

software

CASIO

Database (Casio PB-100) di Massimiliano Biotti - Bolzano

Il programma gestisce un piccolo ma potente archivio che può essere utilizzato per tutti gli usi richiesti: agenda telefonica, mini-vocabolario, appuntamenti, catalogo prezzi, ecc. Il numero di record disponibili dipende dal loro formato:

formato	lunghezza record	numero record
1	7 caratteri	144
2	14 caratteri	72
3	21 caratteri	48
4	28 caratteri	36

Ogni record consiste in una stringa di una certa lunghezza nella quale vengono memorizzati i vari campi separati da spazi. Il formato dei campi è libero e l'unica regola da osservare è quella di separare i vari campi da uno o più spazi.

Il programma permette di inserire e cancellare i vari record, di eseguire una ricerca su ognuno dei campi, o addirittura su parte di essi, e di salvare e caricare su nastro l'intero archivio. Tutte queste funzioni sono ripartite in memoria in vari sottoprogrammi richiamabili separatamente con i tasti da P0 a P5.

P0: inizializza l'intero archivio cancellando la memoria. Viene richiesto il formato dei record; al prompt DIM=? bisogna rispondere con un numero intero da 1 a 4.

P1: permette di inserire un nuovo record. Al prompt RECORD N=? (dove N è il numero del record da inserire) si risponde con una stringa formata dai vari campi separati da spazi. Se il formato della stringa eccede il formato previsto, bisogna ripetere l'input.

P2: serve a cancellare un dato record. Al prompt RECORD=? si risponde con un numero minore o uguale al numero di record memorizzati e maggiore di 0. Al posto del record

cancellato viene inserito l'ultimo in memoria ed il puntatore di fine file viene diminuito di uno.

P3: esegue la ricerca di un record. Dato il particolare modo in cui i dati sono memorizzati, è possibile effettuare vari tipi di ricerca. Se al momento dell'input (con il prompt DATO=?) si introduce una stringa preceduta e seguita da uno spazio, vengono cercati e stampati tutti i record che hanno un campo coincidente con la stringa introdotta. Se invece lo spazio viene inserito solo prima o dopo la stringa, la ricerca viene effettuata solo su campi che cominciano o terminano con tale dato. Infine, se la stringa viene introdotta senza spazi, vengono cercati i record con un campo che contiene tale stringa. È inoltre possibile condurre la ricerca introducendo come stringa due o più campi separati da spazi.

P4: provvede a salvare l'archivio su nastro.

P5: carica in memoria l'archivio da nastro.

Per evitare errori, il programma provvede ad impedire l'accesso ai sottoprogrammi P2 e P3 quando l'archivio è vuoto ed al sottoprogramma P1 quando esso è pieno. È possibile inoltre visualizzare l'intero contenuto della memoria (ad esempio per cercare i dati da cancellare) utilizzando P3 e battendo come stringa da ricercare uno spazio.

Il programma occupa 532 passi ed è necessario ripartire la memoria con un DEFM 128.

Variabili impiegate:

- A : indice
- BS: usata
- CS: usata
- DS: usata
- ES: usata
- F : lunghezza dato da ricercare
- G : indice
- H : usata
- I : puntatore di fine file
- J : formato
- K-K(143): vettore KS(), contiene i vari record
- S : contiene il record corrente

Poker (Casio FX-750P) di Fabrizio Trezzani Magenta (MI)

Il programma che vi proponiamo gira sull'FX-750 P della CASIO e per esso è sufficiente una sola scheda RAM se ci si avvale di un semplice accorgimento: l'eliminazione delle REM (alle righe: 10 / 65 / 70 / 90 / 125 / 175 / 180 / 215 / 220 / 245 / 255 / 305) che sono state aggiunte per permettere una lettura più scorrevole del listato. In altro caso sono necessarie due schede RAM.

