 Kbyte e piena capacità di word processing. Ideale per uso domestico.



JUKI 5520: stampante a matrice di punti per personal computer, di alta qualità e dal prezzo basso. Alta velocità (180 cps), modo grafico e funzione opzionale per la stampa a quattro colori. Quasi qualità lettera (NLQ).



JUKI 6000: stampante a margherita, qualità lettera, creata specificamente per uso domestico.



JUKI 6100-I: versione nuovissima, IBM compatibile, della best seller 6100. Modo grafico e piena capacità di word processing.

JUKI

Tecnologia di vera stampa

JUKI (EUROPE) GMBH · Eiffestr. 74 · 2000 Amburgo 26 · Germania Occidentale
Tel.: (0 40) 2 51 20 71-73 · Telex: 2 163 061 · Facsimile: (0 40) 2 51 27 24.

Rappresentanza esclusiva: **telecom** 20148 Milano · via M. Civitali 75
tel. 4047648 (r.a.) · tlx 335654 TELCOM I

no posizionate tramite la prima fila di tasti del Vic (escluso il tasto "Del" che conserva la propria funzione).

Per quanto riguarda la durata delle note, il programma viene fornito in due versioni: nella prima (Compositore1) essa dipende da quanto tempo teniamo premuto il tasto in fase di composizione mentre nella seconda — ottenuta sostituendo alla prima versione un certo numero di linee come chiaramente indicato in queste pagine — possiamo scegliere quattro durate prestabilite

selezionabili mediante i tasti f1, f3, f5 e f7.

La schermata che rappresenta il pentagramma è realizzata tramite caratteri ridefiniti ed è limitata a tre righe musicali che possono contenere 22 note ciascuno. A seconda del tasto premuto tra quelli della prima fila, verrà posizionata la corrispondente nota; premendo invece la barra spaziatrice otterremo il simbolo e l'effetto della pausa mentre, premendo qualsiasi altro tasto (naturalmente non della prima fila) otterremo il medesimo risultato, ma senza

la corrispondente rappresentazione grafica, che comunque si otterrà richiamando il pentagramma da altre parti del programma. Possiamo infatti ottenere, una volta entrati nel main menu, premendo il tasto <Return>, la lista delle note (nel codice della macchina, cioè mediante numeri in questo caso compresi tra 195 e 240) e la loro durata premendo f3. Con f5 e f7 avvieremo le operazioni di caricamento e salvataggio dei brani composti su nastro mentre con f1 daremo inizio alla composizione. Mentre componiamo, in qualunque istante potremo riascoltare il motivo realizzato fino a quel punto servendoci di f1.

Come ultimo avvertimento, prima di far girare il programma (se caricato da nastro o disco) o di digitarlo occorre far posto ai caratteri ridefiniti spostando, l'inizio del programma stesso a pagina 22, cioè all'indirizzo decimale 5632, con:

```
POKE 642,22:SYS 58232 <Ret>.
```

Per usufruire della versione 2 del programma, dovremo semplicemente cancellare le linee da 300 a 350 nella versione 1 ed inserire tra queste il segmento riportato nella figura accanto.

MC

Compositore (versione 2)

Figura 1 - Questo è il segmento da inserire tra le linee 250 e 400 nel programma del "compositore 1". Quelle contenute nel riquadro presente nel listato stesso vanno omesse.

```
300 goto$: ifa$<"ona$>" then300
310 ifa$=" " then%(v)=30
311 ifa$=" " then%(v)=60
312 ifa$=" " then%(v)=90
313 ifa$=" " then%(v)=120
320 fort=1tot%(v):poke36876,n%(v):nextt:poke36876,0
330 ifv=21thenf=6
340 ifv=43thenf=12
350 v=v+1:ifv>66thenv=v-1:goto400
360 goto100
```

THE ARCADE

i joystick con il -CLIK- per punteggi da "sballo"

- Joystick professionali con micro-switch ad altissima precisione e sensibilità
- Realizzati dalla SUZO, azienda leader nella produzione di macchine per video games da sala giochi.



Super Stick
L. 99.000



The Arcade TURBO
L. 69.000



The Arcade - beige
L. 49.000

The Arcade - nero
L. 49.000

QLUB

EDIZIONE
ITALIANA

Notiziario QL

COMUNICATO PER TUTTI I POSSESSORI DI QL E PER QUANTI HANNO INTENZIONE A DIVENTARLO

IN OTTOBRE È USCITO
IL PRIMO NUMERO DI
QLUB — NOTIZIARIO QL,
RISERVATO IN ESCLUSIVA
A TUTTI I POSSESSORI DEL QL
CON GARANZIA ITALIANA.

SE STAI PER COMPRARE UN QL
CONTROLLA CHE SIA DOTATO
DELLA GARANZIA ITALIANA:
È L'UNICO MEZZO
PER RICEVERE LA RIVISTA
QLUB — NOTIZIARIO DEL QL.

REBIT
COMPUTER

A DIVISION OF G.B.C.

IN COLLABORAZIONE CON LA:

DISTRIBUTTRICE PER L'ITALIA DEI PRODOTTI:

sindclair

software

COMMODORE 64

Questo mese un software abbastanza vario.

Il primo programma che vi proponiamo, *World*, proietta sul teleschermo un planisfero sul quale avremo la possibilità di esercitarci in geografia. Con il secondo, *Numerologia*, abbiamo a che fare con i numeri, questa volta in maniera un po' particolare: il nostro nome viene ridotto ad un numero e attraverso esso pare sia possibile risalire alla nostra personalità. C'è chi ci crede, c'è chi non ci crede. Comunque il programma è ben fatto ed in ogni caso vi accorgete che, anche se preso come un gioco, *Numerologia* vi diventerà moltissimo.

E per finire... *Life*: esso è capace questa volta di lavorare su una matrice di 256 x 200 elementi.

World

di Massimo e Davide Vincenzi

World è una specie di atlante che permette di individuare su un planisfero la posizione di un enorme numero di stati e loro eventuali dipendenze.

Il programma è diviso in due parti, che abbiamo chiamato *World 1* e *World 2*, di cui la prima funge da caricatore per la seconda. Da cassetta, il tempo di caricamento è circa 6 minuti.

L'uso è molto semplice: viene richiesto di inserire il nome dello stato o della dipendenza da ricercare e se esso è corretto, cioè corrisponde ad uno di quelli contenuti in memoria, un cerchietto bianco con un puntino lampeggiante al centro lo indicherà sul planisfero. Per ricercare un altro stato, basta premere un tasto qualsiasi.

Come si vede dalle linee *Data*, l'elenco degli stati e staterelli è ben fornito e naturalmente noi della redazione non abbiamo verificato l'esatta posizione di tutti (anche perché di alcuni francamente non conosciamo neanche il nome!). Comunque, dai test effettuati, il programma risulta perfettamente funzionante e non c'è stato bisogno di nessuna correzione. Se dovessero esserci problemi, fatecelo sapere.

Si pregano i Sigg. Massimo e Davide Vincenzi, autori del programma "World", di mettersi in contatto con la redazione.



Il planisfero generato dal programma *World*.

Numerologia

di Lorella De Luca - L'Aquila

Un po' di teoria

La numerologia è un'antichissima arte risalente ai Caldei ed ai Babilonesi.

Secondo quest'arte ogni nome si può scindere e ridurre in un solo numero, che è poi quello della personalità, dell'io più profondo. Il nome però ci viene dato alla nascita senza la possibilità di rifiutarlo: per tal motivo, quando la nostra personalità non si trova più a suo agio nel suo nome di battesimo, sia interiormente, sia attraverso il giudizio esteriore di altre persone, interviene a liberarla un nome o un soprannome. Ecco perché nella riduzione di un nome a numero, deve essere considerato il nome con cui tutti ci chiamano perché è in esso che noi trasmettiamo le nostre "energie e vibrazioni".

Per avere un quadro completo di noi stessi, dobbiamo aggiungere nella riduzione a numero anche il nostro cognome: da esso si rilevano i caratteri ereditari che sono contenuti in noi perché appartenenti a quella determinata famiglia.

Funzionamento del programma

Dopo il *Run* apparirà sul video una breve presentazione seguita da un menu che ci metterà a disposizione tre opzioni: la prima ci permetterà di accedere alla sezione in cui il nome introdotto verrà ridotto ad un numero e ci verranno forniti gli elementi essenziali della nostra personalità. La seconda metterà a confronto due numeri (personalità) e dirà se può esservi tra loro accordo e disaccordo. Infine, la terza opzione ci farà uscire dal programma.

Per ulteriori dettagli, basterà seguire le indicazioni fornite dal programma.

Life I.m.

di Piergiorgio Bertoli - Roma

Il programma è una versione, realizzata in linguaggio macchina, dell'ormai conosciuto modello sviluppato da Conway ed è capace di lavorare su matrici fino a 256 x 200 elementi. Con la versione proposta si riescono ad esaminare circa 5000 elementi al secondo costringendo l'utente ad un'attesa massima (cioè nel peggiore dei casi) di 10 secondi. Il codice oggetto del programma occupa 900 byte a partire dalla locazione \$C000.

La possibilità di commettere errori a causa della moltitudine di numeri contenuti nei *Data* è ridotta al minimo da un checksum che verifica se il numero di elementi battuti è esatto. Se poi volete evitare completamente la possibilità di errori, servitevi del programma che effettua il checksum sulle singole linee pubblicato sul numero scorso (a questo scopo i listati vengono pubblicati con un numero alla fine di ciascuna linea preceduto dai due punti).

Note per la copiatura dei listati per il 64

Nel n. 44 (settembre 85) è stato pubblicato un programma di Checksum per aiutare i lettori nella copiatura dei listati per il Commodore 64 pubblicati sulla rivista.

Il funzionamento è il seguente:

- copiate il programma Checksum del n. 44 e salvatelo su disco o cassetta.
- Per la successiva copiatura di un listato (con checksum), caricate (dal vostro disco o dal vostro nastro) il programma di checksum e fatelo partire; a questo punto potete copiare le varie linee del listato, compresi i due punti ed il numero che trovate alla fine di ciascuna riga. Alla pressione del return, se la linea è stata copiata bene si può passare a copiare la successiva, altrimenti il programma di checksum vi lascerà "inchiodati" sulla linea mal copiata obbligandovi a correggere l'errore prima di proseguire.

A quanto detto nel numero 44 riguardo al programma Checksum in questione, aggiungiamo che la routine di checksum in LM si avvia con SYS 52480 mentre, in caso di arresto con Run-Stop/Restore, il restart si effettua con SYS 53072.

Attenzione: chi non vuole usare il checksum, NON DEVE copiare i due punti e il numero alla fine delle righe, pena la segnalazione di "syntax error" da parte del computer.

World 1

```

0 PRINTCHR$(142);PRINTCHR$(8);POKE53280,0;POKE53281,0;2250
1 PRINT"*****POKE*****";FORP=8832TO16151:POKEP,0:NEXT:3891
2 X=8832:482
3 IF=0 THENREDOB:X=X+8:GOTO4:1698
4 IF=25 THENREDOB:FORX=8TOX+8-1:POKEX,255:NEXT:GOTO4:3893
5 IF=5 THENREDOB:FORX=8TOX+8-1:POKEX,255:NEXT:GOTO4:1233
6 POKEX,X+X+1:GOTO4:1233
7 POKEX,X+X+1:GOTO4:1233
8 POKEX,X+X+1:GOTO4:1233
9 POKEX,X+X+1:GOTO4:1233
10 POKEX,X+X+1:GOTO4:1233
11 POKEX,X+X+1:GOTO4:1233
12 POKEX,X+X+1:GOTO4:1233
13 POKEX,X+X+1:GOTO4:1233
14 POKEX,X+X+1:GOTO4:1233
15 POKEX,X+X+1:GOTO4:1233
16 POKEX,X+X+1:GOTO4:1233
17 POKEX,X+X+1:GOTO4:1233
18 POKEX,X+X+1:GOTO4:1233
19 POKEX,X+X+1:GOTO4:1233
20 POKEX,X+X+1:GOTO4:1233
21 POKEX,X+X+1:GOTO4:1233
22 POKEX,X+X+1:GOTO4:1233
23 POKEX,X+X+1:GOTO4:1233
24 POKEX,X+X+1:GOTO4:1233
25 POKEX,X+X+1:GOTO4:1233
26 POKEX,X+X+1:GOTO4:1233
27 POKEX,X+X+1:GOTO4:1233
28 POKEX,X+X+1:GOTO4:1233
29 POKEX,X+X+1:GOTO4:1233
30 POKEX,X+X+1:GOTO4:1233
31 POKEX,X+X+1:GOTO4:1233
32 POKEX,X+X+1:GOTO4:1233
33 POKEX,X+X+1:GOTO4:1233
34 POKEX,X+X+1:GOTO4:1233
35 POKEX,X+X+1:GOTO4:1233
36 POKEX,X+X+1:GOTO4:1233
37 POKEX,X+X+1:GOTO4:1233
38 POKEX,X+X+1:GOTO4:1233
39 POKEX,X+X+1:GOTO4:1233
40 POKEX,X+X+1:GOTO4:1233
41 POKEX,X+X+1:GOTO4:1233
42 POKEX,X+X+1:GOTO4:1233
43 POKEX,X+X+1:GOTO4:1233
44 POKEX,X+X+1:GOTO4:1233
45 POKEX,X+X+1:GOTO4:1233
46 POKEX,X+X+1:GOTO4:1233
47 POKEX,X+X+1:GOTO4:1233
48 POKEX,X+X+1:GOTO4:1233
49 POKEX,X+X+1:GOTO4:1233
50 POKEX,X+X+1:GOTO4:1233

```

```

13892
51 DATA224,248,249,245,241,251,63,63,127,127,255,4,248,252,252,248,255,1,252,249
3810
52 DATA255,1,0,4,128,0,1,128,192,0,28,2,6,6,39,15,15,7,6,0,2,1,1,254,190,56,28,3
546
53 DATA28,148,128,2,51,3,15,39,55,31,31,95,255,57,253,253,249,251,243,231,255,25
3890
54 DATA144,184,216,204,238,200,232,236,6,2,0,1,4,0,44,0,52,1,1,3,63,255,15,251
3734
55 DATA255,7,127,253,253,248,224,248,255,1,248,255,3,127,127,63,159,15,255,3,361
3735
56 DATA254,255,5,252,0,1,201,3,215,201,192,0,1,64,192,128,224,2825
57 DATA192,64,0,24,115,51,119,15,7,14,0,1,3,1,135,135,15,159,63,255,2,95,255,357
58 DATA127,255,38,159,255,7,223,255,6,252,255,6,63,255,40,248,236,230,192,
3735
59 DATA224,228,198,143,11,4,0,1,16,0,1,64,0,1,128,0,41,0,48,3,7,7,3,3,1,1,255,
3677
60 DATA24,253,253,249,251,255,4,199,159,220,241,239,207,255,3,2856
61 DATA254,258,246,224,128,0,2,232,112,192,0,37,1,0,3,31,31,31,255,3,246,224,3
62 DATA192,192,128,255,2,247,115,57,28,143,6,255,5,63,63,181,249,219,209,128,362
63 DATA15,143,159,255,4,253,255,42,253,249,243,248,248,248,240,152,24,28
3735
64 DATA15,143,159,255,4,253,255,42,253,249,243,248,248,248,240,152,24,28
3735
65 DATA284,0,0,1,10,14,15,7,15,31,0,5,128,0,1,128,0,40,0,48,1,1,0,6,255,4,127,35
5817
66 DATA5,51,51,255,24,252,252,240,224,128,0,50,31,31,2,1,7,15,63,160,0,2,
5909
67 DATA2,255,4,544
68 DATA15,156,8,0,1,224,192,224,248,17,0,1,17,0,1,12,0,3,255,3,53,1,25,3,3,255,
3635
69 DATA15,199,239,255,55,254,252,254,255,4,140,14,8,1,131,130,192,192,95,127,196,
3778
70 DATA224,160,128,0,2,128,0,47,0,56,51,19,25,57,28,12,6,0,1,255,5,127,63,31,357
3807
71 DATA255,1,243,224,192,192,224,192,192,139,131,195,1,0,3,3,128,128,128,128
3807
72 DATA16,0,128,0,42,32,4,17,7,3,63,127,127,255,13,252,254,255,6,24,30,159,360
6
73 DATA255,5,7,7,239,229,240,248,254,255,6,127,63,31,255,1,207,231,227,288,282,9-
74 DATA255,6,143,38,96,240,255,5,127,15,1796
75 DATA7,255,37,254,252,248,192,192,128,129,0,1,32,32,32,0,58,32,6,0,60,31,31,15
3788
76 DATA15,0,3,192,234,241,255,2,207,7,3,7,234,192,192,128,128,240,248,224,252
77 DATA240,57,3,97,0,5,128,228,0,36,7,15,31,15,7,7,15,255,40,143,149,199,195,3
78 DATA227,241,243,249,255,5,254,252,240,248,240,192,224,128,0,3,7,2,0,6,255,2,3
79 DATA127,127,63,63,31,30,248,240,240,224,192,128,0,2,255,1,127,127,63,63,55,7,
3675
80 DATA3,255,1,254,252,248,254,255,3,240,64,128,128,0,1,128,128,192,0,3,10,14,28
3779
81 DATA28,6,0,56,0,72,3,0,7,248,120,56,28,7,2,0,4,6,30,223,127,63,63,0,2,192,246
3779
82 DATA255,4,0,4,32,128,224,254,0,24,15,7,15,7,3,3,1,0,1,255,7,127,255,6,231,131
3779
83 DATA251,249,255,22,253,252,254,255,5,240,129,4,220,252,248,248,240,0,16,30,14
3808
84 DATA96,4,1,1,0,6,128,128,0,1,1,1,1,0,3,255,1,223,135,130,128,224,112,32,
192,13578
85 DATA192,128,0,3,1,3,40,2,40,212,2,15,139,226,0,56,0,76,4,4,0,5,1,3,3,1,3,63,3
649
86 DATA127,255,22,128,128,192,192,224,234,255,2,0,7,240,0,16,32,0,7,1,0,7,255,1,1
3779
87 DATA63,31,63,127,127,63,31,255,12,248,249,249,255,6,254,252,248,240,224,192,3
642
88 DATA128,0,36,6,19,9,0,1,8,0,3,48,144,128,192,240,120,56,28,3,7,31,127,63,63,3
644
89 DATA63,30,224,192,128,158,208,156,56,28,0,2,64,96,70,14,2,119,0,5,32,120,124,
5760
90 DATA70,12,0,32,-1,886
100 POKE630,32:POKE631,131:POKE632,13:POKE198,3:2879

```

Questo programma è disponibile su cassetta presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 151.