Il gioco del POKER, così com'è stato strutturato, consiste in una variante del poker originale: in primo luogo premettiamo che la partita viene svolta da un solo giocatore. L'utente stabi-

```

P0
1 VAC : I=-1:INPUT
  "FORMATO=",J
2 GOSUB #6:FOR A=
  0 TO I:G=0 TO J-1:G=
  K$(A*J+G):NEXT G
3 IF LEN(G)=7:GOTO 1
4 IF LEN(G)=14:GOTO 2
5 IF LEN(G)=21:GOTO 3
6 IF LEN(G)=28:GOTO 4
7 GOTO 1
P1
1 IF I*J+J=144:END
2 PRINT "RECORD=":
  I+2:":":INPUT
  S:IF LEN(S)>J*7
  THEN 2
3 I=I+1:GOSUB #6:
  K$(A*J+G)=S$(G)
  :NEXT G:GOTO 1
P2
1 IF I=-1:END
2 INPUT "RECORD="
  ,A:A=A-1:IF A>I
  THEN 2
3 FOR G=0 TO J-1:
  K$(A*J+G)=K$(I*
  J+G)
4 K$(I*J+G)="":
  NEXT G:I=I-1:
  GOTO 1
P3
1 IF I=-1:END
2 INPUT "DATO=":S
  :IF LEN(S)>J*7
  THEN 2
3 GOSUB #6:FOR A=
  0 TO I:G=0 TO J-1:G=
  K$(A*J+G):NEXT G
4 S=S+" ":FOR G=1
  TO LEN(S)-F+1
5 IF F<LEN(S):IF
  BS+CS+DS+ES=MID
  (G,F) THEN B
6 NEXT G
7 NEXT A:GOTO 2
8 PRINT "RECORD":
  A+1:H=2:FOR G=2
  TO LEN(S)
9 IF MID(G,1)=" "
  :PRINT MID(H,G-
  H):G=G+1:H=G
10 NEXT G:GOTO 7
P4
1 PUT I,K(143)
P5
1 GET I,K(143)
P6
1 F=LEN(S):FOR A=
  1 TO J:A$(A)=" "
  :G=A*7-6:IF F>G
  +6:A$(A)=MID(G,
  7)
2 IF F>G:IF F<G+6
  :A$(A)=MID(G)
3 NEXT A:RETURN
  
```

Listato del programma Database.

liscie quale sarà la carta più bassa del mazzo che resterà invariata per ogni successiva partita; se si desidera cambiarla sarà quindi necessario rispondere «N» alla domanda «ALTRA PARTITA?» ed eseguire di nuovo il programma. Una volta apparse sul display le 5 carte generate casualmente tramite la funzione RANDOM, il giocatore avrà la possibilità di cambiarne una, alcune o tutte, a sua scelta. In caso di risposta affermativa egli dovrà fornire la posizione delle carte che si vogliono cambiare, a meno che non abbia scelto di cambiarle tutte. La sostituzione delle carte cambiate sarà eseguita e apparirà sul display la nuova serie di carte. Quindi avverrà la verifica dell'eventuale poker; tale controllo verrà effettuato in ogni caso: sia sulle carte di partenza, se non è avvenuta nessuna sostituzione, sia sulle nuove carte, se una sostituzione c'è stata. Nel caso in cui sia stato realizzato un poker apparirà la scritta «!\$! POKER !\$!», ed infine verrà offerta la possibilità di effettuare un'altra partita.

Esposto in sintesi l'andamento del gioco, esaminiamo ora dettagliatamente la struttura del listato.

1-5: Rilevante in queste prime righe è il controllo sull'input richiesto all'utente «CARTA+BASSA (4..10)»? in cui vengono accettati solo valori di G compresi tra 4 e 10; in altro caso la domanda viene formulata nuovamente.

10-65: Viene effettuata la costruzione del mazzo di carte in relazione al valore di G inserito. Dopo il dimensionamento delle due matrici M\$(3,G) e C\$(3,G) e del vettore F\$(4) si procede con due cicli FOR uno interno all'altro che assegnano un valore ad M\$(I,J) occupandosi di un seme alla volta. Il ciclo FOR esterno si occupa delle righe della matrice, quello interno delle colonne.