Dato il Run, se tutto è a posto, si entrerà in hi-res e verrà visualizzato uno sprite indicante il limite inferiore della finestra (matrice) in cui vogliamo far evolvere le generazioni: esso potrà essere mosso verticalmente (precisiamo che in orizzontale la

finestra è sempre composta da 256 punti) servendosi del joystick posto nella porta numero 2. Premendo il fuoco verrà stabilito il limite inferiore della matrice e sarà visualizzato un pennino: per creare la popolazione sarà sufficiente spostarlo — sempre con il joystick — e premere il fuoco ogni volta che vorremo posizionare la singola cellula. Per passare alla fase evolutiva

basterà premere il tasto Restore. In tale fase, il pennino scompare e viene sostituito da un piccolo sprite che, muovendosi sulla destra dall'alto verso il basso, ci indicherà a che punto dell'evoluzione si è giunti ad ogni passaggio.

Per uscire dal programma si preme Run/Stop-Restore e, per rientrarvi, Sys 49153. **MC**

```

1 REM -----
2 REM LIFE
3 REM DI PIERGIORGIO BERTOLI
4 REM -----
10 PRINT "POKE53280,2:POKE53281,2:1659
20 NL=100:AD=49152:L=808:A#="DATI":GOSUB5000:2690
90 REM -----DATI L.M.-----1925
100 DATA24,216,120,169,193,141,25,3,3758:1821
110 DATA169,147,141,24,3,169,94,133,3989:1853
120 DATA255,169,192,133,254,169,0,170,5601:1945
130 DATA168,24,165,254,157,248,199,165,6969:2041
140 DATA255,157,248,200,24,230,254,232,7503:2004
150 DATA200,192,8,208,236,24,230,255,6670:1918
160 DATA230,250,24,165,254,105,56,133,5074:1972
170 DATA254,144,3,24,230,255,24,165,5071:1882
180 DATA254,157,248,199,165,255,157,248,7802:2132
190 DATA200,160,0,224,0,208,202,169,5686:1897
200 DATA96,133,254,169,205,133,255,169,7016:2084
210 DATA207,133,251,32,230,194,169,128,6131:2075
220 DATA162,0,157,0,207,74,232,224,5784:1888
230 DATA8,208,247,173,2,221,9,3,3536:1751
240 DATA141,2,221,173,0,221,41,252,5385:1891
250 DATA9,2,141,0,221,169,104,141,4667:1861
260 DATA24,208,173,17,208,9,32,141,3729:1678
270 DATA17,208,160,0,132,254,132,252,6293:1775
280 DATA169,88,133,255,169,96,133,251,6388:1870
290 DATA32,214,194,169,143,133,251,32,5500:1846
300 DATA230,194,169,1,141,82,207,141,5159:1815
310 DATA83,207,169,129,141,0,208,169,5289:1842
320 DATA149,141,1,208,169,88,141,248,5866:1856
330 DATA91,169,1,141,21,208,169,128,4812:1798
340 DATA133,252,173,0,220,41,2,208,4436:1737
350 DATA3,238,1,208,173,0,220,41,4303:1643
360 DATA1,208,3,206,1,208,173,1,3978:1620
370 DATA208,201,232,208,5,169,36,141,4813:1880
380 DATA1,208,173,1,208,201,35,208,5351:1780
390 DATA5,169,231,141,1,208,169,144,5444:1856
400 DATA133,255,169,150,133,251,32,214,6113:2003
410 DATA194,173,0,220,41,16,208,194,4985:1881
420 DATA238,248,91,173,1,208,24,233,5240:1884
430 DATA20,133,246,32,246,194,165,246,6925:2004
440 DATA74,24,105,4,133,253,105,21,3795:1849
450 DATA141,1,208,169,128,133,252,169,6253:2018
460 DATA128,133,255,169,159,133,251,32,5697:2092
470 DATA214,194,173,0,220,41,3,201,4352:1863
480 DATA3,240,17,201,2,240,8,238,5004:1774
490 DATA1,208,230,253,76,68,193,206,6162:1959
500 DATA1,208,198,253,173,0,220,41,5012:1899
510 DATA12,201,12,240,17,201,8,240,4933:1901
520 DATA8,238,0,208,230,252,76,94,5518:1639
530 DATA193,206,0,208,198,252,165,253,7374:1852
540 DATA201,7,208,12,166,246,202,134,5935:1795
550 DATA253,138,24,105,21,141,1,208,3899:1758
560 DATA165,253,197,246,208,9,169,29,5011:1837
570 DATA141,1,208,169,8,133,253,173,5692:1786
580 DATA8,220,41,16,208,153,32,186,4553:1729
590 DATA194,177,254,29,0,207,145,254,5971:1866
600 DATA76,31,193,169,254,141,25,3,3964:1772
610 DATA169,71,141,24,3,238,248,91,4993:1788
620 DATA169,0,133,252,169,1,141,16,3798:1786
630 DATA208,169,32,141,0,208,166,246,5840:1892
640 DATA202,134,253,32,186,194,240,2,5403:1889
650 DATA169,1,166,252,157,0,202,169,5484:1868
660 DATA8,133,253,32,186,194,240,2,5207:1818
670 DATA169,1,166,252,157,0,203,157,5395:1887
680 DATA8,205,230,253,32,186,194,240,6922:1934
690 DATA2,169,1,166,252,157,0,204,5097:1808
700 DATA230,252,165,252,208,200,133,252,7680:2101
710 DATA133,254,168,169,8,133,253,169,6038:2038
720 DATA32,133,255,24,165,245,233,62,5837:1985
730 DATA133,251,32,230,194,169,37,141,5278:2044
740 DATA1,208,166,252,189,0,203,141,5673:1948
750 DATA82,207,202,24,189,0,202,125,4813:1950
760 DATA8,203,125,0,204,232,24,125,4617:1895
770 DATA8,202,125,0,204,232,24,125,4615:1647
780 DATA8,202,125,0,203,240,26,125,4672:1660
790 DATA8,204,170,189,80,207,240,17,5388:1747
800 DATA32,186,194,24,165,255,233,63,5828:1822
810 DATA133,255,177,254,29,0,207,145,5200:1810
820 DATA254,169,1,141,82,207,230,252,6693:1829
830 DATA165,252,208,182,162,0,230,253,6721:1878
840 DATA230,253,189,0,203,157,0,202,5132:1776
850 DATA189,0,204,157,0,203,134,252,5857:1804
860 DATA32,186,194,240,2,169,1,166,4561:1770
870 DATA252,157,0,204,232,138,208,226,6890:1926
880 DATA133,252,238,1,208,198,253,165,6930:1944
890 DATA253,24,105,1,197,246,240,45,5377:1849
900 DATA165,253,197,246,208,49,160,0,4956:1923
910 DATA132,254,132,252,169,32,133,253,6292:2013
920 DATA169,96,133,255,230,245,177,252,7911:2042
930 DATA145,254,200,208,249,230,253,230,8577:2087
940 DATA255,165,255,197,245,208,239,32,6796:2075
950 DATA246,194,76,174,193,189,0,205,5553:1975
960 DATA157,0,204,232,138,208,246,76,6221:1962
970 DATA2,194,166,253,189,248,199,133,7046:2050
980 DATA254,189,248,200,133,255,165,252,7798:2158
990 DATA41,248,168,165,252,41,7,170,4872:1956
1000 DATA177,254,61,0,207,96,169,3,3942:1868
1010 DATA145,254,200,208,249,230,255,165,8071:2165
1020 DATA255,197,251,208,241,96,169,0,5454:2041
1030 DATA145,254,200,208,249,230,255,165,8071:1930
1040 DATA255,197,251,208,241,96,169,0,5454:1806
1050 DATA133,252,165,246,133,253,32,186,6267:1907
1060 DATA194,169,255,145,254,24,165,252,6718:1931
1070 DATA105,8,133,252,208,240,166,255,7466:1875
1080 DATA232,134,245,32,186,194,230,255,7363:1935
1090 DATA169,128,145,254,198,253,165,253,7819:2020
1100 DATA208,241,96,0,0,0,0,1234:1421
1900 NL=2000:AD=22016:L=192:A#="SPRITE":GOSUB5000:3001
1990 REM-----DATI SPRITE 1-----2077
2000 DATA0,0,0,0,0,0,0,0,256:1239
2010 DATA0,0,0,0,0,0,0,0,256:1249
2020 DATA0,0,15,36,224,8,52,144,3129:1692
2030 DATA14,44,144,8,36,144,15,36,2259:1817
2040 DATA224,0,0,0,31,255,248,0,3901:1701
2050 DATA0,64,0,0,128,0,1,0,1031:1240
2060 DATA0,10,0,0,12,0,0,0,30,576:1200
2070 DATA0,0,24,0,0,32,0,0,520:1155
2075 REM-----DATI SPRITE 2-----1908
2080 DATA0,0,0,14,0,0,29,0,515:1174
2090 DATA0,50,128,0,35,64,0,20,1459:1452
2100 DATA160,0,8,80,0,4,40,0,1064:1353
2110 DATA2,16,0,1,8,0,0,132,1390:1310
2120 DATA0,0,66,0,0,33,0,0,652:1218
2130 DATA17,128,0,10,64,0,4,32,1173:1488
2140 DATA0,3,16,0,0,200,0,0,1510:1322
2150 DATA52,0,0,14,0,0,3,0,385:1248
2155 REM-----DATI SPRITE 3-----1989
2160 DATA0,0,0,0,0,0,0,0,256:1144
2170 DATA0,0,0,0,0,0,0,0,256:1154
2180 DATA0,0,0,0,0,0,0,0,256:1164
2190 DATA0,0,0,15,128,0,48,128,2316:1549
2200 DATA0,193,0,0,48,128,0,15,1770:1564
2210 DATA128,0,0,0,0,0,0,0,394:1303
2220 DATA0,0,0,0,0,0,0,0,256:1204
2230 DATA0,0,0,0,0,0,0,0,256:1214
3000 SYS49152:END:800
4990 REM----LETTURA E CONTROLLO DATI----2348
5000 FORI=0TO L-1 STEP 8:1364
5010 CT=0:542
5020 FORX=1TO8:839
5030 READLM:POKEAD-1+I+X,LM:1733
5040 PRINT"XXXXXXXXXXXX"TAB(10):A#;"CARICATI AL ";
INT((I+X-1)/(L/100))+1;"%":4012
5050 CT=CT+LM*X:1268
5060 NEXTX:433
5070 READCK:502
5080 IFCK<256THENPRINT"ERRATO NUMERO DI DATI
ALLA LINEA ";NL:END:3836
5090 IF(CT+256)<>CKTHENPRINT"DATI ERRATI
ALLA LINEA ";NL:END:3744
5100 NL=NL+10:1008
5110 NEXTI:468
5120 RETURN:162

```

PC ADV 86 2ª SERIE

SEMPRE 100% IBM PC COMPATIBILE

ANCORA PIÙ
AFFIDABILE

ANCORA PIÙ
CONVENIENTE



Completamente rinnovato nell'estetica, l'Advance - ora nella nuova versione ADV86 - è il risultato di due anni di continui miglioramenti derivati dall'esperienza di migliaia di installazioni.

Non solo. In occasione della 2ª serie l'ADV86 offre caratteristiche e possibilità completamente nuove; fra queste la monoscheda, floppy da 360 e 720 K, hard disk fino a 33MB, possibilità di multiposto di lavoro, pacchetti di elaborazione testi, di tabellone elettronico e di data base.

Inoltre la disponibilità di manuali opzionali e manuali per l'operatore e per l'addestramento scritti in italiano. E infine l'inedito materiale doppio strato (metallo/plastica) con cui è realizzato.

Nessun compatibile ha finora annunciato una vera seconda serie con simili caratteristiche. Questo vuol dire che l'ADV86 ha ancora un bel vantaggio.

ALTRE CARATTERISTICHE

comprese nel prezzo

- schede colore e grafica • interfaccia RS232 e parallela
- espandibile a 640K sulla scheda madre • alimentatore da 130 WATT

non comprese nel prezzo

la gamma di schede e di software Condor disponibile per l'IBM PC

PREZZI DELLE CONFIGURAZIONI BASE

(gli add-on sono sempre inclusi)

- configurazione con 128K, 2 floppy da 360K e monitor 3.600.000 lire
- idem ma con hard disk da 10MB e 1 floppy da 360K 5.950.000 lire
- configurazione biutente con 640K e disco da 10MB 9.250.000 lire

La Condor si riserva la possibilità di modificare caratteristiche e prezzi anche senza preavviso

**FORTUNATI QUEI RIVENDITORI DI PC
CHE VOGLIONO SAPERNE DI PIÙ
E SPEDISCONO QUESTO TAGLIANDO**

NOME _____

SOCIETA' _____

INDIRIZZO _____

CITTA' _____

TEL. _____

MC


DISTRIBUTORE ESCLUSIVO

CONDOR INFORMATICS ITALIA, VIA GRANCINI 8, 20145 MILANO
TEL. 02/43.45.62 - 49.87.549 - 49.87.713, TELEX 326818

CONDOR INFORMATICS CENTRO, VIA ERNESTO BASILE 21,
00128 ROMA, TEL. 06/52.04.158 - 52.06.349 - 52.03.604

CONDOR INFORMATICS SICILIA, VIA LATINA 1, ISOLATO 350
ANGOLO VIALE BOCCETTA, 98100 MESSINA, TEL. 090/41.584

TECNOLOGIA GIAPPONESE DI AVANGUARDIA

CHINON

DISK DRIVE

PREZZI
IVA
ESCLUSA

CASELLA POSTALE 142
56025 PONTEDERA (PI)
VIA MISERICORDIA, 84
TEL. 0587 - 212.312



10 VOLTE PIÙ SILENZIOSI DEGLI ALTRI!!!

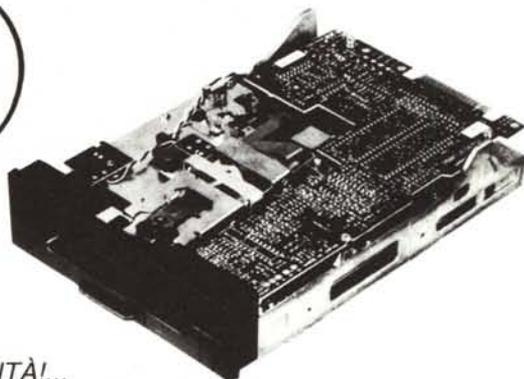
PER APPLE
* 48 TPI
* 96 TPI

PREZZO
NETTO
IMPOSTO
L. 288.000
+ IVA

GARANZIA
QUALITÀ
1 ANNO

PREZZO
NETTO
IMPOSTO
L. 299.000
+ IVA

PER IBM
* 500K
* 1MB



DATAFLEX PROFESSIONAL

DISCHETTI 5 1/4
CON BOX PLASTICA OMAGGIO
SS/DD 30 PZ. L. 2.550
SS/DD 100 PZ. L. 2.350
DS/DD 30 PZ. L. 3.400
DS/DD 100 PZ. L. 3.150

EH SÌ, PRIMA O POI SI ARRIVA ALLA QUALITÀ!...

MOLTI COSTRUISCONO DISK DRIVES MA **NESSUNO** PROVVEDE A REGOLARE ELETTRONICAMENTE, MEDIANTE FOTOCPELLULA, IL MOVIMENTO DELLA TESTINA: È UN BREVETTO DELLA CHINON GRAZIE AL QUALE VENGONO EVITATI RIPETUTI SBATTIMENTI CONTRO IL «FINE CORSA» RENDENDOL'APPARECCHIO SUPER SILENZIOSO E MENO SOGGETTO A STARATURE.

PROVATELI!!!... VI ACCORGERETE DELLA DIFFERENZA !!!...

PC/XT COMPATIBILE

L. 1.750.000

N. 2 DRIVE DS/DD 360K,
CONTROLLER, MAIN BOARD
128K ESP. A256K, ALIM. 130W,
TASTIERA STAFF. K7,
CABINET METALLO APRIBILE



STESSO MA CON 1 DRIVE 360K + HARD DISK 10 MB L. 3.250.000
STESSO MA CON 1 DRIVE 360K + HARD DISK 20 MB L. 4.150.000
STESSO MA CON CABINET MIGLIORE E TASTIERA K5 L. 1.920.000
DRIVE DA 1MB L. 395.000
MONITOR PHILIPS BM 7513 XIBM L. 227.000
MONITOR COLORI CABEL X IBM L. 439.000
CAVO X CABEL IBM L. 19.000
MONITOR PHILIPS TP-200 L. 122.000

INTERFACCE E VARIE PER IBM PC/XT

MONOCROME DISPLAY	285.000	AD-DA (12 BIT, 16 CAN.)	499.000
COLOR GRAPHIC	320.000	I/O PLUS	420.000
COLOR GRAPHIC (4 LAYER)	454.000	GAME I/O	88.000
COLOR GRAPHIC + PRINTER	395.000	RS-232	160.000
PARALL. PRINTER	145.000	RS-232 (2 PORTE) + CAVI	220.000
CAVO X STAMPANTE	48.000	512K RAM (Ø RAM)	166.000
CONTROLLER + CAVO	227.000	KIT 9 CHIPS 4164	18.000
ALIMENTATORE 130W	360.000	MULTIFUNCTION 256K	274.000
SET N. 3 MANUALI (1.000 PAG.)	60.000	MULTIFUNCTION 384K	383.000

Novità

EPROM WRITER PER IBM PC

L. 560.000
CON DISCO E MANUALE



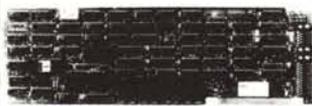
8255 X IBM PC
L. 290.000
CON MANUALE

DRIVE 140K PER APPLE



L. 168.000
TIPO TEAC - TRAZ. DIR.
MITAC FULL SIZE 268.000

MONOCROME / GRAPHIC / PRINTER HERCULES TIPO II



L. 364.000
CON DISCO
E MANUALE

ELITE - E (II E COMPATIBILE)



L. 599.000
SCHEMA SISTEMA
+ 80 COI. + 64 K
L. 170.000

NET WORK

RETI LOCALI

DISPONIBILI SUBITO!

A PREZZI VERAMENTE FAVOLOSI!!!

STAMPANTE A COLORI X IBM PC (CON SET GRAFICO IBM)

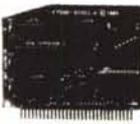


TESTINA A 18 AGHI!!!

132 COL.
180 CPS
NEAR LETTER
QUALITY

L. 1.199.000

80 COLONNE II E



L. 38.000

DATO L'INSTABILE MERCATO DEI CAMBI PREGASI TELEFONARE PER CONFERMA PREZZI E DISPONIBILITÀ
— RICHIEDETECI IL CATALOGO — SCONTI AI SIG. RIVENDITORI —

Prestiti

di Mauro Mancini - Terni

Quando ci si trova nella necessità di ricorrere ad un mutuo per far fronte ad una esigenza particolare, per acquistare una casa, o intraprendere una nuova attività, non sempre le condizioni e i modi che ci vengono proposti concordano con le proprie disponibilità economiche del momento.

Quindi, prima di fare un contratto, sarebbe bene poter analizzare comodamente, e con calma, le condizioni che più si conformano con la somma disponibile mensilmente per il rimborso. Bisogna eseguire calcoli con tassi d'interesse diversi, oppure variando il valore del mutuo per confrontare prestiti tra loro. Tutti questi conti sono certamente noiosi e ripetitivi ed è facile sbagliarsi.

Con il programma PRESTITI è possibile effettuare confronti tra mutui, variando di volta in volta un elemento base.

Un mutuo viene confrontato in PRESTITI sulla base della durata (OPZIONE 1), del valore (OPZIONE 2), e del tasso d'interesse (OPZIONE 3). Con la quarta opzione, si hanno visualizzate e scomposte le rate di un mutuo. Infatti, come certamente si saprà, l'importo della rata (costante) di un qualsiasi mutuo è dato dalla somma del valore degli interessi e il valore che viene effettivamente rimborsato.

L'opzione quattro, appunto, scompone ogni rata in questi due fattori, calcolando anche quanto resta da versare in corrispondenza di ogni rata.

Come si usa

Dopo il lancio di PRESTITI appare l'installazione del programma, che è sostituita dopo alcuni secondi dal menu principale.

Da qui possiamo scegliere tra quattro opzioni, mentre la quinta ci consente di uscire dal programma una volta terminato il lavoro. Procediamo con ordine provando una alla volta le possibilità offerte.

Scelta in base alla durata

Si accede a questa opzione premendo il tasto "1".

È presentata una maschera di spiegazione divisa in due paragrafi, nel primo sono trattati i risultati che si ottengono, e nel

secondo vengono elencati i dati in input necessari al programma. Impostando come dati fissi il valore del mutuo e del tasso d'interesse, sono calcolati gli interessi totali e l'importo della rata di ogni periodo compreso tra la durata minima del mutuo e la durata massima. Un esempio di output che spiega meglio l'utilità del programma è dato dalla figura 1, in cui per un mutuo di 35 milioni, al tasso del 24 per cento, sono elaborati dati per prestiti della durata di 5, 7 e 9 anni, poiché si è impostato un inter-

----- SCELTA IN BASE ALLA DURATA -----
 VALORE DEL MUTUO 35000000
 TASSO D'INTERESSE 24 %

DURATA DEL MUTUO	IMPORTO MENSILE	TOTALE INTERESSI
5 ANNI	1006000	25412000
7 ANNI	863000	37546000
9 ANNI	793000	50695000

Figura 1

Le Routine dell'Applesoft

Dopo aver visto la scorsa volta il funzionamento e la posizione in memoria dei due accumulatori in virgola mobile denominati FAC1 e FAC2, ecco ora una lista di funzioni aritmetiche del Basic che ne fanno uso. Dal Basic è possibile passare un valore ad una routine in linguaggio macchina tramite FAC1; è qui infatti che la funzione `USR()` mette il valore tra parentesi, e sempre dal FAC1 preleva il valore da assegnare alla variabile a sinistra dell'uguale una volta terminata la routine in LM (vedi manuale Applesoft a pagina 45). In FAC1, come si può vedere dalla tabella, viene anche messo sempre l'ultimo risultato di ogni operazione aritmetica o di assegnazione. Con le routine del Basic è possibile realizzare dei programmi in LM che gestiscono anche l'aritmetica in virgola mobile risparmiando tempo di sviluppo e spazio in memoria.

Loc.	Funzione
E10C	AVINT: Rende intero il FAC1
E2F2	GIVAYF: Rende float il numero a 16 bit in A e Y A parte alta, Y parte bassa.
E301	SGNFLT: Rende float il numero a 8 bit in Y.
EAF9	MOVFM: Trasferisce in FAC1 la variabile puntata da (Y,A).
EB2B	MOVFM: Trasferisce in memoria (Y,A) il FAC1.
EB53	MOVFA: Trasferisce FAC2 in FAC1.
EB63	MOVAF: Trasferisce FAC1 in FAC2.
E7AA	FSUBT: Calcola FAC1= FAC2 - FAC1.
E7C1	FADDT: Calcola FAC1= FAC2 + FAC1.
E9B2	FMULT: Calcola FAC1= FAC2 x FAC1.
EA39	MULIO: Calcola FAC1= FAC2 x 10.
EA55	DIVIO: Calcola FAC1= FAC2 / 10.
EA69	FDIVT: Calcola FAC1= FAC2 / FAC1.
E9E3	MOVAM: Trasferisce in FAC2 la variabile puntata da (Y,A).
EB82	SIGN: A = segno di FAC1 (1 +, FF -, 0 0)
EB93	FLOAT: Rende float l'intero con segno in A.
EBAF	ABS: FAC1= ABS(FAC1)
EC23	INT: FAC1= INT(FAC1)
EE8D	SQRT: FAC1= SQR(FAC1)
EED0	NEG: FAC1= +/- FAC1
EF09	EXP: FAC1= e * FAC1.
EFAE	RND: FAC1= RND(1).
EFEA	COS: FAC1= COS(FAC1)
EFF1	SIN: FAC1= SIN(FAC1)
FO3A	TAN: FAC1= TAN(FAC1)
FO9E	ATN: FAC1= ATN(FAC1)
EC4A	FIN: legge un numero float da Basic.
ED2E	PRTFAC: Stampa il contenuto di FAC1.

Esempio

```
0300- 20 4A EC JSR #ECA4 FIN
0303- 20 39 EA JSR #EA39 MULIO
0306- 20 8D EE JSR #EED8 SQRT
0309- 4C 2E ED JMP #ED2E PRTFAC

esegua SQR( N x 10)

10 POKE 1014,0: POKE 1015,3
20 & 25.56

RUN
15.9874951
```

due valori. C'è ora da inserire il valore del mutuo che deve essere espresso in migliaia, avere un minimo di tre cifre e un massimo di sei, questo per non incorrere in una rappresentazione in notazione scientifica, quando il valore del mutuo è stampato completo delle migliaia. Il quarto input è il tasso d'interesse che può assumere valori da 0.1 a 40 per cento. Quinto e ultimo dato è il passo desiderato per i calcoli. L'intervallo di calcolo consentito è logicamente funzione della durata minima e massima, e più precisamente è dato come valore massimo, dalla differenza tra le due grandezze prima citate. Come detto prima l'unica fra-

zione di un anno è 6 mesi, equivalente a 0.5. Attivando la routine d'errore, nel messaggio è indicato il valore massimo consentito per l'intervallo di calcolo, in funzione dei valori inseriti.

Volendo in un qualsiasi istante uscire dalla fase di input, premere CTRL-C RETURN, come indicato nella parte bassa dello schermo. Inserito l'ultimo dato si può decidere di stampare o meno i risultati, in caso affermativo viene abilitata la stampante per lo slot 1. Come si vede dalle figure, ogni videata contiene 10 righe di output, premere un qualsiasi tasto per continuare la visualizzazione. Terminato di

stampare, con il tasto "S" si ritorna da capo all'input di questa opzione per riprovare con altri dati, oppure un tasto per tornare al menu.

Scelta in base al valore del mutuo

Premendo il tasto "2" si accede all'opzione che confronta i mutui in base al loro ammontare. Visualizzazione della maschera di trattazione dei punti salienti dell'opzione, dalla quale si prosegue con il tasto "C".

Inserendo come dati fissi il tasso d'interesse e la durata del mutuo, vengono calcolati gli interessi totali e l'importo della rata,

```

1 REM *****
2 REM
3 REM          PRESTITI
4 REM
5 REM          APPLE IIe 23/03/85
6 REM
7 REM *****
8 REM
9 REM
10 GOTO 170
11 VTAB 24: PRINT M$(27): RETURN
12
13 FOR LL = 1 TO 5: VTAB 21: PRINT
14 M$(F):I$: FOR TT = 1 TO 400:
15 NEXT : VTAB 21: PRINT "
16
17 " : FOR TT = 1 TO 300: NEXT
18 : NEXT : RETURN
19
20 VTAB 23: PRINT M$(8): VTAB 23
21 : HTAB 16: GET D$
22
23 RETURN
24
25 VTAB 23: PRINT M$(11): HTAB 2
26 7: VTAB 23: GET D$
27
28 IF D$ < > "C" THEN 180
29
30 RETURN
31
32 VTAB 10: PRINT M$(21): VTAB
33 10: HTAB 25: GET D$
34
35 ST = 0: IF D$ < > "S" THEN RETURN
36
37
38 PR# 1: PRINT CHR$(27): CHR$(
39 64): CHR$(27): CHR$(77): CHR$(
40 20):ST = 1: RETURN
41
42 PRINT M$(5): PRINT : PRINT :
43 PRINT : PR# 0: RETURN
44
45 VTAB 22: PRINT "VUOI RIPROVA
46 RE": PRINT "CON ALTRI DATI ?
47 (S/N) ": VTAB 23: HTAB 26:
48 GET D$
49
50 Z = 0: RETURN
51
52 GOSUB 1810: TEXT : HOME : VTAB
53 14: PRINT M$(5): VTAB 16: HTAB
54 6: PRINT "PIANIFICAZIONE DI
55 UN PRESTITO": PRINT : PRINT
56 M$(5): FOR Y = 1 TO 2200: NEXT
57
58
59 TEXT : HOME : VTAB 3: HTAB 1
60 0: PRINT "### MENU" OPZIONI
61 ###": VTAB 7: PRINT " 1) SC
62 ELTA IN BASE ALLA DURATA": PRINT
63
64 PRINT " 2) SCELTA IN BASE A
65 L VALORE DEL MUTUO": PRINT
66 PRINT " 3) SCELTA IN BASE A
67 I TASSI D'INTERESSE"
68
69 PRINT " 4) PIANO DI RIBORS
70 O DI UN MUTUO": PRINT
71 PRINT " 5) PER USCIRE DAL P
72 ROGRAMMA"
73
74 VTAB 20: PRINT " Seleziona
75 un'opzione premendo": PRINT
76 : HTAB 11: PRINT "il numero
77 corrispondente": HTAB 36: VTAB
78 22
79
80 GET D$: IF D$ = " " THEN 240
81
82
83 IF D$ = "1" THEN 320
84 IF D$ = "2" THEN 730
85 IF D$ = "3" THEN 1110
86 IF D$ = "4" THEN 1500
87 IF D$ = "5" THEN : HOME : END
88
89
90 GOTO 240
91
92 REM SCELTA IN BASE ALLA DUR
93 ATA
94
95 TEXT : HOME : INVERSE : VTAB
96 2: PRINT M$(1): NORMAL
97 VTAB 5: PRINT M$(9):": PRINT
98 : PRINT "PER OGNI PERIODO CO
99 MPRESO TRA LA DURATA MINORE,
100 E QUELLA MAGGIORE,"
101
102 PRINT "IL CORRISPONDENTE IMP
103 ORTO MENSILE,E IL TOTALE DE
104 GLI INTERESSI."
105
106 VTAB 13: PRINT M$(10):" :": PRINT
107
108 : PRINT "LA DURATA MINORE": TAB(
109 23)"(IN ANNI).":
110
111 PRINT "LA DURATA MAGGIORE
112 (IN ANNI).":
113
114 PRINT "IL VALORE DEL MUTUO
115 (IN MIGLIAIA).":
116
117 PRINT "IL TASSO D'INTERESSE
118 APPLICATO."
119
120 PRINT "OGNI QUANTI ANNI SI D
121 ESIDERA AVERE " : PRINT TAB(
122 24)"IL CONFRONTO.": GOSUB 80
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

per ogni valore del mutuo compreso tra il valore minimo e massimo impostati. La figura 2 mostra un esempio di elaborazione, per un valore del mutuo che va da 5 a 8 milioni, per una durata di 3 anni, e un intervallo di calcolo di 500 mila lire. Il primo dato da inserire è l'importo minimo che si desidera per il mutuo, il quale non deve essere inferiore alle 100 mila e superiore ai 999 milioni. Il secondo dato riguarda l'importo massimo per il prestito, che deve assumere un valore superiore all'importo minimo, e inferiore al miliardo. Proseguendo con l'inserimento dei dati, troviamo ora la durata del mutuo, che può

- SCELTA IN BASE AL VALORE DEL MUTUO -
AMMORTAMENTO IN 3 ANNI
TASSO D'INTERESSE 19 %

VALORE DEL MUTUO	IMPORTO MENSILE	TOTALE INTERESSI
5000000	183000	1598000
5500000	201000	1757000
6000000	219000	1917000
6500000	238000	2077000
7000000	256000	2237000
7500000	274000	2397000
8000000	293000	2556000

Figura 2

assumere periodi di tempo che vanno da 6 mesi ad un anno. Da ricordare ancora una volta che la durata 6 mesi, è l'unico perio-

do valido al di sotto di un anno, ed equivalente a 0.5. Per il tasso d'interesse sono accettate quote da 0.1 a 40 per cento.

Come valore minimo per l'intervallo di calcolo sono valide 100 mila lire, per quanto riguarda il tetto massimo, restano validi i criteri esposti all'opzione precedente, in ogni caso provvederà il programma stesso a suggerire i valori ammessi.

Esauriti gli input, bisogna rispondere con "S" oppure "N" per la richiesta di stampa, dopodiché vengono comunque vi-