In ogni riga ritroveremo così le carte da G all'Asso (carta più alta del mazzo) dello stesso seme ed, in ordine, avremo:

- alla riga 0 le carte di picche;
- alla riga 1 le carte di cuori;
- alla riga 2 le carte di quadri;
- alla riga 3 le carte di fiori.

L'assegnazione dell'Asso e delle figure (JACK, QUEEN, KING) è regolata con delle IF che sistemano queste carte rispettivamente in colonna 0, (G-2), (G-1), G. L'assegnazione delle carte che si trovano tra l'Asso e il Jack è anch'essa regolata da una IF. Si nota dal listato che per l'Asso e le figure si è usata la funzione CHR\$ mentre per le altre carte la funzione STR\$; questo perché dell'Asso e delle figure si conosce il valore in codice ASCII, da cui la necessità di convertire tali valori in caratteri, mentre delle altre carte si conosce il valore numerico, da cui la necessità di convertirli in stringhe di caratteri.

Il problema di convertire il valore delle carte in caratteri nasce dal fatto

che il seme è espresso da un carattere e la somma tra numeri e caratteri sappiamo non essere possibile. Concluso il ciclo FOR interno, prima di ritornare a quello esterno, notiamo l'incremento del valore di C in modo che alla riga successiva compaia un altro seme.

70-90: Azzeramento della matrice di controllo. Ad ogni elemento di questa matrice viene inizialmente assegnato il valore «0».

95-115: Inizializzazione di alcune variabili: «pulizia» dei vettori O,V,P e della variabile W, necessaria per aumentare la memoria disponibile; estrazione delle carte e loro memorizzazione in F\$(4); visualizzazione delle carte estratte.

Può essere interessante esaminare qui la subroutine di riga 180 per ottenere una visuale più omogenea del listato:

180-215: Estrazione delle carte. All'inizio della partita ne saranno estratte necessariamente 5; a partita in corso il loro numero dipenderà dalla richiesta dell'utente. Utilizzando le funzioni INT e RND saranno scelte casualmente delle coppie di valori: l'uno compreso tra 0 e 33, l'altro compreso tra 0 e G, che individuano rispettivamente una riga ed una colonna di M\$ al cui incrocio troviamo la carta estratta. Ovviamente prima che ogni carta

```

1 CLEAR: WAIT 60: LOCATE 6: PRINT "!!! POKER !!!"
5 INPUT "CARTA + BASSA (4..10) ": G: IF G<4 THEN 5
  ELSE IF G>10 THEN 5
10 REM ***COSTRUZIONE DELLE CARTE DA GIOCO***
15 G=14-G: DIM M$(3,G), C$(3,G), F$(4): C=232
20 FOR I=0 TO 3: FOR J=0 TO G
25 IF J=0 THEN M$(I,J)=CHR$(74)+CHR$(C)
30 IF J=(G-2) THEN M$(I,J)=CHR$(74)+CHR$(C)
35 IF J=(G-1) THEN M$(I,J)=CHR$(81)+CHR$(C)
40 IF J=G THEN M$(I,J)=CHR$(75)+CHR$(C)
45 IF J>0 THEN IF J<(G-2) THEN M$(I,J)=STR$(J+13-G) +
  CHR$(C)
50 NEXT J
55 C=C+1
60 NEXT I
65 REM ***
70 REM ***AZZERAMENTO DELLA MATRICE DI CONTROLLO***
75 FOR I=0 TO 3: FOR J=0 TO G
80 C$(I,J)="0"
85 NEXT J: NEXT I
90 REM ***
95 X=0: U=0: D=0: C=4: ERASE O,V,P,W: GOSUB 185
100 FOR K=0 TO 4
105 F$(K)=M$(O(K),V(K))
110 NEXT K
115 WAIT 999: LOCATE 2: PRINT F$(0): " "; F$(1): " ";
  F$(2): " "; F$(3): " "; F$(4)
120 D=D+1: IF D>1 THEN 260 ELSE 250
125 REM *** CAMBIO DELLE CARTE ***
130 INPUT "QUANTE CARTE (1,2,3,4,5) ": C
135 C=C-1: ERASE O,V: GOSUB 185
140 IF C=4 THEN 100
145 DIM P(C): INPUT "POSIZIONE (1,2,3,4,5) ": P(O)
150 IF C<>0 THEN GOSUB 225
155 FOR K=0 TO C
160 F$(P(K)-1)=M$(O(K),V(K))
165 NEXT K
170 LOCATE 2: PRINT F$(0): " "; F$(1): " "; F$(2): " ";
  F$(3): " "; F$(4): GOTO 260
175 REM ***
180 REM *** ESTRAZIONE DELLE CARTE ***
185 DIM O(C), V(C)
190 FOR K=0 TO C
195 O(K)=INT(RND*3+0.5): V(K)=INT(RND*G+0.5)
200 IF C$(O(K),V(K))<>"1" THEN C$(O(K),V(K))="1" ELSE
  K=K-1
205 NEXT K
210 RETURN
215 REM ***
220 REM *** POSIZIONE DELLE CARTE DA CAMBIARE ***
225 FOR K=1 TO C
230 INPUT P(K)
235 NEXT K
240 RETURN
245 *** REM
250 INPUT "CAMBIO CARTE (Y/N) ": W$: IF W$="Y" THEN 130
255 REM *** VERIFICA POKER ***
260 FOR I=1 TO 4
265 IF LEFT$(F$(O),1)=" " THEN D=2 ELSE D=1
270 IF LEFT$(F$(O),D)=LEFT$(F$(I),D) THEN X=X+1
275 NEXT I
280 FOR I=0 TO 3
285 IF LEFT$(F$(4),1)=" " THEN D=2 ELSE D=1
290 IF LEFT$(F$(4),D)=LEFT$(F$(I),D) THEN U=U+1
295 NEXT I
300 IF U=3 THEN 315 ELSE IF X=3 THEN 315
305 REM ***
310 INPUT "ALTRA PARTITA (Y/N) ": W$: IF W$<>"Y" THEN
  END ELSE 75
315 BEEP: BEEP1: WAIT 60: LOCATE 6: PRINT "!!$! POKER
  !$!": GOTO 310

```

Listato del programma Poker.

venga memorizzata verrà effettuato un controllo sulla matrice CS; se nella posizione corrispondente compare uno «0» essa può essere estratta e a tale elemento di CS viene assegnato il va-

lore «1», in caso contrario la carta non viene estratta e si procede ad una nuova scelta.