```

T1: "%": PRINT
1010 PRINT TAB(4)"VALORE": TAB(
11)M*(6): PRINT TAB(4)"DEL
MUTUO": TAB(17)M*(7): PRINT
: POKE 34,10
1020 FOR I = MI TO MA STEP A
1040 S = VI / ((1 + VI) ^ (N * 12
) - 1) + VI
1050 SS = S * I * B = INT (SS * N *
12 - 1) * 1000: SS = INT (SS
): PRINT TAB(10) - LEN (STR
(I)); I * 1000: TAB(21) - LEN
(STR (SS)); SS * 1000: TAB(
37) - LEN (STR (B)); B: Z =
Z + 1
1060 IF Z = 10 THEN PR# 0: GOSUB
60: HOME : Z = 0: IF ST = 1 THEN
PR# 1
1070 NEXT : GOSUB 140: GOSUB 150
1080 IF D# = "S" THEN GOTD 790
1090 TEXT : GOTD 180
1100 REM SCELTA IN BASE AI TASS
I
1110 TEXT : HOME : INVERSE : VTAB
2: PRINT M*(3): NORMAL
1120 VTAB 5: PRINT M*(9): ": PRINT
: PRINT "PER OGNI TASSO D'IN
TERESSE COMPRESO TRA IL MINO
RE E IL MAGGIORE,": PRINT "I
L CORRISPONDENTE IMPORTO MEN
SILE, E IL TOTALE DEGLI INTE
RESSI."
1130 VTAB 13: PRINT M*(10): ":
PRINT
1140 PRINT "VALORE MINIMO DEL TA
SSO D'INTERESSE."
1150 PRINT "VALORE MASSIMO DEL T
ASSO D'INTERESSE."
1160 PRINT "VALORE DEL MUTUO (
IN MIGLIAIA)."
1170 PRINT "LA DURATA DEL MUTUO"
TAB(27)"(IN ANNI)."
1180 PRINT "OGNI QUANTO DEVONO E
SSERE CALCOLATI GLI INTERE
SSI" TAB(25)"(IN UNITA)":
GOSUB 80
1190 POKE 34,3: HOME : GOSUB 40
1200 VTAB 8: E = 11: INPUT "VAL.M
IN.DEL TASSO D'INTERESSE :":
MI: IF MI = 0 THEN 1200
1210 IF MI > 40 THEN F = 18: GOSUB
50: GOTD 1200
1220 VTAB 10: E = 12: INPUT "VAL.
MAS.DEL TASSO D'INTERESSE :":
MA: IF MA = 0 THEN 1220
1230 IF MA = MI THEN F = 23: GOSUB
50: GOTD 1200
1240 IF MA > 40 THEN F = 18: GOSUB
50: GOTD 1220
1250 IF MA < MI THEN F = 12: GOSUB
50: GOTD 1200
1260 VTAB 12: E = 13: INPUT "VALD
RE DEL MUTUO(IN MIGLIAIA)":
V: IF V = 0 THEN 1260
1270 IF V < 100 THEN F = 25: GOSUB
50: GOTD 1260
1280 IF V > 999999 THEN F = 16: GOSUB
50: GOTD 1260
1290 VTAB 14: E = 14: INPUT "DURA
TA DEL MUTUO (IN ANNI)":
N: IF N = 0 THEN 1290
1300 IF N < 1 AND N > 0.5 THEN
F = 24: GOSUB 50: GOTD 1290
1310 IF N > 50 THEN F = 17: GOSUB
50: GOTD 1290
1320 VTAB 16: PRINT "INTERVALLO
DESIDERATO PER"
1330 E = 15: INPUT " I CALCOLI
(IN UNITA)": : A: IF A =
0.1 THEN A = 0.1001
1340 IF A = 0 THEN 1320
1350 IS = (MA - MI): IF A > IS THEN
F = 19: IS = STR (IS): GOSUB
50: IS = " ": GOTD 1320
1360 VI = (MA / 100) / 12: S = VI /
((1 + VI) ^ (N * 12) - 1) +
VI: SS = S * V: B = INT (SS *
N * 12 - V): IF B < 999999 THEN
1380 F = 20: GOSUB 50: GOTD 1200
1380 HOME : VTAB 8: GOSUB 110: HOME
: VTAB 2: INVERSE : PRINT M*
(3): NORMAL : PRINT
1390 PRINT "DURATA DEL MUTUO ": N
: ANNI": PRINT
1400 PRINT "VALORE DEL MUTUO ": V
* 1000: PRINT
1410 PRINT TAB(4)"CON UN": TAB(
17)M*(6)
1420 PRINT TAB(4)"TASSO DEL": TAB(
17)M*(7): PRINT : POKE 34,10
1430 FOR I = MI TO MA STEP A: VI =
(I / 100) / 12: S = VI / ((1 +
VI) ^ (N * 12) - 1) + VI: SS =
S * V: B = INT (SS * N * 12 -
V) * 1000
1440 SS = INT (SS): PRINT TAB(
7) - LEN (STR (FN F(I)))
: FN F(I): "%": TAB(21) - LEN
(STR (SS)): SS * 1000: TAB(
37) - LEN (STR (B)): B: Z =
Z + 1
1450 IF Z = 10 THEN PR# 0: GOSUB
60: HOME : Z = 0: IF ST = 1 THEN
PR# 1
1460 NEXT : GOSUB 140: GOSUB 150
1470 IF D# = "S" THEN GOTD 1190
1480 TEXT : GOTD 180
1490 REM PIANO DI RIMBORSO
1500 TEXT : HOME : VTAB 2: INVERSE
: PRINT M*(4): NORMAL
1510 VTAB 5: PRINT M*(9): ": PRINT
: PRINT "L'AMMONTARE DELLA R
ATA MENSILE CHE VIENE D
IVISA IN VALORE RIMBORSATO E
QUOTA PER GLI INTERESSI
IL RESTO ANCORA DA VERSARE
PER IL SALDO DEL MUTUO."
1520 VTAB 13: PRINT M*(10): ":
PRINT
1530 PRINT "IL VALORE DEL MUTUO
(IN MIGLIAIA)": PRINT "IL N
UMERO DELLE RATE."
1540 PRINT "IL TASSO D'INTERESSE
APPLICATO.": GOSUB 80
1550 POKE 34,2: HOME : GOSUB 40
1560 VTAB 8: E = 16: INPUT "VALOR
E DEL MUTUO (IN MIGLIAIA)":
V: IF V = 0 THEN 1560
1570 IF V < 100 THEN F = 25: GOSUB
50: GOTD 1560
1580 IF V > 999999 THEN F = 16: GOSUB
50: GOTD 1560
1590 VTAB 10: E = 17: INPUT "NUME
RO DELLE RATE
: : R: IF R = 0 THEN 1590
1600 IF R > 130 THEN F = 22: GOSUB
50: GOTD 1590
1610 VTAB 12: E = 18: INPUT "TASS
O D'INTERESSE APPLICATO
: ": T: IF T = 0 THEN 1610
1620 IF T > 40 THEN F = 18: GOSUB
50: GOTD 1610
1630 HOME : VTAB 8: GOSUB 110: HOME
: VTAB 2: INVERSE : PRINT M*
(4): NORMAL : PRINT
1640 PRINT " MUTUO ": V * 1000:
: "TASSO "T": "%": PRINT
1650 T = T / 100: L = T / 12: F =
.
S
1660 RM = INT ( INT (F + V * (L /
(1 - (1 + L) ^ (- R))) * 10
0) / 100)
1670 PRINT "IMPORTO MENSILE ": R
M * 1000: PRINT
1680 PRINT " M QUOTA INT.
NETTO RESTO": PRINT
: POKE 34,9
1690 FOR I = 1 TO R: VA = INT ( INT
(F + V * L * 100) / 100)
1700 PRINT TAB(5) - LEN (STR
(I)): I: TAB(14) - LEN (STR
(VA)): VA * 1000: TAB(25) -
LEN (STR (RM - VA)): (RM -
VA) * 1000:
1710 V = INT ( INT (F + (V - (RM
- VA)) * 100) / 100)
1720 PRINT TAB(37) - LEN (STR
(V)): V * 1000: Z = Z + 1
1730 IF Z = 10 THEN PR# 0: GOSUB
60: HOME : Z = 0: IF ST = 1 THEN
PR# 1
1740 NEXT : GOSUB 140: GOSUB 150
1750 IF D# = "S" THEN GOTD 1550
1760 TEXT : GOTD 180
1780 IF PEEK (222) = 255 THEN 1
80
1790 IF PEEK (222) = 254 THEN
1800 ON E GOTD 410,440,490,510,5
30,800,830,870,890,910,1200,
1220,1260,1290,1320,1560,159
0,1610
1810 DEF FN F(I) = INT (I * 10
^ 1) / 10 ^ 1
1820 DIM M*(30)
1830 ONERR GOTD 1780
1840 M*(1) = "----- SCELTA IN BAS
E ALLA DURATA -----": M*(2) =
"----- SCELTA IN BASE AL VALORE
DEL MUTUO -----": M*(3) = "----- SCEL
TA IN BASE AI TASSI D'INTERE
SSE -----": M*(4) = "----- PIANO D
I RIMBORSO DI UN MUTUO -----"
1850 M*(5) = "-----": M*(6) =
"IMPORTO TOTALE": M*(7) =
"----- MENSILE INTERESSI": M*(8)
= "Premi un tasto": M*(9) =
"LA SEGUENTE OPZIONE CALCOLA
": M*(10) = "NECESSITA DEI SE
GUENTI DATI"
1860 M*(11) = "Per continuare pre
mi (C)": M*(12) = "Valori dei
tassi invertiti": M*(13) =
"Valori di durata invertiti":
M*(14) = "Valori dei mutui i
nvertiti": M*(15) =
"
1870 M*(16) = "Valore del mutuo e
rrato (MAX 999999)": M*(17) =
"Valore di durata errato (MA
X 50)": M*(18) = "Valore del
tasso errato (MAX 40)": M*(19
) = "L'intervallo consentito
e' MAX ": M*(20) = "Inserisc
i dati piu' piccoli per favo
re"
1880 M*(21) = "Vuoi la stampa ? (
S/N)": M*(22) = "Troppe le ra
te introdotte (MAX 130)": M*(
23) = "Mancanza di intervall
o per i calcoli": M*(24) = "P
er dur.sotto i 12 mesi, solo
6 = 0.5"
1890 M*(25) = "Valore del mutuo e
rrato (MIN 100)": M*(26) = "P
er intervalli minori di 1,so
lo 0.5": M*(27) = "Per uscire
premi CTRL-C RETURN"
1900 RETURN

```

sualizzati i dati elaborati. Se la visualizzazione occupa più di una pagina di schermo, premere un tasto qualsiasi per proseguire. Alla fine dell'elaborazione, come per gli altri casi, è possibile riprovare la stessa opzione, con valori diversi, senza che sia necessario tornare al menu principale.

Scelta in base ai tassi d'interesse

Si accede a questo ultimo confronto digitando il numero "3". Un esempio di elaborazione eseguita da questa opzione è riportato in figura 3, dove per un mutuo di 25 milioni, della durata di 8 anni, si eseguono confronti di tasso dal 14 al 19 per cento, con un passo dell'1 per cento.

Come nelle precedenti scelte, ci viene chiesto per primo il valore minimo, in que-

sto caso del tasso d'interesse. Può essere inserito per tutti i valori superiori a 0 e uguali o minori di 40. Il successivo input, ossia il valore massimo del tasso d'inter-

- SCELTA IN BASE AI TASSI D'INTERESSE -		
DURATA DEL MUTUO 8 ANNI		
VALORE DEL MUTUO 25000000		
CON UN TASSO DEL	IMPORTO MENSILE	TOTALE INTERESSI
14 %	434000	16691000
15 %	448000	18068000
16 %	463000	19469000
17 %	478000	20891000
18 %	493000	22335000
19 %	508000	23801000

Figura 3

se, deve essere superiore al minimo e comunque non superare il tetto del 40%. Si ha poi il valore del mutuo; se si sono lette le spiegazioni delle opzioni precedenti, non dovrebbero esserci problemi per il suo inserimento, così come per la durata del mutuo.

Ultimo input come sempre, prima dell'elaborazione, l'intervallo di calcolo.

Piano di rimborso di un mutuo

Selezionando il tasto "4" si ha la possibilità di utilizzare l'opzione PIANO DI RIMBORSO DI UN MUTUO. Il primo dato è il valore del mutuo, che anche in questo caso deve essere superiore alle 100 mila lire e inferiore al miliardo. Bisogna inserire ora il numero delle rate di cui è

Apple - posta

"&" e compilatori

Come si possono passare dei comandi ad una routine in linguaggio macchina da un programma compilato, visto che nessun compilatore attualmente in commercio accetta l'istruzione '&'?

Giuseppe Falcone
Francavilla al Mare (CH)

Ci sono due strade possibili: la prima consiste nel "POKARE" direttamente in una certa locazione il codice del comando; ad esempio:

POKE 255,1:CALL 768.

La seconda soluzione è quella di scrivere il comando in una stringa che poi verrà letta dal programma in Linguaggio Macchina direttamente dalla area variabili, facendo attenzione al fatto che molti compilatori permettono di spostare l'area variabili in qualsiasi posizione. La riga di chiamata avrà perciò una forma simile: CMD\$="PRINT USING": CALL 768.

Quale stampante

Carissimi di MC, vi ho scritto perché desidererei avere alcuni consigli e chiarimenti.

A — Quali sono i parametri fondamentali per valutare le prestazioni di una stampante... verso quali modelli mi dovrei orientare non superando le 800 mila lire?

B — Le schede 80 colonne con e senza RAM del IIe sono utilizzabili sul II+?

C — Ho un monitor da 18 MHz di banda passante, va bene per le 80 colonne?

D — Che cos'è il Tool Kit HRCG?

Vorrei porvi poi alcuni quesiti di ordine tecnico:

1) cosa avviene nel computer dal momento in cui si preme un tasto al momento in cui il carattere appare sul video?

2) è possibile immettere in un programma Basic una funzione matematica direttamente da tastiera?

3) come funziona il mouse?

Sergio Sartori (BO)

A — La cosa più importante in una stampante è la precisione della meccanica che non deve presentare giochi tra i vari componenti, poi interessa la qualità di stampa e gli accessori: frizione, inseritore di foglio singolo, trattori a passo variabile e anche le varie interfacce: seriale, parallela, con buffer e senza. Altrettanto importante è il fatto che la stampante sia compatibile con modelli molto noti, questo perché sia facilmente gestibile dal software in commercio.

B — No, per il vecchio Apple ci sono le apposite schede da inserire nello slot 3.

C — Sì, 18 MHz sono più che sufficienti.

D — HRCG sta per High Resolution Characters Generator: un insieme di programmi per la generazione e la gestione di caratteri programmabili in alta risoluzione.

1) Non succede poi molto: il computer si limita a leggere il contenuto della locazione \$C000 che corrisponde al tasto premuto, se non è un carattere di controllo (Return, ESC, <— o —>) lo deposita nel buffer di riga che inizia alla locazione \$200 e lo invia alla routine \$FDED del monitor che lo scrive sullo schermo.

2) Con la routine pubblicata sul numero 39 si.

3) Tramite due trasduttori ottici trasforma il movimento della sfera, che si trova sotto al Mouse, in una serie di impulsi che incrementano o decrementano due contatori, il valore dei contatori corrisponde alle coordinate attuali del Mouse.

EPROM e Apple Writer

Ho acquistato la Vs EPROM per le miniscopie su Apple II+. Ho seguito le istruzioni presentate sugli articoli comparsi su MC microcomputer ed il tutto ha funzionato egregiamente, tranne purtroppo i comandi per l'editor con l'APPLE WRITER, tipo: !NP, !LM, !RM, ecc.

È possibile avviare a ciò?

Franco Scavizzi - Ponte S. Giovanni

Dopo la modifica occorre scrivere i comandi in minuscolo; quindi !np, !lm, !rm, ecc.

"&" linker

Spett. MC Posta, sono un vostro lettore fedele e possiedo da circa un anno un Apple IIe. Penso di fare una cosa gradita agli applisti che hanno rilocato alla \$300 la routine di autonumerazione (Mc n. 14) poiché alla \$953A avevano la routine di EDIT del numero di aprile di MC, inviandovi questo programmino.

Date CALL-151 e battete:

```
9000: A9 4C A2 10 A0 90 8D F5
9008: 03 8E F6 03 8C F7 03 60
9010: AA F0 FC 20 B1 00 E0 41
9018: F0 07 C6 B8 F0 F2 4C 00
9020: 94 20 67 DD 20 52 E7 A6
9028: 50 A4 51 8E B8 03 8C B9
9030: 03 20 BE DE 20 67 DD 20
9038: 52 E7 A6 50 A4 51 8E BA
9040: 03 8C BB 03 4C 00 03 00
```

E salvare con BSAVE AUTO.&, A\$9000,LS47

E ora le spiegazioni: questa routine è un miglioramento della routine comparsa su MC numero 32 (Bo Arnkhit non si offenda!) e serve per agganciare all'autonumerazione il comando &. Per attivare la routine basta dare, (con l'autonumerazione alla \$300) BRUN AUTO.& e battere:

&A[linea iniziale], [incremento]

es. &A600,5 numerare le linee dalla 600 di 5 in 5.

Detto questo vi saluto porgendo i miei complimenti per l'ottima rivista.

Francesco Meschia - Asti

Non credo che Bo si offenda, visto oltretutto che la routine di Autonumerazione è mia (e per questo dovrei offendermi) non mi offendo invece per il fatto che qualcuno migliori una mia routine, anzi questo mi fa piacere perché significa che a qualcuno serve, e serve tanto da doverla anche adattare alle proprie esigenze; offendersi per queste cose sarebbe un po' come se un costruttore si offendesse perché l'inquilino si arreda la casa!

Lista variabili

VS = "SCELTA IN BASE ALLA DURATA --"
 US = "SCELTA IN BASE AL VALORE DEL MUTUO --"
 XS = "SCELTA IN BASE AI TASSI D'INTERESSE --"
 PS = "PIANO DI RIMBORSO DI UN MUTUO ----"
 GS = "-----"
 IS = "IMPORTO TOTALE"
 MS = "MENSILE INTERESSI"
 TS = "Premi un tasto"
 DMS = "LA SEGUENTE OPZIONE CALCOLA"
 NS = "NECESSITA DEI SEGUENTI DATI"
 CS = "Per continuare premi (C)"
 ST = Variabile che assumendo il valore 1
 abilita la stampante
 V = Valore del mutuo
 T = Tasso d'interesse
 B = Totale degli interessi
 MI = Valore minimo del confronto
 MA = Valore massimo del confronto
 SS = Importo mensile
 A = Valore dell'intervallo di calcolo
 N = Durata del mutuo
 I = Contiene il passo per i calcoli dal
 valore minimo al valore massimo
 R = Numero delle rate
 RM = Importo della rata mensile

composto il prestito. Il numero massimo digitabile è 130 pari ad un periodo superiore ad un anno. Ultimo input è il tasso d'interesse, il cui range è identico alle con-

----- PIANO DI RIMBORSO DI UN MUTUO -----

M	QUOTA INT.	NETTO	RESTO
1	50000	603000	4397000
2	43000	610000	3787000
3	37000	616000	3171000
4	31000	622000	2549000
5	25000	628000	1921000
6	19000	634000	1287000
7	12000	641000	646000
8	6000	647000	-1000

Figura 4

Le mie routine, in genere, sono scritte proprio ridotte all'osso per permettere agli utenti di modificarle e aggiustarle come più gli aggrada; solo che se dovessimo pubblicare tutte le modifiche che ci mandano non basterebbe una rivista dedicata unicamente all'Apple.

Visto poi che siamo in tema di modifiche (e per dimostrare che nessun programma può dirsi mai perfetto) ho apportato una piccola modifica alla sua modifica. Scherzi a parte ne ho approfittato per far vedere come si può agganciare una nostra routine al comando & anche se precedentemente ve ne era agganciata un'altra, naturalmente senza perdere l'altro comando. Il trucco consiste nel non modificare brutalmente il puntatore della &, ma di salvare prima il

vecchio valore in modo che se il comando non viene riconosciuto valido per la nuova routine, il controllo viene passato alla routine che era attiva prima della nostra, e così via in cascata fino all'ultima (anzi alla prima) routine che, non avendo "antecedenti" ritornerà al Basic.

In questo modo è possibile lanciare in esecuzione più routine facenti capo alla & a patto che i suffissi siano differenti, ad esempio se una routine usa il comando &A nessun'altra lo può usare, in caso contrario l'ultima routine caricata ha il sopravvento sulle altre.

Una Eprom di Utility?

Ci sono per l'Apple molte comode routine di Utility che farebbe piacere avere sempre residenti, magari su una Rom. Si può realizzare qualcosa del genere, possibilmente spendendo anche poco?

(vari Lettori)

Il modo migliore e, in fondo, non troppo dispendioso di creare una biblioteca di Utility sempre residenti consiste nell'acquisto di un programmatore di Eprom che, oltre ovviamente a permettere la scrittura delle stesse, ne consente anche la lettura, e quindi l'esecuzione. Un programmatore per Apple costa ormai intorno alle 100.000 e consente di scrivere le routine desiderate su una Eprom che poi verrebbe lasciata nel programmatore e usata come una specie di disco. Unico problema consiste nel fatto che le routine devono essere scritte per girare nella zona di memoria che va da C800 a CFFF; se così non fosse occorrerà rilocarle. Naturalmente si potranno avere anche più Eprom ciascuna con le applicazioni preferite: Grafica, Editing, Calcolo e Utility.

I colori dell'Apple

Spett. le Redazione MC, da qualche tempo ho acquistato un sistema Apple compatibile.

Il motivo di questa lettera è la cattiva gestione dei colori in alta risoluzione.

siderazioni precedenti. Anche per questa opzione sono validi i controlli sui valori digitati, quindi niente paura di sbagliare.

Una volta premuto RETURN sull'ultimo input, e terminata l'elaborazione, vengono visualizzati il valore del mutuo, il tasso d'interesse e l'importo della rata.

Ogni singola rata è suddivisa in quota interessi e quota rimborso, e calcolato di fianco ad ognuna il resto da versare. Se la visualizzazione occupa più di una pagina video premere come al solito un tasto per proseguire.

In figura 4 è portato come esempio un prestito di 5 milioni, al tasso del 12 per cento, per 8 rate.

In alta risoluzione i colori disponibili sono 7, detti colori vengono rispettati solo nelle linee orizzontali, così tracciando ad esempio gli assi per il centro del video si verifica che i colori sull'asse orizzontale corrispondono mentre per l'asse verticale sono corrispondenti solo per i colori 1 e 5.

Vorrei conoscere l'eventuale motivo di ciò e, se è possibile cambiando qualche integrato rimediare a ciò.

Maurizio Mellone
Bassano del Grappa (VI)

Il difetto è dovuto al modo di gestire il colore sull'Apple II. Per risparmiare memoria il colore di due punti adiacenti non può essere scelto a piacere tra i sette disponibili. Per decidere di che colore deve essere un punto l'Apple guarda al suo vicino, se questo è acceso, di qualsiasi colore, tutti e due i punti diventano bianchi, se invece il punto è isolato il suo colore dipenderà dalla colonna su cui si trova. Le colonne dispari generano punti di colore verde o arancio, quelle pari di colore violetto o blu. La scelta del primo o del secondo colore della colonna, ovvero tra verde e arancio oppure tra blu e viola, dipende dal valore del settimo bit del byte cui il punto appartiene.

In pratica esistono le seguenti limitazioni: tutti i punti contigui (in orizzontale) sono bianchi, i punti di uno stesso byte saranno o verdi e blu o arancio e viola.

A questo punto si può ottenere una figura in bianco e nero su un monitor a colori in soli due modi: o si plottano tutte le righe verticali doppie (due punti sono sempre bianchi) oppure si disabilita il colore modificando hardware la scheda colore. Chi se la sente può effettuare la modifica (avendo lo schema elettrico) interrompendo la pista che porta alla scheda il segnale TEXT (presente sul BUS) e fissando alto questo ingresso con una resistenza da 1 Kohm verso i +5 volt oppure (meglio) collegando la pista TEXT della scheda colore con una delle uscite Annunciators dell'Apple che si possono settare da software.

MC

0300-	AD F6 03	LDA	#03F6
0303-	BD 40 03	STA	#0340
0306-	AD F7 03	LDA	#03F7
0309-	BD 41 03	STA	#0341
030C-	A9 17	LDA	#17
030E-	BD F6 03	STA	#03F6
0311-	A9 03	LDA	#03
0313-	BD F7 03	STA	#03F7
0316-	60	RTS	
0317-	AA	TAX	
0318-	F0 25	BEQ	#033F
031A-	A5 B8	LDA	#B8
031C-	48	PHA	
031D-	A5 B9	LDA	#B9
031F-	48	PHA	
0320-	E0 45	CPX	#45 "E"
0322-	D0 15	BNE	#0339
0324-	20 B1 00	JSR	#00B1
0327-	C9 44	CMP	#44 "D"
0329-	D0 0E	BNE	#0339
032B-	20 B1 00	JSR	#00B1
032E-	C9 49	CMP	#49 "I"
0330-	D0 07	BNE	#0339
0332-	20 B1 00	JSR	#00B1
0335-	C9 54	CMP	#54 "T"
0337-	F0 09	BEQ	#0342
0339-	68	PLA	
033A-	B5 B9	STA	#B9
033C-	68	PLA	
033D-	B5 B8	STA	#B8
033F-	4C 59 FF	JMP	#FF59
0342-	68	PLA	
0343-	68	PLA	
0344-	4C 00 94	JMP	#9400

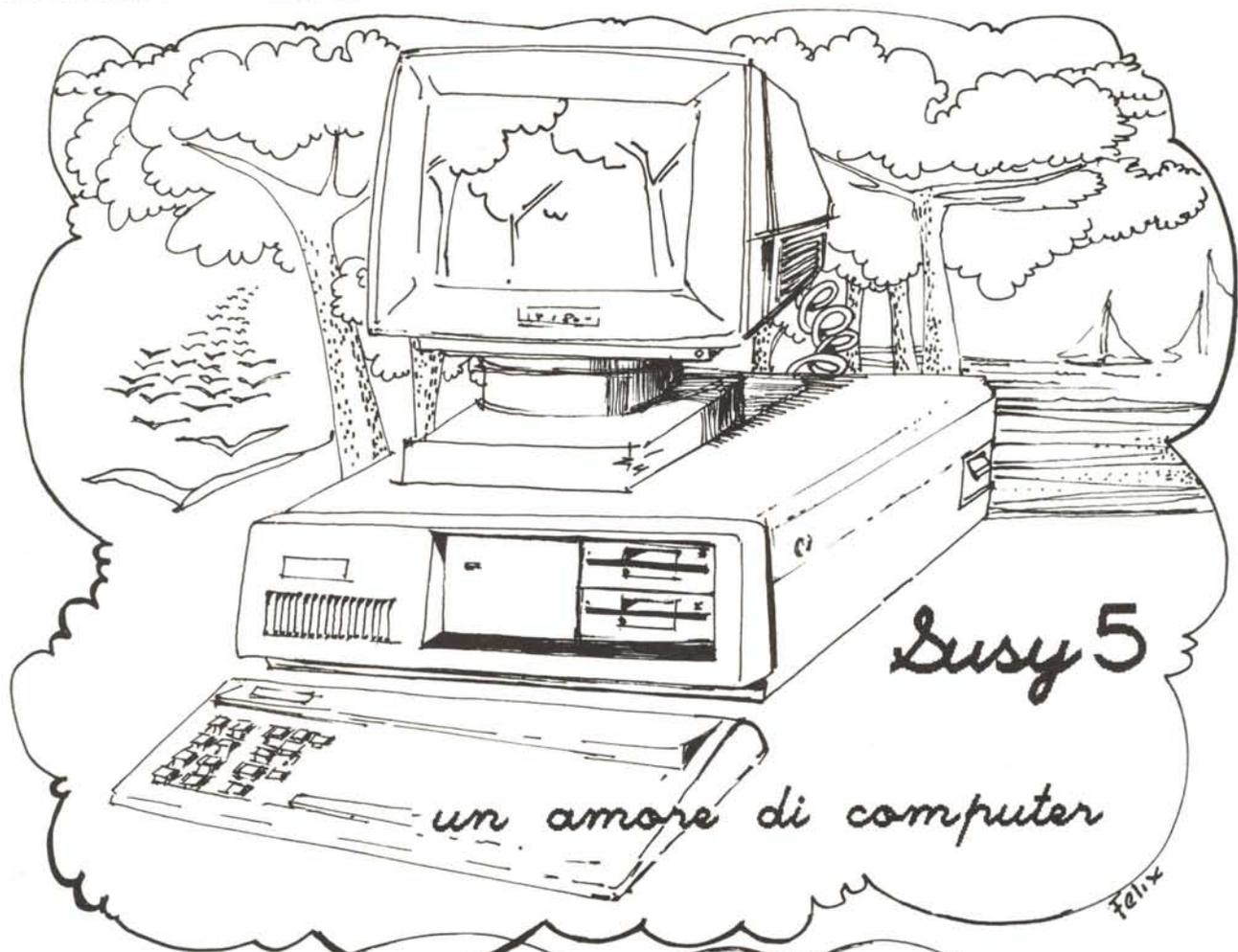
Disassemblato della routine che aggancia un programma al comando &, lasciando attivi i vecchi comandi. I numeri in neretto vanno sostituiti se si riloca la routine in una pagina diversa dalla 3. Il JMP alla riga 344 deve puntare alla nuova routine.

già IL BITTEGONE di felice pagnani

computerline^{srl}

via ubaldo comandini 49

00173 roma - t. 6133025 7970559 tx.621166 fepag i



*Ama il duro lavoro da
sola e collegata in rete
con altre compagne,
colloquia con l'host come
con un vecchio amico.
Soprattutto e' fedele:
Non ti pianta mai in asso!*

software SPECTRUM

Per la rubrica di questo mese abbiamo tre interessanti programmi. Il primo genera alcune curve di Peano (se non sapete cosa sono le curve di Peano recitate subito un contrito mea culpa e correte a leggermi l'articolo intitolato "Passeggiando nel piano" pubblicato sul numero 42); il secondo invece è un simpatico gioco sul tema delle corse dei cavalli, che dimostra tra l'altro come un pizzico di linguaggio macchina (appena 30 byte in questo caso) possa migliorare notevolmente la qualità di un programma. Il terzo, infine, non è un programma vero e

proprio, ma una serie di utili aggiunte al listato "Prospettiva", pubblicato sul numero di febbraio.

■ Peano

di Luca Olivetti - Scandiano (RE)

Incuriosito da un articolo della rubrica MC giochi ho voluto provare a scrivere un programma che generasse alcune curve di Peano. Da qualche parte avevo già visto una definizione in Logo per tracciare la curva di Hilbert.

Non avendo a disposizione il Logo per lo Spectrum ho allora provato a tradurla in Basic. Purtroppo la definizione della curva è ricorsiva; in realtà in Basic ad una subroutine è permesso richiamare se stessa, ma vi è il problema delle variabili e del passaggio dei parametri.

Nel caso si usino molte variabili si può simulare opportunamente uno stack dove porre le variabili all'ingresso della subroutine e da cui recuperarle all'uscita. Ovviamente questa non è una soluzione molto elegante ed inoltre raramente il gioco vale

Listato 1

```

10 REM *****
11 REM *
12 REM *   CURVE DI PEANO   *
13 REM *
14 REM *****
15 REM
20 REM   *** MENU ***

30 BORDER 7: PAPER 7: INK 0: B
RIGHT 0: FLASH 0: CLS
40 PRINT "----- Curve di Pe
ano -----"
50 PRINT "1) curva
di Hilbert"
50 PRINT "2) curva
di Mandelbrot"
50 PRINT "3) curv
a di Koch"
50 PRINT "4) fine"
50 PRINT "Scegli...."
60 PAUSE 0: LET a$=INKEY$: IF
a$<"1" OR a$>"4" THEN GO TO 60
70 IF a$="4" THEN CLEAR: GO T
O 9999
80 CLS: RESTORE 140: FOR i=1
TO VAL a$: READ b$: NEXT i: PRIN
T b$
90 IF a$="1" THEN LET a=90: LE
T direzione=1: LET x=250: LET y=
1: GO SUB 180: GO SUB 220: GO TO
130
100 GO SUB 150: LET d=0: GO SUB
750
110 IF a$="2" THEN GO SUB 380:
GO TO 130
120 GO SUB 600
130 PRINT #0: "Premi un tasto p
er continuare": PAUSE 1: PAUSE
0: GO TO 30
140 DATA "----- curva di Hilb
ert"
140 DATA "----- curva di Man
delbrot"
140 DATA "----- curva d
i Koch"
150 INPUT "x iniziale ";x
160 INPUT "y iniziale ";y
170 INPUT "direzione iniziale "
```

```

; a
180 INPUT "dimensione ";dimensi
one: IF dimensione<1 THEN GO TO
180
190 INPUT "ordine ";ordine: IF
ordine<1 THEN GO TO 190
200 CLS: PRINT #0;AT 0,0;b$: R
ETURN

210 REM   ++++++
+ HILBERT +
+++++

220 IF ordine=0 THEN RETURN:
REM IF ordine=0 [STOP]
230 LET ordine=ordine-1: REM
si abbassa il livello per
le chiamate successive
240 LET d=direzione*90:
GO SUB 750:
REM LEFT direzione+90
250 LET direzione=-direzione:
GO SUB 220:
LET direzione=-direzione:
REM HILBERT dimensione
ordine-1
direzione
260 GO SUB 810:
REM FORWARD dimensione
270 LET d=direzione*90:
GO SUB 780:
REM RIGHT direzione+90
280 GO SUB 220:
REM HILBERT dimensione
ordine-1
direzione
290 GO SUB 810:
REM FORWARD dimensione
300 GO SUB 220:
REM HILBERT dimensione
ordine-1
direzione
310 LET d=direzione*90:
GO SUB 780:
(continua a pagina 146)
```

la candela. Nei casi molto semplici, come quello presentato dalla definizione della curva di Hilbert si può però tentare di simulare la ricorsività senza fare uso di uno stack, sostituendo i valori alle variabili prima di chiamare la subroutine e ripristinandoli all'uscita; il listato dovrebbe contribuire a chiarire le idee.

Così ho scritto in Basic la routine Hilbert; nel listato ho inserito tutte le Rem necessarie per evidenziare la corrispondenza tra le istruzioni Basic e le istruzioni Logo.

Successivamente ho scritto in Logo la definizione per le curve di Mandelbrot e Koch, peraltro molto semplici, e le ho tradotte come la precedente nel Basic dello

Spectrum. Ho poi "abbellito" il tutto con un semplice menu per la scelta di una delle tre curve e per l'immissione dei dati di partenza, ottenendo questo programma.

Consiglio di copiarlo senza le varie Rem che, come ho già detto, servono solo per evidenziare il passaggio Logo-Basic; ov-

(segue da pagina 145)

```

REM RIGHT :direziohe*90
320 GO SUB 810:
REM FORWARD :dimensione
330 LET direzione=-direziohe:
GO SUB 220:
LET direzione=-direziohe:
REM HILBERT :dimensione
:ordine-1
:direziohe
340 LET d=direziohe*90:
GO SUB 750:
REM LEFT :direziohe*90
350 LET ordine=ordine+1: REM
:si ripristina l'ordine
360 RETURN :
REM END

370 REM *****
* MANDELBROT
*****

380 LET d=90: FOR i=1 TO 4:
GO SUB 410:
GO SUB 750:
NEXT i:
REM REPEAT 4 [MAND1 :dimensione
:ordine
LEFT 90]
390 RETURN :
REM END
400 REM *** MAND1 ***
410 IF ordine=1 THEN
GO SUB 810: RETURN :
REM IF :ordine=1
[FORWARD :dimensione
STOP ]
420 LET ordine=ordine-1: REM
:si abbassa l'ordine per
:le chiamate successive
430 GO SUB 410:
REM MAND1 :dimensione :ordine-1
440 GO SUB 750:
REM LEFT 90
450 GO SUB 410:
REM MAND1 :dimensione :ordine-1
460 GO SUB 780:
REM RIGHT 90
470 GO SUB 410:
REM MAND1 :dimensione :ordine-1
480 GO SUB 780:
REM RIGHT 90
490 GO SUB 410:
REM MAND1 :dimensione :ordine-1
500 GO SUB 410:
REM MAND1 :dimensione :ordine-1
510 GO SUB 750:
REM LEFT 90
520 GO SUB 410:
REM MAND1 :dimensione :ordine-1
530 GO SUB 750:
REM LEFT 90
540 GO SUB 410:
REM MAND1 :dimensione :ordine-1
550 GO SUB 780:
REM RIGHT 90
560 GO SUB 410:
REM MAND1 :dimensione :ordine-1
570 LET ordine=ordine+1: REM
:si ripristina ordine

```

```

580 RETURN :
REM END

590 REM *****
* KOCH
*****

600 FOR i=1 TO 3:
GO SUB 630:
LET d=120: GO SUB 750:
NEXT i:
REM REPEAT 3 [KOCH1 :dimensione
:ordine
LEFT 120]
610 RETURN :
REM END

620 REM *** KOCH1 ***

630 IF ordine=1 THEN
GO SUB 810: RETURN :
REM IF :ordine=1 [FORWARD
:dimensione
STOP ]
640 LET ordine=ordine-1: REM
:si abbassa l'ordine per le
chiamate successive
650 GO SUB 630:
REM KOCH1 :dimensione :ordine-1
660 LET d=60: GO SUB 780:
REM RIGHT 60
670 GO SUB 630:
REM KOCH1 :dimensione :ordine-1
680 LET d=120: GO SUB 750:
REM LEFT 120
690 GO SUB 630:
REM KOCH1 :dimensione :ordine-1
700 LET d=60: GO SUB 780:
REM RIGHT 60
710 GO SUB 630:
REM KOCH1 :dimensione :ordine-1
720 LET ordine=ordine+1: REM
:si ripristina l'ordine
730 RETURN :
REM END

740 REM *** LEFT ***

750 LET a=a+d: LET a=a-INT (a/3
60)*360
760 LET h=COS (a*PI/180): LET v
=SIN (a*PI/180): RETURN

770 REM *** RIGHT ***

780 LET a=a-d: LET a=a-INT (a/3
60)*360
790 LET h=COS (a*PI/180): LET v
=SIN (a*PI/180): RETURN

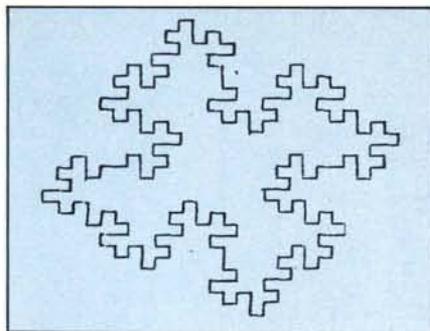
800 REM *** FORWARD ***

810 IF x<0 OR x>255 OR y<0 OR y
>175 THEN LET f=1: GO TO 830
820 LET f=0: PLOT x,y
830 LET x=x+dimensione*h: LET y
=y+dimensione*v: IF x<0 OR x>255
OR y<0 OR y>175 OR f THEN RETUR
N
840 DRAW dimensione*h,dimensione
*v: RETURN

```

vamente non è poi necessario rispettare la particolare indentazione del listato, utilizzata per migliorare la leggibilità delle routine. Il programma chiede, per ogni curva, dimensione ed ordine, cioè la lunghezza in pixel di ogni trattino e l'ordine della curva. Inoltre per le curve di Mandelbrot e Koch chiede le coordinate del punto da cui si vuole far partire il tracciamento della curva e la relativa direzione, espressa in gradi.

La dimensione della curva aumenta notevolmente col crescere dell'ordine, quindi la scelta delle coordinate iniziali è determi-



◀ Curva di Hildebrot del 3° ordine.

nante per far apparire sullo schermo la maggior parte della curva. Quando lo Spectrum sembra paralizzato non preoccupatevi: sta semplicemente disegnando fuori dallo schermo.

Horses

di Andrea Tagliapietra - Milano

Questo programma, che gira sullo Spectrum 48K, si propone di simulare una giornata alle corse ippiche; si tratta di un giochino senza grandi pretese, ma piuttosto piacevole e divertente, specie se giocate in compagnia.

Vediamo subito qual è la procedura per

ottenere una copia su nastro del programma.

Per prima cosa bisogna digitare il programma principale (listato 2) e salvarlo su nastro con un SAVE "horses" LINE 1. Poi si deve dare il NEW, caricare il programma del listato 3 e dare il RUN. Quest'ultimo programma genera una schermata che va salvata come SCREEN\$ subito dopo il programma principale; il SAVE è automa-

tico grazie alla linea 110, quindi basta far partire il registratore non appena compare sullo schermo la solita richiesta di "Start tape then ...".

A questo punto bisogna inserire in memoria, con un caricatore decimale, i 30 byte della parte in linguaggio macchina a partire dall'indirizzo 35000 e salvarli con SAVE "l/m" CODE 35000, 30. L'ultima operazione necessaria è il caricamento dei

Listato 2

```

1 REM *****
2 REM *
3 REM *          HORSES
4 REM *
5 REM *****
6 REM
7 CLEAR 34999: LOAD ""CODE 40
000: LOAD ""CODE 35000: LOAD ""C
ODE
10 DIM v(6): DIM c$(6,10): DIM
x(6): DIM p(6,3): DIM q(6,2): F
OR n=1 TO 6: LET p(n,1)=1: NEXT
n
19 REM **NOMI CAVALLI**
20 BORDER 0: PAPER 0: CLS : IN
K 7
30 PRINT #0; AT 1,0; "Vuoi inser
ire i nomi dei cavalli?"
: PAUSE 0
40 IF INKEY$ <> "s" THEN GO TO 8
0
45 FOR n=1 TO 6
50 INPUT "CAVALLO N°"; (n); " : "
: LINE c$(n): PRINT AT n*2,1; "CA
VALLO N°"; n; " : "; c$(n)
60 PRINT #0; AT 1,0; "E' corrett
o?": PAUSE 0: IF INKEY$="n" THEN
GO TO 50
70 NEXT n: GO TO 90
80 RESTORE 330: FOR n=1 TO 6:
READ c$(n): NEXT n
89 REM **NOMI GIOCATORI**
90 BORDER 4: BRIGHT 1: PAPER 4
: CLS : INK 1
100 INPUT "N° GIOCATORI (1/6)? "
; ng: IF ng < 1 OR ng > 6 THEN GO TO
100
110 DIM z(ng): DIM g$(ng,10): D
IM s(ng): DIM u(ng,4)
115 FOR n=1 TO ng
120 INPUT "GIOCATORE N°"; (n); "
: LINE g$(n): PRINT AT n*2,1; "
GIOCATORE N°"; n; " : "; g$(n)
130 PRINT #0; AT 1,0; "E' corrett
o?": PAUSE 0: IF INKEY$="n" THE
N GO TO 120
140 LET s(n)=500000: NEXT n

```

```

149 REM **CICLO PRINCIPALE**
150 FOR g=1 TO 8: FOR n=1 TO 6:
LET p(n,3)=0: LET x(n)=(RND*1.5
)+.5: NEXT n: GO SUB 350
160 FOR f=1 TO ng: FOR p=1 TO 6
: LET q(p,1)=1.5: NEXT p: GO SUB
500: GO SUB 400
165 IF z(f)=1 THEN NEXT f: GO T
O 250
170 PRINT #0; "PUNTATA di "; g$(f
): PAUSE 0
180 INPUT "CAVALLO N°"; u(f,1):
IF u(f,1) < 1 OR u(f,1) > 6 THEN GO
TO 180
190 INPUT "1 VINCENTE 2 PIAZZAT
0 : "; u(f,2): IF u(f,2) < 1 OR u(f,
2) > 2 THEN GO TO 190
200 INPUT "SOMMA "; u(f,3): IF
u(f,3) > s(f) THEN PRINT #0; AT 1,0
; "TI MANCA LA FRESCA !!!!!": BE
EP 1.2, -40: GO TO 200
210 LET s(f)=s(f)-u(f,3)
220 IF u(f,2)=1 THEN LET u(f,4)
=q(u(f,1),1)
230 IF u(f,2)=2 THEN LET u(f,4)
=q(u(f,1),2)
240 LET p(u(f,1),3)=p(u(f,1),3)
+1
250 NEXT f
260 GO SUB 400: PRINT #0; "PREMI
UN TASTO PER CONTINUARE ": PAUSE
E 0
270 GO SUB 600: GO SUB 700: GO
SUB 800
280 FOR n=1 TO ng: IF s(n) <> 0 T
HEN LET som=1
285 IF s(n)=0 THEN LET z(n)=1
290 NEXT n: IF som=0 THEN CLS :
PRINT AT 10,4; "VI HO SBANCATI !
!": FOR n=-40 TO 40: BEEP .01,n
: NEXT n: GO TO 310
300 LET som=0: LET p(v(1),1)=p(
v(1),1)+1: LET p(v(2),2)=p(v(2),
2)+.5: NEXT g
305 CLS : PRINT AT 10,4; "LE COR
SE SONO FINITE "
310 PRINT AT 12,4; "VOLETE RICO
INCIARE (s/n)?": PAUSE 0: IF INK

```

(continua a pagina 148)

```

EY$="n" THEN STOP
320 RUN 10
330 DATA "MAESTOSO", "CONTESSA",
"NRARICE", "FULMINE", "REGINA", "STA
LLONE"
340 REM ++SUB TERRENO++
350 LET v=INT (RAND*3): IF v=0 T
HEN LET t$="PESANTE": LET q(1,1)
=q(1,1)-1: LET q(2,1)=q(2,1)-1
355 IF v=1 THEN LET t$="BUONO":
LET q(3,1)=q(3,1)-1: LET q(4,1)
=q(4,1)-1
360 IF v=2 THEN LET t$="OTTIMO"
: LET q(5,1)=q(5,1)-1: LET q(6,1)
=q(6,1)-1
365 RETURN
399 REM ++SUB TOTALIZZATORE++
400 BRIGHT 0: BORDER 1: BRIGHT
1: PAPER 1: CLS: INK 7
405 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175
410 PRINT AT 4,1: "CAVALLI";TAB
21: "V";TAB 26: "P";TAB 30: "A"
415 PRINT AT 13,1: "GIOCATORE";T
AB 17: "CAPIT.":TAB 26: "PUNT."
420 PLOT 0,152: DRAW 255,0: PLO
T 0,151: DRAW 255,0: PLOT 232,80
: DRAW 0,70: PLOT 152,80: DRAW 0
,94: PLOT 192,0: DRAW 0,175: PLO
T 128,0: DRAW 0,80
425 FOR n=128 TO 80 STEP -8: PL
OT 0,n: DRAW 255,0: NEXT n: PLOT
0,79: DRAW 255,0
430 FOR n=56 TO 8 STEP -8: PLOT
0,n: DRAW 255,0: NEXT n
435 OVER 1: PRINT AT 1,1: "CORSA
N":TAB 10: INVERSE 1;9: INVERS
E 0:TAB 20: "***":TAB 25: t$
440 FOR n=1 TO 6: PRINT INK 6;A
T 5+n,1;n: "":c$(n):AT 5+n,20;q(
n,1):AT 5+n,25;q(n,2): FOR j=1 T
O 6: IF v(j)=n THEN PRINT AT 5+n
,30;j
441 NEXT j: BEEP .01,10: NEXT n
445 FOR n=1 TO ng: PRINT AT 14+
n,1: INK 6;g$(n):TAB 17:s(n):TAB
25:u(n,3): NEXT n: OVER 0
450 RETURN
499 REM ++SUB QUOTE ++
500 FOR n=1 TO 6: LET q(n,1)=q(
n,1)/((p(n,1)+p(n,2))/1.7+(p(n,
3)/2))/2)
505 LET q(n,1)=q(n,1)/x(n): LET
a$=STR$ q(n,1): LET a$=a$(TO 3
): LET q(n,1)=VAL a$
510 LET q(n,2)=q(n,1)/2: NEXT n
520 RETURN
599 REM ++SUB CARA ++
600 CLS: BORDER 0: PAPER 4: IN
K 0: RANDOMIZE USR 35023
604 PRINT AT 20,0: INK 7: "
605 FOR n=1 TO 6: PRINT AT 6+2*
n,0;n:AT 6+2*n,1: " ";AT 7+2*n,
1: " ": NEXT n
606 PRINT #0;AT 1,0: "ATTENDI PE
R FAVORE"
610 POKE 23624,7: DIM c(6,42):
LET arr=50: LET i=1
615 FOR n=2 TO 41: LET arr=arr-
1
620 FOR s=1 TO 6
625 LET k=INT (RAND*20)+1
635 IF k<20-(q(s,1)*(RAND*2.5)+(
x(s))) AND c(s,42)=0 THEN LET c(
s,n)=c(s,n-1)+1: GO TO 642
640 LET c(s,n)=c(s,n-1)
642 IF c(s,n)+2)=arr AND c(s,42

```

```

=1: LET i=i+1: GO TO 645
645 NEXT s
650 IF i>6 THEN LET b=n: LET n=
42
655 NEXT n
656 FOR n=1 TO 6: PRINT AT 6+2*
n,0;" ": NEXT n: PRINT #0;AT 1,0
;"
660 GO SUB 900: LET arr=50
665 FOR i=2 TO b: LET arr=arr-1
670 FOR n=1 TO 6: PRINT AT 6+n*
2,c(n,i); " "; PRINT AT 7+n*2,
c(n,i); " "; NEXT n: RANDOMIZE
USR 35000
675 FOR n=1 TO 6: PRINT AT 6+n*
2,c(n,i); " "; PRINT AT 7+n*2,
c(n,i); " "; NEXT n: RANDOMIZE
USR 35000
680 FOR n=1 TO 6: PRINT AT 6+n*
2,c(n,i); " ";AT 7+n*2,c(n,i);
" "; NEXT n: RANDOMIZE USR 35
000
685 IF arr<=31 THEN PRINT AT 20
,arr: INK 7: "
690 NEXT i: PRINT #0;AT 1,0;" P
REMI UN TASTO PER CONTINUARE "
PAUSE 0
695 RETURN
699 REM ++ SUB ORDINE ARRIVO++
700 BRIGHT 0: BORDER 3: BRIGHT
1: PAPER 3: CLS: INK 7
705 LET t$="***** ORDINE DI AR
RIVO *****": LET s$="*****
*****"
710 PLOT 2,2: DRAW 253,0: DRAW
0,173: DRAW -253,0: DRAW 0,-173:
PLOT 1,174: DRAW 0,-173: DRAW 2
53,0: PLOT 0,173: DRAW 0,-173: D
RAW 253,0
715 FOR n=1 TO 30: PRINT INK 6;
AT 2,n;t$(n):AT 1,n;s$(n):AT 3,n
:s$(n): BEEP .05,-50: PAUSE 3: N
EXT n
720 FOR n=1 TO 6
725 IF n=1 OR n=2 THEN FLASH 1
730 PRINT ;AT 5+n*2,6;v(n);"-":
BEEP .05,0
735 FOR i=1 TO 10: PRINT AT 5+2
*n,7+i;c$(v(n),i): BEEP .05,-50:
PAUSE 3: NEXT i
740 FLASH 0: NEXT n: BEEP .5,10
745 PRINT #0;AT 1,0;"PREMI UN T
ASTO PER CONTINUARE": PAUSE 0
750 BEEP .3,20: RETURN
799 REM ++SUB VINCITE++
800 FOR n=1 TO ng
805 IF u(n,1)=v(1) THEN LET s(n)
=s(n)+u(n,3)+u(n,3)*u(n,4)
810 IF u(n,1)=v(2) AND u(n,2)=2
THEN LET s(n)=s(n)+u(n,3)+u(n,3
)*u(n,4)
811 LET u(n,3)=0: NEXT n
812 FOR n=1 TO 6
813 IF n=v(1) OR n=v(2) THEN NE
XT n: RETURN
814 IF p(n,1) OR p(n,2)<=.5 THE
N LET p(n,1)=1.5: LET p(n,2)=1.5
815 LET p(n,1)=p(n,1)-.2: LET p
(n,2)=p(n,2)-.1: NEXT n: RETURN
899 REM ++SUB MUSICA++
900 RESTORE 910: FOR n=1 TO 14:
READ x,y: BEEP x,y: NEXT n
905 PAUSE 50: BEEP .5,20: RETUR
N
910 DATA .20,12,.50,17,.06,59,.
06,12,.06,59,.09,17,.09,59,.03,1
2,.03,19,.09,17,.09,59,.03,12,.0
3,59,1,17

```

153 byte che ridefiniscono i caratteri grafici a partire dall'indirizzo 65368, ed il loro salvataggio con SAVE "udg" CODE 65368, 153.

Il programma, subito dopo il RUN, chiede se si vogliono inserire i nomi dei cavalli; in caso di risposta negativa assegna sei nomi predefiniti. Successivamente vanno specificati il numero dei giocatori ed i loro nomi, che non dovranno essere lunghi più di sei caratteri ciascuno.

Da questo momento inizia la fase riservata alle scommesse; ogni giocatore può puntare su un solo cavallo per corsa, vincente o piazzato.

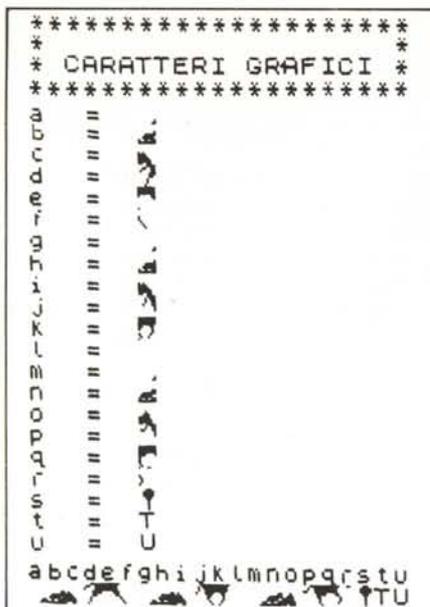
Terminate le puntate bisogna attendere per una manciata di secondi l'inizio della corsa, questo perché il vincitore viene determinato prima della partenza dei cavalli sullo schermo, altrimenti la lentezza della corsa sarebbe insostenibile. Sul rendimento dei cavalli influisce anche la condizione del terreno, visibile in alto a destra sul video al momento delle puntate, che viene

determinata casualmente da un'apposita subroutine.

Elenco delle variabili principali

GS: nomi dei giocatori
CS: nomi dei cavalli
TS: stato del terreno
P: prestazioni
Q: quote
NG: numero giocatori
U: puntate giocatori
V: ordine di arrivo

Questo programma è disponibile su cassetta presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 151.



A sinistra in alto potete vedere la schermata principale del programma Horses; in basso il programma che la genera. A destra si trovano i listati con i codici decimali delle routine in linguaggio macchina ed i codici dei caratteri grafici.

```
*****
*          CODICI L/M          *
*          *****          *
33      0      64      22      64
31      126    229    35      78
113     35     16     248    119
21      32     240    201     0
64      156    17      0      64
0       27     237    176    201
```

Codici decimali della routine in L/M.

```
*****
*          CODICI UDG          *
*          *****          *
0       0      0      0      0
1       3      0      0      0
31      62     147    255    0
0       150    240    248    252
15      51     67     7      6
8       16     255    255    255
1       0      0      0      120
0       128    128    64     32
0       0      0      0      0
1       3      0      0      0
31      63     147    255    0
0       150    240    248    252
15      23     19     35     1
0       0      255    255    255
130     2      132    72     128
0       0      0      0      0
0       0      0      0      0
1       3      0      0      0
63      124    147    255    0
0       16     56     248    252
15      19     99     148    1
0       0      255    255    255
128     128    128    64     140
0       128    64     64     128
28      62     62     23     8
8       8      0      254    16
```

Codici decimali dei byte dei caratteri grafici ridefiniti.

Listato 3

```
1 REM *****
2 REM *
3 REM * GENERATORE *
4 REM * SCHERMATA *
5 REM *
6 REM *****
7 REM
10 BRIGHT 1: PAPER 4: CLS : BO
RDER 0: INK 7
20 PLOT 0,167: DRAW 255,0: PLO
T 0,160: DRAW 255,0: PLOT 0,159:
DRAW 255,0: FOR i=0 TO 255 STEP
3: PLOT i,160: DRAW 0,7: NEXT i
30 PLOT 0,117: DRAW 255,0: PLO
T 0,118: DRAW 255,0: FOR i=1 TO
255 STEP 15: PLOT i,112: DRAW 0,
7: PLOT i+1,112: DRAW 0,7: NEXT
i
35 FOR x=255 TO 0 STEP -45: PL
OT x,127: DRAW 0,32: PLOT x-1,12
7: DRAW 0,32: NEXT x
40 FOR y=123 TO 156 STEP 6: FO
R x=3 TO 252 STEP 6
50 CIRCLE x,y,3
70 NEXT x: NEXT y
80 FOR i=22526 TO 22559: POKE
i,104: NEXT i
90 FOR i=22560 TO 22591: POKE
i,87: NEXT i
100 FOR i=22592 TO 22591+32*5:
POKE i,15: NEXT i
110 SAVE "SCHERMATA"SCREEN$
```

Modifiche e Prospettive

di Adriano Gandolfo - Borgaretto (TO)

Dopo aver provato il programma "Prospettiva" pubblicato sul numero 38 ho pensato di aggiungere alcune opzioni al menu principale per renderlo più versatile.

Le nuove opzioni sono:

- 1) Uscita dei dati su stampante.
- 2) La possibilità di salvare su nastro tutti i dati riguardanti l'oggetto appena definito
- 3) Caricamento dei dati da nastro
- 4) Cancellazione dei dati in memoria, in modo da permettere l'inserimento di nuovi dati
- 5) Uscita dal programma

Analisi del listato

9: Variazione del clic dei tasti, definizione della variabile pr

15: Presentazione del programma

16-18-28: Queste linee vanno cancellate

dal programma originale

31: Input per la scelta delle opzioni. A quelle già esistenti si aggiungono:

Uscita dei dati su stampante	(P)
Save dei dati su nastro	(R)
Load dei dati su nastro	(L)
Cancellazione memoria	(M)
FINE	(F)

32-50: Controllo dell'opzione scelta con invio alla opportuna routine

80-89: Routine di ingresso dati; la prima volta richiede il numero dei nodi dimensionando la matrice dei dati, e pone pr=1

3090-3110: Routine di stampa della matrice di unione, con apertura dello stream 4 che invia i dati verso la stampante o verso il video a seconda del valore di d\$

3140-3160: Routine di stampa della matrice delle coordinate

5001-5007: Routine per il salvataggio dei dati; per salvare su nastro solamente le variabili basta "far credere" al computer che tra PROG e VARS (le variabili di sistema che indicano, rispettivamente, l'indiriz-

zo iniziale del programma e l'indirizzo iniziale dell'area variabili) non vi sia nulla. Per fare ciò basta rendere PROG uguale a VARS con le istruzioni

POKE 23635,PEEK 23627:POKE 23636,PEEK 23628

immediatamente prima del SAVE, e poi ripristinare il vecchio valore con

POKE 23635,203:POKE 23636,92 (questo però solo a patto che non siano collegati l'interfaccia 1 ed i microdrive; un metodo più corretto di ripristinare PROG è quello di salvare il valore originale prima della modifica con

LET PROG1=PEEK 23635:LET PROG2=PEEK 23636

e di eseguire dopo il SAVE le istruzioni POKE 23635,PROG1:POKE 23636,PROG2 N.d.r.)

5010-5015: Routine di caricamento dei dati con un MERGE

5020-5024: Routine per la cancellazione di dati in memoria

5030-5034: Uscita dal programma **MC**