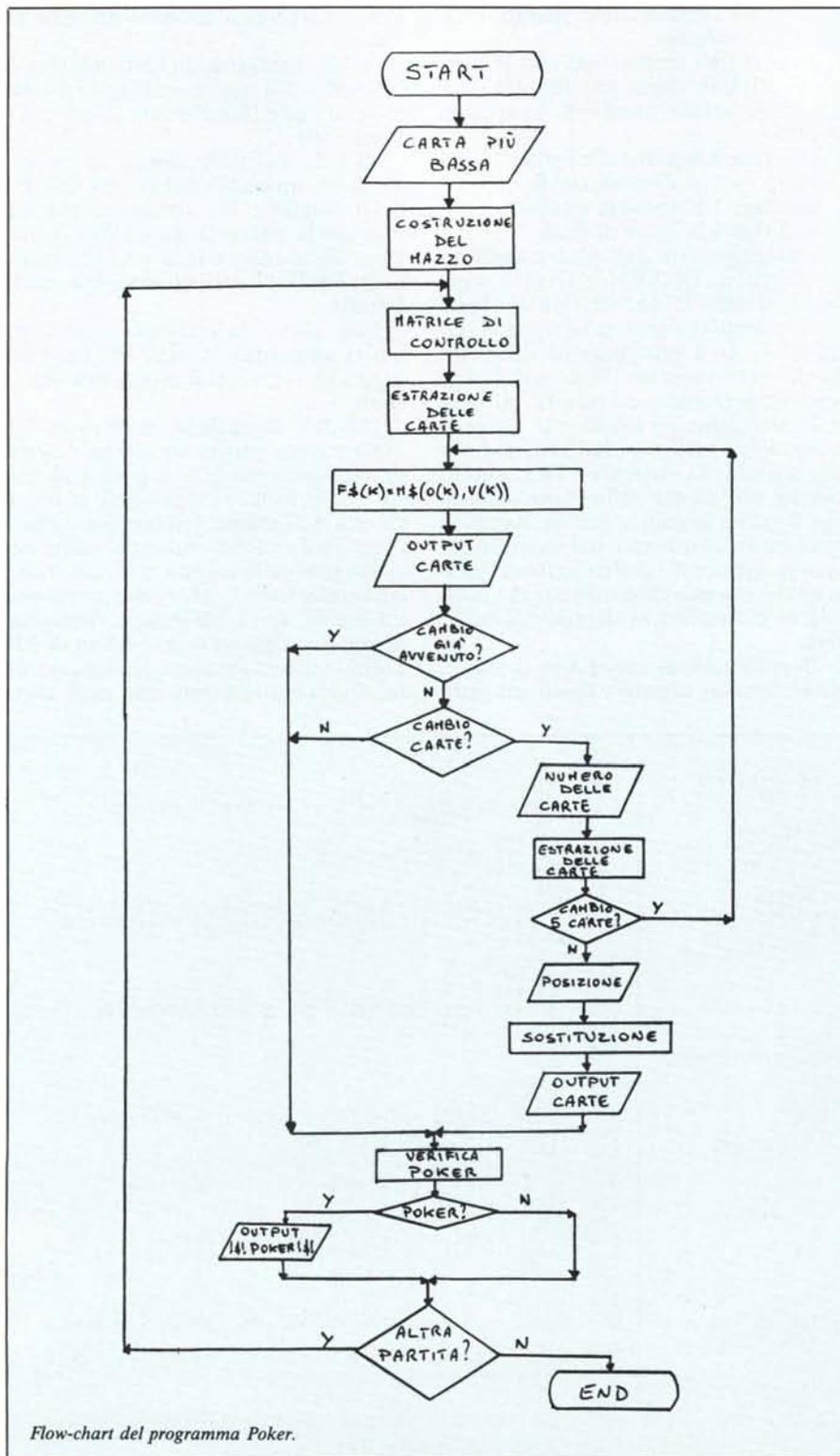
Dopo questa breve parentesi torniamo alla riga 120: dopo la visualizzazione delle carte estratte sarà offerta la possibilità di cambiare le carte solo una volta, per cui la variabile D op-

portunamente incrementata controlla questa parte del gioco. Nel caso in cui un «cambio» sia già avvenuto si procede con la ricerca dell'eventuale poker che più avanti esamineremo.

125-175: Si offre la possibilità di cambiare le carte. A risposta negativa si procede con la ricerca dell'eventuale poker, mentre a risposta affermativa si chiede il numero di carte che si vogliono cambiare. Si procede alla loro estrazione e nel caso in cui l'utente abbia scelto di cambiarle tutte, esse vengono immediatamente memorizzate e visualizzate. In altro caso verrà richiesta la posizione delle stesse che viene automaticamente memorizzata nel caso di una sola carta e se le carte da cambiare sono più di una, un'opportuna subroutine richiederà in input la posizione di ognuna di esse (righe 220-245). Si procede quindi alla memorizzazione delle carte estratte nella loro giusta posizione e l'assegnazione $FS(P(K) - 1) = MS(O(K), V(K))$ evita che vengano perse le carte che non dovevano essere sostituite.