```

1 REM *****
2 REM * NUOVE OPZIONI PER *
3 REM * IL PROGRAMMA *
4 REM * PROSPETTIVA *
5 REM *
6 REM *****
7 REM
8 LIST
9 POKE 23609,40: LET pr=0
15 PRINT " STUDIO PROSPE
TTIVE " " VERSIONE 2.0":
PAUSE 80
16 REM Queste 3 linee vanno
18 REM cancellate
20 REM dal programma originale
31 INPUT "Ingresso dati

```

```

[C] Correzione dati
[R] Stampa dati
[D] Disegno
[P] Uscita su stampante
[S] Save dati su nastro
[L] Load dati da nastro
[M] Cancellazione memori
[F] FINE

```

```

32 IF a$="P" THEN LET d$="p":
GO TO 370
33 IF a$="R" THEN GO TO 5001
34 IF a$="L" THEN GO TO 5011
35 IF a$="M" THEN GO TO 5021
36 IF a$="F" THEN GO TO 5031
40 IF a$="S" THEN LET d$="s":
GO TO 370
50 IF a$="I" THEN GO TO 81
80 REM ++INGRESSO DATI++
81 IF pr=1 THEN GO TO 90
83 CLS: BEEP 0.7,0: INPUT "NU
mero nodi ";k: LET pr=1
84 LET pv=1: LET up=0: LET nn=
0
89 DIM t(k,4): DIM c(k,3): DIM
b(3,4): DIM a(2,3): DIM d(2,3):
DIM g(2,3)
3090 OPEN #4,d$: CLS: PRINT "
MATRICE DI UNIONE"
3095 PRINT #4;AT 2,0;"nodo pr";A
T 2,9;"nodo cn";AT 2,17;"nodo cn
";AT 2,25;"nodo cn": PRINT #4
3110 PRINT #4;TAB 3,i;TAB 11;c(i

```

```

1);TAB 19;c(i,2);TAB 27;c(i,3)
3140 OPEN #4,d$: CLS: PRINT "
MATRICE COORDINATE"
3145 PRINT #4;AT 1,0;"c";AT 1,5;
"x";AT 1,13;"y";AT 1,21;"z";AT 1
,29;"t4"
3149 PRINT #4
3160 PRINT #4;TAB 0,i;TAB 5;t(i,
1);TAB 13;t(i,2);TAB 21;t(i,3);T
AB 30;t(i,4)
5000 REM ++SAVE SU NASTRO++
5001 BEEP 0.1,0: CLS: INPUT "Ch
e nome vuoi dare? ";n$
5002 IF LEN n$>10 OR LEN n$<1 TH
EN GO TO 5001
5003 PRINT "Avvia il nastro e pr
emi un tasto per salvare"; FL
ASH 1;"";n$;""; FLASH 0;"Las
cia scorrere il nastro!"
5004 POKE 23635,PEEK 23627: POKE
23636,PEEK 23628
5005 SAVE n$
5006 POKE 23635,203: POKE 23636,
92
5007 PAUSE 100: GO TO 30
5010 REM ++LOAD DA NASTRO++
5011 BEEP 0.1,0: CLS: INPUT "Ch
e nome hai dato? ";n$
5012 IF LEN n$>10 OR LEN n$<1 TH
EN GO TO 5011
5013 PRINT "Avvia il nastro sto
per caricare "; FLASH 1;"";n$;
" "; FLASH 0;"lascia scorrere i
l nastro!"
5014 MERGE n$
5015 GO TO 30
5020 REM ++NEW RUN++
5021 BEEP 0.1,0: CLS: INPUT "Se
i sicuro di voler cancellarei
dati?";s$
5022 IF s$="S" THEN RUN
5023 IF s$<>"N" THEN GO TO 5021
5024 GO TO 30
5030 REM ++FINE++
5031 BEEP 0.1,0: CLS: INPUT "Se
i sicuro di voler finire?";s$
5032 IF s$="S" THEN INK 0: PAPER
7: BORDER 7: CLS: GO TO 9999
5033 IF s$<>"N" THEN GO TO 5031
5034 GO TO 30

```



Elenco del software disponibile su cassetta o minifloppy

Per ovviare alle difficoltà incontrate da molti lettori nella digitazione dei listati pubblicati nelle varie rubriche di software sulla rivista, MCmicrocomputer mette a disposizione i programmi più significativi direttamente su supporto magnetico. Riepiloghiamo qui a fianco i programmi disponibili per le varie macchine, ricordando che i titoli non sono previsti per computer diversi da quelli indicati.

Il numero della rivista su cui viene descritto ciascun programma è riportato nell'apposita colonna; consigliamo gli interessati di procurarsi i relativi numeri arretrati, eventualmente rivolgendosi al nostro Servizio Arretrati utilizzando il tagliando pubblicato in fondo alla rivista.

Per l'ordinazione inviare l'importo (a mezzo assegno, c/c o vaglia postale) alla Technimedia srl, Via Carlo Perrier 9, 00157 Roma.

Le cassette utilizzate sono Basf C-60 Compusette II; i minifloppy sono Basf singola faccia singola densità.

Codice	Titolo programma	MC n.	Prezzo	Note

APPLE II				
DA2/00	Shape Tablet	22	15000	
DA2/01	Motomuro	26	15000	
DA2/02	WDEBUG	28	15000	
DA2/03	EDIT + INPUT	29	15000	
DA2/04	Basic modulare	34	15000	
DA2/05	ANNA Animation Lang.	35/37	15000	
DA2/06	Miniset + Leva-DOS	37	15000	
DA2/07	27 programmi grafici	38	30000	
DA2/08	Adventure Editor	38	15000	
DA2/09	Animazione funzioni	42	15000	
DA2/10	Il mondo di VA-TOR	43	15000	
DA2/11	Contest LOG	43	15000	
DA2/12	Rout.grafiche estese	44	15000	

COMMODORE 64				
C64/01	Briscola	25	17000	
C64/02	Serpentone	29	17000	
C64/03	Othello	29	17000	
C64/04	Chase	33	17000	
C64/05	Spreadsheet	34	30000	
C64/06	Bilancio familiare	35	17000	
C64/07	The dark wood	36	17000	
C64/08	Totocalcio: sis.rid.	37	17000	
C64/09	Orchetes	37	17000	
C64/10	Wordprocessor	38	17000	
C64/11	Helicopt	38	17000	
C64/12	Finestra grafica	39	17000	
C64/13	Paroliame	39	17000	
C64/14	Scarabeo	40	17000	
C64/15	Magazzino	41	17000	
C64/16	Rubrica	44	17000	
C64/17	World	45	17000	
D64/01	Spreadsheet	34	15000	
D64/02	ADP Basic	da 35 a	15000	
D64/03	Wordprocessor	38	15000	
D64/04	Paroliame	39	15000	
D64/05	Data base Galileo	40/41	15000	
D64/06	Magazzino	41	15000	

COMMODORE VIC-20				
CVC/01	VIC-Maze	19	17000	Config. base
CVC/02	Pic-Man	23	17000	Config. base
CVC/03	Briscola	25	17000	Config. base
CVC/04	Grand Prix	28	17000	Config. base
CVC/05	Frogger	26	17000	RAM: almeno + 3 K
CVC/06	Invaders	29	23000	RAM: + 16 K
CVC/07	Othello	29	17000	RAM: + 16 K
CVC/08	SKI	31	17000	Config. base
CVC/09	VIC-quiz	32	17000	RAM: almeno + 8 K
CVC/10	Zigurat	33	17000	Config. base
CVC/11	Extended Basic	36	17000	RAM: + 16 K
CVC/12	Fireman	36	17000	Config. base
CVC/13	Accordi per chitarra	39	17000	RAM: almeno + K
CVC/14	Piramide di Iunnuh	39	17000	RAM: almeno + K
CVC/15	Il castello	40	17000	RAM: + 16 K
CVC/16	Tool grafico	43	17000	RAM: + 16 K
DVC/01	EXMA	27/28	15000	RAM: + 16 K

MSX				
CNX/01	Sound editor	42	17000	
CNX/02	WP Reporter	43	30000	
CNX/03	Foresta maledetta	44	17000	
CNX/04	Monitor disassembler	45	17000	

SINCLAIR SPECTRUM				
CSS/01	TRILAB	28	17000	
CSS/02	SET di caratteri	27/29	17000	
CSS/03	Grafica TREDIM	29	17000	
CSS/04	Ippica	30	17000	
CSS/05	Graphic-Comp	32	17000	48 K RAM
CSS/06	Macchina del tempo	34	17000	48 K RAM
CSS/07	Piramide di Iunnuh	35	17000	48 K RAM
CSS/08	Over Basic	37	17000	48 K RAM
CSS/09	Prospettiva	38	17000	48 K RAM
CSS/10	Motomuro	39	17000	48 K RAM
CSS/11	Othello	40	17000	
CSS/12	The dark wood	40	17000	48 K RAM
CSS/13	Musica	41	17000	48 K RAM
CSS/14	Calcolo matriciale	42	17000	48 K RAM
CSS/15	Database	42	17000	
CSS/16	Snake	43	17000	
CSS/17	Life	44	17000	
CSS/18	Horses	45	17000	48 K RAM

TEXAS TI-99/4A				
CT9/01	Macchina del tempo	27	17000	
CT9/02	Simon	29	17000	
CT9/03	Babilonia	30	17000	
CT9/04	Labirinto 3D	31	17000	
CT9/05	Piramide di Iunnuh	33	17000	Extended Basic
CT9/06	Scrabble	34	17000	
CT9/07	Morphy	35	17000	
CT9/08	Equo canone	37	17000	
CT9/09	Scopa	39	17000	
CT9/10	Montecarlo	39	17000	Extended Basic
CT9/11	Totocalcio	41	30000	

Nota:				
l'iniziale del codice e' C per le cassette, D per i minifloppy				

DISEGNA E SCRIVE



GRAF

**MEGLIO
DI UNA STAMPANTE,
COSTA POCO, LA
TECNOLOGIA È
GIAPPONESE, MA VIENE
DA SAN MARINO:
PS-80, PLOTTER PER
HOME A 80 COLONNE!**



IL PLOTTER DI FAMIGLIA.

- Disegna e scrive (caratteri da mm. 1,2 a mm. 75) a 4 colori su carta, cartoncino fino a 21 cm. di larghezza (formato A4) e carta a rullo.

- Il PS-80 è munito di una porta parallela Centronics e di porta seriale RS232 (optional). Quindi è compatibile con la grande maggioranza degli Home e Personal disponibili sul mercato. Sono inoltre disponibili interfacce per i seguenti modelli:

APPLE II/II+/IIe/IIc
COMMODORE 64/16/PLUS4
SINCLAIR ZX/QL

MINICAD

Un programma di Grafica in Alta Risoluzione capace di generare solidi tridimensionali e di manipolarli con varie funzioni tra i quali: Ingrandimento, Riduzione, Rotazione, Duplicazione, ecc. Tutti i disegni ottenuti con MINICAD possono essere memorizzati su disco e/o stampati con il Plotter Stampante PS-80. Con le dovute interfacce possono essere utilizzati il C 64, C 16, PLUS/4, C 128.

Realizzato dalla
LEONI INFORMATICA MILANO

SOFT DUMP

È il programma che consente di disegnare con il PS-80 le immagini del video.

L'Hard-copy del video viene eseguito nei 4 colori disponibili sul PS-80.

- Programma su disco o nastro

- Istruzioni in italiano

Realizzato da COMPUTRON-SHOP - ROMA.

- Garanzia di un anno
- Assistenza e ricambi garantiti dalla ACS della Repubblica di San Marino.

Agenti ACS

Milano e Provincia
Passarella Giuseppe
C.so Lodi, 111
20137 Milano
Tel. 02/5691653

Modena/Reggio E./Parma
Computer House
Via Secchi 28/B
42100 Reggio Emilia
Tel. 0522/35890-35872

Marche
Ing. Bona Luciano
Via Manzoni, 44
60131 Ancona
Tel. 071/890292

Roma Città
Computron
Largo Forano, 7/8
00199 Roma
Tel. 06/8391556

ACS

ADVANCED COMPUTER SYSTEMS

Via Nonaguardaria, 24
47031 Cailungo Repubblica di San Marino
Tel. 0541/902459

software

SHARP PC-1500

Abbiamo ricevuto da Maurizio Sichera di Milano, lo stesso autore di SGV1500 pubblicato sul n. 43, il CALC1500.

Come dice il nome stesso, si tratta di uno spreadsheet elettronico realizzato per il pocket Sharp, dalle interessanti caratteristiche e prestazioni.

Data la quantità del materiale descrittivo e dei listati concernenti questo lavoro abbiamo preferito proporlo ai lettori in due puntate: ciò è stato inoltre semplificato dal fatto che il programma è realizzato con elementi modulari che ne facilitano l'impiego e la comprensione.

CALC1500

(prima parte)

di Maurizio Sichera - Milano

CALC1500 è un programma di foglio elettronico per il calcolatore tascabile SHARP PC-1500, che semplifica notevolmente la soluzione di tutti quei problemi in cui i dati ed i risultati possono essere organizzati sotto forma di tabella. Usando CALC1500, il programmatore può concentrare la sua attenzione sui calcoli che risolvono il suo problema, senza preoccuparsi dell'introduzione dei dati, dell'esame dei risultati, e della loro stampa: a queste operazioni, semplici ma noiose, provvede appunto CALC1500.

Il video del PC-1500 viene usato come una finestra che si sposta sopra la tabella e dà accesso ad un elemento per volta: l'utente può esaminare i valori dei vari elementi e modificarli con una procedura molto simile a quella normalmente usata per correggere un programma Basic. È anche possibile riempire un certo numero di elementi contigui con una successione di valori in progressione aritmetica o geometrica.

È poi possibile stampare il contenuto della tabella sulla stampante CE-150, in modo completo o selettivo, utilizzando un formato standard o specificando un certo numero di personalizzazioni.

È infine possibile salvare il contenuto della tabella su cassetta e rileggerlo in seguito, completamente o in modo selettivo.

Una caratteristica importante di CALC1500 è che tutte le personalizzazioni, che permettono di risolvere un particolare problema e di impaginare la stampa

nel modo più opportuno, devono essere scritte in linguaggio Basic: questa soluzione è quella che sfrutta al meglio le risorse del PC-1500, ed inoltre permette la massima flessibilità.

1.2. Configurazione richiesta

Il programma CALC1500 richiede l'unità base PC-1500 o PC-1500A, la stampante/ interfaccia CE-150 ed una opportuna espansione di memoria. La memoria necessaria non è facilmente quantificabile in quanto dipende dalla complessità del problema e dalla quantità di dati: a titolo puramente indicativo si può dire che con il PC-1500A + CE-150 si risolvono problemi di media complessità.

1.3. Composizione di CALC1500

CALC1500 è costituito da 5 programmi Basic ed un programma in linguaggio macchina:

CALC1500/VID è il programma di esame, introduzione e modifica dei dati su video;

CALC1500/150 è il programma di stampa su CE-150;

CALC1500/SCA è il programma di scrittura su cassetta;

CALC1500/LCA è il programma di lettura da cassetta;

CALC1500/UTE è un prototipo di programma applicativo che deve essere modificato caso per caso;

SGV1500 (Sottoprogramma di Gestione del Video) pubblicato su MC n. 43.

Per un comodo uso di CALC1500, risulta infine quasi indispensabile il programma PCFILE (in L/M), di Fulvio Peruggi, pubblicato sulla rivista "MCmicrocomputer" del dicembre 1984, o la versione migliorata dello stesso programma, pubblicata nella stessa rivista nel maggio 1985.

1.4. Operazioni preliminari

1) Introdurre in memoria il programma in L/M PCFILE all'indirizzo più basso possibile: per esempio, nel caso si abbia l'espansione da 8 kbyte CE-150, PCFILE può occupare le locazioni &38C5 -- &3AA9.

2) Introdurre in memoria il programma in L/M SGV1500 immediatamente dopo PCFILE: continuando l'esempio precedente, SGV1500 occuperà le locazioni &3AAA -- &3C9A (è lungo 497 byte, esadecimale 1F1).

3) Stabilire la base del programma Basic immediatamente dopo SGV1500: secondo il nostro esempio, occorrerà impostare il comando NEW &3C9B.

4) Introdurre in memoria il programma applicativo prototipo con il comando CLOAD "CALC1500/UTE -- V1"; modificare questo programma in modo da risolvere un problema specifico. In alternativa, introdurre manualmente il programma applicativo di utente, usando CALC 1500/UTE come guida.

5) Introdurre in memoria gli altri programmi con i comandi:

- * MERGE "CALC1500/VID-V1A" (sempre)
- * MERGE "CALC1500/150-V1A" (sempre)
- * MERGE "CALC1500/SCA-V1B" (se occorre)
- * MERGE "CALC1500/LCA-V1A" (se occorre)

6) Poiché nessuno è perfetto, occorre sempre essere in grado di modificare il programma applicativo vero e proprio. Se si usa PCFILE, non vi sono problemi. In caso contrario è assolutamente indispensabile che CALC1500/UTE sia il primo programma caricato in memoria e che alla fine dei vari MERGE venga impostato il comando

POKE &7869,PEEK(&7865),PEEK(&7866) per posizionare il puntatore di editing sul primo programma caricato e proteggere da modifiche tutti i programmi seguenti.

Il programma CALC1500/UTE

Esaminiamo ora la lista del programma CALC1500/UTE (v. fig. 1). Le linee iniziali sono riservate per commenti descrittivi: se l'occupazione di memoria non è critica, i commenti sono sempre utili ...

Le linee 100 -- 550 contengono il programma di inizializzazione, che viene attivato con il comando DEF/Z (si può usare RUN, se questo programma è stato caricato per primo).

130 Viene calcolato l'indirizzo iniziale di SGV1500, nell'ipotesi che questo preceda immediatamente il programma Basic (in caso contrario, occorre modificare l'istruzione).

150-160 Le variabili interne a CALC 1500 vengono allocate tutte all'inizializzazione del programma; in questo modo un eventuale overflow di memoria viene scoperto immediatamente, e si può tentare di correre ai ripari.

210 Se si usano funzioni trigonometriche, è bene essere sicuri che il modo angolare sia quello voluto.

220-260 Le variabili OG, IS, FG\$, NR e NC costituiscono i parametri globali di CALC1500: IS e FG\$ riguardano la stampa e verranno spiegate nel capitolo relativo. NR e NC sono le dimensioni della tabella dei dati D(NR,NC); secondo le regole del Basic la tabella ha NR+1 righe, numerate da 0 a NR, e N+1 colonne, numerate da 0 a NC. Questa tabella contiene tutti e soli i dati applicativi conosciuti da

CALC1500, siano essi dati di ingresso, risultati intermedi o risultati finali.

In alcune applicazioni, l'uso di indici che partono da zero può risultare scomodo: in tal caso si può porre $OG=1$ per indicare a CALC1500 che deve ignorare completamente la riga 0 e la colonna 0 della tabella.

310 Viene allocata la tabella dei dati e due vettori alfanumerici destinati a contenere i titoli di riga e di colonna. Questi titoli vengono usati come intestazione del tabulato dal programma di stampa; i primi 7 caratteri di ogni titolo vengono usati dal programma di visualizzazione e modifica per identificare l'elemento che compare sotto la finestra. La lunghezza massima delle stringhe memorizzabili in R\$ e C\$ può essere stabilita in modo esplicito, ma comunque non può superare 16.

410-430 A questo punto si devono inizializzare i titoli di riga e di colonna (se si usano istruzioni READ, occorre aggiungere le frasi DATA) e si può inizializzare, in tutto o in parte, la tabella dei dati. Si noti che i titoli di riga e di colonna possono anche essere letti da cassetta.

510-550 Fine del programma di inizializzazione. Se si preferisce, l'istruzione END può essere sostituita con GOTO "D", per passare immediatamente al programma di visualizzazione e modifica.

A partire dalla linea 1000 si possono inserire dei sottoprogrammi di personalizzazione del formato di stampa, che verranno descritti nel capitolo relativo.

A partire dalla linea 4000 si possono inserire dei sottoprogrammi di controllo della lettura selettiva da cassetta: anche questi verranno descritti più avanti.

Le linee 7000 -- 7175 contengono delle funzioni di utilità che possono essere usate dal programma applicativo di calcolo:

"MAX" Massimo di un gruppo di elementi

"MIN" Minimo di un gruppo di elementi

"AVG" Media di un gruppo di elementi

"SUM" Somma di un gruppo di elementi

"RND0" Arrotondamento di un gruppo di elementi verso l'unità inferiore in valore assoluto

"RND9" Arrotondamento di un gruppo di elementi verso l'unità superiore in valore assoluto.

"RND5" Arrotondamento di un gruppo di elementi verso l'unità più vicina, fatto in modo da poter essere usato per le dichiarazioni IRPEF, IVA, ecc.

Tutti questi sottoprogrammi hanno alcune caratteristiche in comune:

1) Lavorano su un rettangolo di elementi caratterizzato da una riga iniziale "ri", una riga finale "rf", una colonna iniziale "ci" ed una colonna finale "cf" (con $ri \leq rf$ e $ci \leq cf$).

2) Come argomento ai sottoprogrammi viene passata la variabile alfanumerica AS, con il formato "rirfci" (sempre 2 cifre per ogni valore).

3) Usano le variabili numeriche T, U, V, W, X, Y, Z e ne distruggono il contenuto.

4) I primi quattro sottoprogrammi lasciano il risultato nella variabile Z.

5) Gli ultimi tre sottoprogrammi arrotondano ogni elemento e riscrivono il risultato al suo posto. La variabile N deve contenere il numero di cifre decimali (ad esempio, $N=2$ per arrotondare al centesimo, $N=0$ per arrotondare alle unità, $N=-3$ per arrotondare alle migliaia).

Alla linea 10000 comincia infine il programma di calcolo applicativo vero e proprio, che deve essere scritto dall'utente e su cui quindi c'è poco da dire. Questo programma viene attivato con il comando DEF/= e può quindi ricevere un parametro (istruzione AREAD); se si preferisce, anche qui l'istruzione END può essere sostituita da GOTO "D". Questo programma può usare liberamente tutte e sole le variabili A-Z e AS-ZS.

Il programma CALC1500/VID

3.1. Uso

Questo programma permette l'esame e la modifica dei dati contenuti nella tabella, in modo molto simile alla modifica di un programma Basic. Permette inoltre di riempire un gruppo di elementi contigui della tabella con una successione di valori, in progressione aritmetica o geometrica.

Il programma viene attivato con il comando DEF/D ed immediatamente sul video compare la scritta