A questo punto le nuove carte vengono visualizzate.

255-315: Si giunge così alla parte finale del programma in cui una subroutine più volte richiamata analizza le 5 carte del giocatore per individuare l'eventuale poker. Tale analisi si basa dapprima sulla lettura della prima carta: se il primo carattere che compare è un blank (" "), nel qual caso la carta potrà essere 4, 5, 6, 7, 8, 9, 10, verrà esaminato solo il secondo carattere di ogni carta che sarà confrontato con il secondo carattere della prima carta; se il primo carattere che compare non è un blank, nel qual caso la carta potrà essere A, J, Q, K, verrà esaminato solo il primo carattere di ogni carta che sarà confrontato con il primo carattere della prima carta. La variabile X fungerà da contatore in questi controlli tenendo conto del numero di carte uguali alla prima (ovviamente ad eccezione del seme). Un controllo dello stesso tipo viene fatto a partire dall'ultima carta che viene quindi usata come riferimento. Questo in seguito alla osservazione che se è stato realizzato un poker, tra le quattro carte uguali devono comparire necessariamente anche la prima o l'ultima o entrambe. Nel controllo fatto sull'ultima carta funge da contatore la variabile U. Infine viene controllato il valore di U e di X e se una di esse è 3 si ha che esistono tre carte uguali alla prima o all'ultima, per cui è stato realizzato un poker. Apparirà la scritta «!\$! POKER !\$!» e in ogni caso sarà offerta la possibilità di effettuare un'altra partita. In caso di risposta negativa l'esecuzione del programma termina.



Flow-chart del programma Poker.

MASTERBIT MIPECO

VENDITA PER
 CORRISPONDENZA



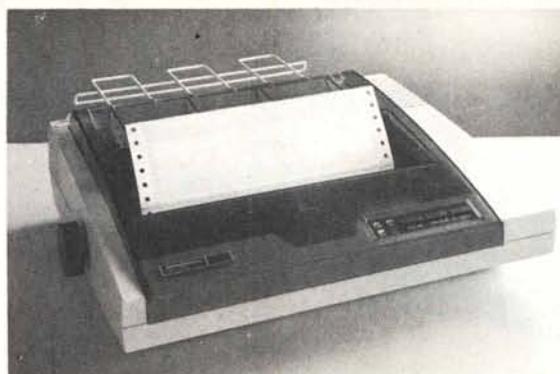
INTERFACCIA PARLANTE CURRAH L. 75.000

Scrivete le parole da pronunciare "Lei" le leggerà:
LET S\$ = "sAlve" enter
sentirete la parola salve dall'altoparlante del T.V.

Molti programmi prevedono già il suo uso (Birds and the Bees, Lunar jet man, ma ziacs, VOICE CHESS ecc.

Compreso nel prezzo manuale completo in italiano più un programma compilatore per farle pronunciare in italiano qualsiasi parola richiesta.

Parla attraverso il televisore con chiara voce sintetica.



MANNESMANN MT 80 + L. 599.000

80 col. - 100 cps - interfaccia Centronics - foglio singolo e modulo continuo - bidirezionale.

QL SINCLAIR 128K 549.000

Tutto compreso
6 mesi di garanzia



CPU MOROTOLA 68008 da 32 BIT e 2 microdrive. Ultima versione con nuovi programmi, alimentatore, manuale in inglese, manuale in italiano, 4 cartucce con 4 programmi gestionali + 1 cartuccia con giochi originali (PIRATE, ZETA, PED, GUN, SREAKOUT, HUNT) e in regalo un ottimo copiatore per mdv e floppy di Massimo Rossi

SPECTRUM 48K PLUS 299.000

Tutto compreso
6 mesi di garanzia

con lo SPECTRUM plus manuale in italiano e in regalo 5 programmi in italiano (conto corrente, grafica, funzioni, bioritmi, esapedone + il Supercopiatore di Massimo Rossi)

QL 512K 880.000

Espansione da 512K montata internamente, non necessita di alimentazione supplementare e lascia il connettore libero per altre periferiche.