```
xxxxxxx/yyyyyyy = vvvvvvvvvvv
```

dove "xxxxxxx" rappresenta un titolo di riga, "yyyyyyy" un titolo di colonna (entrambi troncati a 7 caratteri) e "vvvvvvvvvv" il valore dell'elemento corrispondente. Il cursore non è visibile: il programma è in stato di esame. In questo stato è possibile spostare la finestra nelle 4 direzioni, usando i quattro tasti che circondano il tasto SPACE (tutti e quattro i tasti si autoripetono, se premuti abbastanza a lungo)

Freccia in su — sposta la finestra in alto (righe decrescenti)

Freccia in giù — sposta la finestra in basso (righe crescenti)

Selezione Riserva — sposta la finestra a sinistra (col. decrecenti)

RCL — sposta la finestra a destra (colonne crescenti)

Sono anche possibili spostamenti della finestra su posizioni prestabilite utilizzando i tasti numerici: i tasti 7, 9, 1 e 3 spostano la finestra nei quattro angoli della tabella, i tasti 8 e 2 la spostano contro il bordo con un movimento orizzontale, infine il tasto 0 sposta la finestra sul primo elemento della riga successiva (va a capo).

Premendo uno dei tasti di spostamento del cursore, appare il cursore sul video: il programma passa in stato di modifica e si può correggere il valore dell'elemento su cui è posizionata la finestra.

Conviene ora definire in modo preciso l'effetto di ogni tasto nei due stati del programma (vedi tabella di pag. 156).

Tutti gli altri caratteri (cioè OFF, MODE, SHIFT/MODE, SHIFT/CL e i vari DEF/lettera) sono illegali in entrambi gli stati e provocano un BEEP di protesta.

I dati vengono acquisiti come stringhe di caratteri e poi convertiti mediante la funzione VAL: pertanto non sono ammesse espressioni, ma semplici valori numerici. Sono tuttavia disponibili limitate capacità aritmetiche, secondo le seguenti regole:

1) Quando il programma è in stato di esame, i caratteri + - * / e circonflesso vengono interpretati come operatori aritmetici, i cui operandi sono il valore preesistente dell'elemento ed il numero introdotto da tastiera.

2) Quando il programma è in stato di esame, il carattere "radice quadrata" provoca la sostituzione del valore preesistente con la sua radice quadrata.

3) In caso di errore nei calcoli di cui sopra, il valore preesistente non viene modificato.

4) Quando il programma è in stato di esame, il carattere "pi greco" provoca la sostituzione del valore preesistente con il valore 3.14 ...

3.2. Passaggio ad altri programmi

Il tasto DEF/= provoca il passaggio al programma di calcolo; analogamente, i tasti DEF/SPACE e DEF/N provocano il passaggio al programma di stampa. In tutti i casi, la variabile Z\$ viene annullata.

3.3. Introduzione multipla di dati

È possibile attribuire una successione di valori, in progressione aritmetica o geometrica, ad un gruppo di elementi contigui, situati sulla stessa riga (o sulla stessa colonna) della tabella. Si può specificare il valore iniziale ed un incremento (assoluto o percentuale), oppure il valore iniziale, il valore finale ed il tipo di progressione (aritmetica o geometrica).

Per una introduzione multipla sulla stessa riga, specificando il valore iniziale e l'incremento, si procede come segue:

a) si posiziona la finestra sul primo elemento e, se occorre, se ne modifica il valore;

b) si batte il tasto "." per segnalare a CALC1500 la posizione iniziale;

c) si sposta la finestra, a destra o a sinistra, sull'elemento finale;

d) si batte il tasto F1: sul video compare la domanda "Incremento?";

e.1) se si vuole una progressione aritmetica si introduce il valore dell'incremento (può essere positivo o negativo, o anche nullo per avere dati costanti);

e.2) se si vuole una progressione geometrica, si batte il valore dell'incremento percentuale (che può essere positivo o negativo) immediatamente seguito dal carattere "%".

Per una introduzione multipla sulla stessa

sa riga, specificando i valori iniziale e finale, si eseguono i passi (a), (b) e (c) come sopra, poi:

d) si modifica, se occorre, il valore finale;

e.1) se si vuole che i valori intermedi vengano calcolati in progressione aritmetica, si batte il tasto F2;

e.2) se si vuole una progressione geometrica, si usa il tasto F3.

Se al punto (c) si sposta la finestra in modo che il valore finale non è più sulla stessa riga di quello iniziale, le cose funzionano ancora, con qualche variante:

1) l'elemento iniziale e quello finale individuano un rettangolo composto da un certo numero di righe, con una colonna iniziale ed una colonna finale;

2) l'operazione viene ripetuta indipendentemente per ogni riga, utilizzando come valore iniziale (finale) quello che si trova sulla riga e sulla colonna iniziale (finale);

3) se si usa il tasto F1, l'incremento è lo stesso per tutte le righe.

Per una introduzione multipla su una o più colonne, vale tutta la logica descritta fin qui, ma si usano i tasti F4, F5 e F6.

3.4. Commenti al programma

La gestione dell'I/O su video/tastiera è effettuata mediante il programma in L/M SGV1500, descritto separatamente in un'altra nota dello stesso autore. L'inizializza-

zione di CALC1500 ha memorizzato nella variabile GV l'indirizzo del sottoprogramma.

13010 Le coordinate iniziali di una introduzione multipla (K e L) vengono poste uguali ad un valore illegale. Il parametro di controllo per il sottoprogramma SGV1500 (X) viene posto uguale a 3. La finestra viene posizionata nell'angolo superiore sinistro della tabella (R e C).

13020-13055 Costruzione del prompt.

13060-13070 Transazione su video/tastiera.

13075 Si arriva qui solo se il parametro X vale 0 (solo display senza input), cosa che si verifica solo se l'ultimo terminatore è stato un carattere di spostamento della finestra. A questo punto si usa la funzione INKEY\$, in quanto non richiede di rilasciare il tasto e di premerlo nuovamente e permette quindi l'effetto di ripetizione automatico.

13080 Il controllo viene smistato in funzione del codice del carattere terminatore.

13085 Terminatori illegali

13118 Selezione Riserva

13120 Freccia in giù

13122 Freccia in su

13126 ENTER

13128 BREAK

13134 F1

13136 F2

13138 F3

13140 F4

13142 F5

13144 F6

13148 CL

13150 RCL
13184 *
13186 +
13190 -
13192 . (punto)
13194 /
13196 0
13198 1
13200 2
13202 3
13204 4
13208 6
13210 7
13212 8
13214 9
13282 radice quadrata
13286 pi greco
13288 circonflesso
13356 DEF/SPACE
13384 DEF/N
13414 DEF/=

13700-13735 Trattamento del carattere ENTER.
13750-13765 Tratt. del carattere F1.
13770-13785 Tratt. del carattere F4.
13800-13815 Tratt. del carattere F2.
13820-13835 Tratt. del carattere F5.
13850-13865 Tratt. del carattere F3.
13870-13885 Tratt. del carattere F6.
13900-13910 Sottoprogrammi di servizio per le funzioni F1-F6.
13950 Conclusione comune per le funzioni F1-F6.
13970-13980 Trattamento degli errori nelle funzioni F3 e F6.

(continua sul prossimo numero).

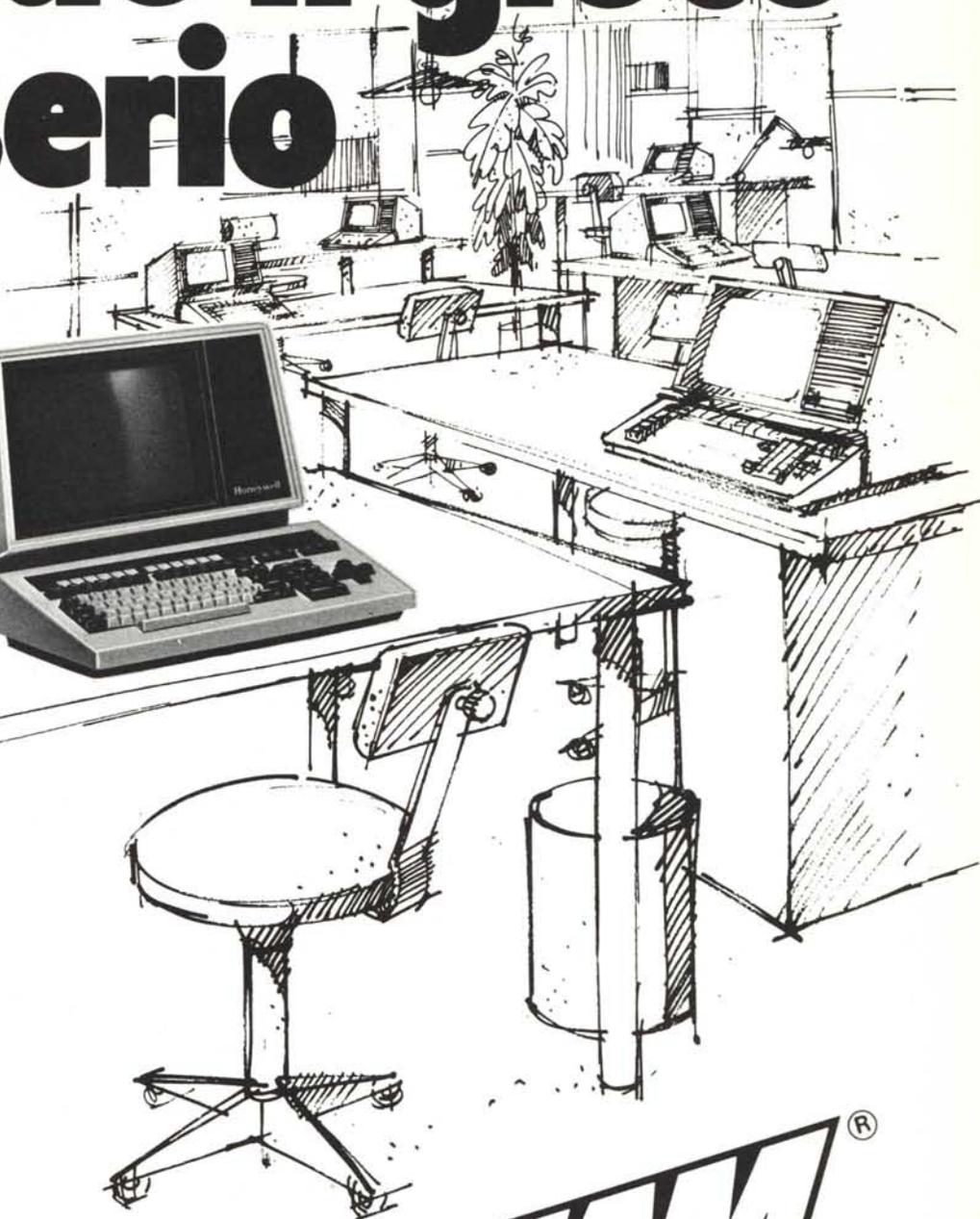


Tasto	Stato di esame	Stato di modifica	Tasto	Stato di esame	Stato di modifica
ON/BREAK	! Interrompe il programma.	! Interrompe il programma.	DEL	! Illegale (BEEP).	! Cancella il carattere sotto il cursore.
SHIFT DEF SML	! Funzionano da prefissi, come al solito.	! Funzionano da prefissi, come al solito.	CL	! Cancella il valore e resta in stato di esame.	! Cancella il valore e resta in stato di modifica.
Selez. Riserva	! Sposta la finestra a sinistra, se possibile.	! Torna in stato di esame perdendo le modifiche, poi sposta la finestra.	F1-F6	! Esegue la funzione di introduzione multipla (vedi oltre).	! Torna in stato di esame perdendo le modifiche, poi esegue la funzione.
RCL	! Sposta la finestra a destra, se possibile.	! Torna in stato di esame perdendo le modifiche, poi sposta la finestra.	. (punto)	! Definisce la posizione iniziale per un comando di introduzione multipla.	! Carattere normale (viene inserito sul video).
Freccia su	! Sposta la finestra in alto, se possibile.	! Torna in stato di esame perdendo le modifiche, poi sposta la finestra.	0 1 2 3 4 6 7 8 9	! Spostano la finestra.	! Caratteri normali (vengono inseriti sul video).
Freccia giu'	! Sposta la finestra in basso, se possibile.	! Torna in stato di esame perdendo le modifiche, poi sposta la finestra.	+ - * / circonflesso	! Operatori aritmetici binari: il programma passa in stato di modifica (vedi oltre).	! Caratteri normali (vengono inseriti sul video).
ENTER	! Termina il programma.	! Assegna il nuovo valore all'elemento sotto la finestra, poi torna in stato di esame.	radice quadrata pi greco	! Operatori aritmetici unari: il programma modifica l'elemento e resta in stato di esame.	! Caratteri normali (vengono inseriti sul video).
Cursore a sin.	! Passa in stato di modifica con il cursore posizionato alla fine del valore da modificare.	! Sposta il cursore a sinistra, se possibile.	altri caratteri stampabili	! Illegali (BEEP).	! Caratteri normali (vengono inseriti sul video).
Cursore a destra	! Passa in stato di modifica con il cursore posizionato all'inizio del valore da modificare.	! Sposta il cursore a destra, se possibile.	DEF/SPACE DEF/N DEF/=	! Provocano il passaggio ai corrispondenti programmi.	! Si perdono le eventuali modifiche, poi si passa ai programmi relativi.
INS	! Illegale (BEEP).	! Apre lo spazio per inserire un carattere.			

Quando il gioco si fa serio

"Multipersonal": un termine che vi giunge nuovo perché è stato appena coniato da Honeywell. Multipersonal è il nuovo X-Superteam, un computer tutto italiano che entra nel mondo dell' "industry standard" perché si avvale dei sistemi operativi Unix™ e Xenix™ che offrono un patrimonio applicativo molto vasto.

Concepito per servire contemporaneamente più utilizzatori intenti a soddisfare esigenze diverse, X-Superteam può interconnettersi per formare una rete locale. Quando il gioco si fa serio, pensate a X-Superteam, un grande computer che sa stare anche sotto il tavolo, se occorre.



X-SUPERTEAM®

IL MULTIPERSONAL