Nuovo SPECTRUM 48K + 299.000

Manuale in italiano, cavetti alimentatore, cassette dimostrative e oltre 50.000 lire di software originale e in italiano.

CALCOLATORE SCIENTIFICO SHARP ... 49.000

EL - 506 P. Funz. Trig. BIN-OCT-HEX

10 RULLI di carta termica 29.000

MANNESMANN TALLY tutti i modelli

MT 80 + 599.000

Foglio singolo e continuo, interfaccia Centronics, 100 cps vari set di caratteri - Bidirezionali.

MT85 899.000

Interfaccia Centronics o seriale a scelta 180 cps 80/136 col. foglio singolo e continuo.

DISCHI 3"1/2 13.000

DISCHI 3"1/2 10 pezzi 110.000

Garantiti doppia faccia e doppia densità.

INTERFACCIA PER JOYSTICK

UNA PRESA 25.000

Tipo Kempston, per tutti i joystick stand. 9 PIN D.

INTERFACCIA PER JOYSTICK

DUE PRESE 35.000

Tipo Kempston, per tutti i joystick stand. P PIN D.

JOYSTICK STANDARD 9 PIN D 14.000

CONVERTITORE 99.000

Da RS232 a Centronics per interfaccia 1 o per QL cavi e connettori speciali compresi.

INTERFACCIA CENTRONICS

SPECTRUM 99.000

Senza software tutto su Rom compreso il copy.

8 CARTUCCE x MICRODRIVE 49.000

TRISLOT 27.000

Presi tripla per connettore Spectrum.

MANUALE IN ITALIANO x SPECTRUM .. 16.000

«Come usare il tuo Spectrum».

ROM «JS» NUOVO TIPO (256K + 128K) .99.000

Trasforma il tuo QL in un «JS».

INTERFACCIA PARLANTE CURRAH 75.000

Manuale completo in italiano.

ESPANSIONE x 32K x SPECTRUM 59.000

Issue 2 o 3 specificare, facilissima da montare, istruzioni dettagliate in italiano con fotografie, porta il VS. Spectrum da 16K a 48K. Montaggio gratis.

STAMPANTE ALPHACOM 32 149.000

Per Spectrum ZX 81 istruzioni in italiano 2 rulli di carta in regalo.

DISK DRIVE 3"1/2 x INTERF. x QL 619.000

Oltre 700K formattati.

Monitor ANTAREX - BOXER 12"

Fosfori verdi 215.000

14" colore media risoluzione 450.000

KIT DI ESPANSIONE x QL a 512 249.000

Si monta all'interno del QL, si consiglia l'assistenza di un tecnico specializzato.

ESPANSIONE DEL VOSTRO QL A 512K 349.000

Montata all'interno del vostro QL e collaudata con garanzia di 3 mesi spedite il Computer solo dopo aver avuto un contatto telefonico.

TOOLKIT II x QL SU ROM 89.000

MASTERBIT MIPECO

VENDITA PER
 CORRISPONDENZA

PARTI DI RICAMBIO PER SPECTRUM E QL

GARANZIA 48H: oltre la normale Garanzia di 6 mesi per i Computer e di 3 mesi per gli accessori, la MASTERBIT MIPECO si impegna a sostituire tutto il materiale trovato malfunzionante, entro 48 ore dal ricevimento.

AVVERTENZE - tutti i prezzi sono comprensivi di IVA e spese postali - per ordini inferiori alle 50.000 lire aggiungere L. 5.000 per contributo spese di spedizione - pagamento contrassegno al ricevimento del pacco - è gradito un contatto telefonico - sconti quantità.

MASTERBIT
MIPECO

Viale del Romagnoli 35
00121 OSTIA LIDO RM - CAS. POST. 3016

ORDINI TELEFONICI (ore 8.30/20.30): 06/5611251