

software MBASIC

Dopo aver analizzato nelle prime due puntate di questa rubrica le caratteristiche dei due tipi di file (sequenziali e random) previsti dall'MBASIC, in questa terza puntata presentiamo un programma di esempio, utilizzando tutti e due i tipi di file ed un certo numero di istruzioni dell'MBASIC che abbiamo conosciuto nelle precedenti puntate.

Diciamo subito che si tratta di un archivio di dati, quasi un "mini-database" dal momento che non vi sono solo le funzioni di input e di output, ma anche un primo abbozzo di funzione di ricerca. Evidentemente non è un programma completo di tutte le raffinatezze presenti in altri database, ma può servire di aiuto ai lettori che volessero intraprendere lo studio di tale problematica.

Un data base per archiviare i programmi di MC

Con questo programma, è il caso di dirlo, uniamo l'utile con il dilettevole, in quanto vedremo un'applicazione pratica delle nozioni fin qui apprese ed inoltre avremo già le basi (funzionanti!) di un archivio di tutti i programmi pubblicati da MCmicrocomputer a partire dal n°1.

Vedremo via via quali sono stati i ragionamenti che hanno portato a certe scelte di programmazione (ad esempio il perché della convivenza in un unico programma di due file, uno sequenziale e l'altro random), in base alle caratteristiche iniziali che il nostro archivio doveva avere.

Vediamo dunque tali caratteristiche: record formato da quattro campi per un totale di 32 byte per record. I quattro campi sono rispettivamente:

- MCS, campo indicante il numero intero della rivista, 2 byte.
- AS, campo indicante l'argomento, 26 byte.
- CS, campo indicante il codice del computer, 2 byte.
- LS, campo indicante il codice del linguaggio, 2 byte

In particolare si può vedere che, a parte il campo relativo all'argomento che è ovviamente una stringa di caratteri, i rimanenti campi conterranno una coppia di byte rappresentativi di tre quantità numeriche intere: mentre sulla prima non c'è nulla da dire, in quanto si tratta del numero della rivista, per le altre due si può già vedere

una prima scelta per snellire la fase di input di dati. Si è infatti scelto di codificare sia il computer che il linguaggio di programmazione con un valore intero, per non essere costretti, in fase di input dati, a scrivere per esteso il nome di un certo computer per tutti i record che lo prevedono: in questo

modo (e lo stesso discorso vale per i linguaggi) si è anche risparmiato moltissimo spazio del record, spazio che secondo le considerazioni delle puntate precedenti era pure difficile da stabilire con precisione.

Osservando ad esempio la routine tra le linee 800 e 860 si può vedere che se si deve impostare un codice di un nuovo computer basta semplicemente impostare il valore 0 e poi, a seguito della richiesta del nome effettivo, rispondere (e solo quella volta!) con il nome completo, al massimo di 26 caratteri.

Automaticamente questo nuovo nome viene accodato all'elenco dei nomi prece-

```

10 CLEAR 2000:CLS:PRINT:PRINT:PRINT:PRINT"inizializzazione...":PRINT:PRINT:PRINT
20 NC=30:REM numero max computer
30 NL=10:REM numero max linguaggi
40 DIM CO$(NC),LI$(NL)
50 OPEN "r",#1,"archivio",32
60 FIELD #1,2 AS MC$,26 AS A$,2 AS C$,2 AS L$
70 GET #1,1:IF CVI(MC$)=0 GOTO 180
80 PRINT"lettura vettori...":OPEN "i",#2,"vettori"
90 INPUT#2,NR%,NC%,NL%:REM n.record,n.computer,n.linguaggi
100 IF (NC%=<NC) AND (NL%=<NL) GOTO 150
110 PRINT"n.computer =";NC%
120 PRINT"n.linguaggi =";NL%
130 PRINT"correggere i valori di nc e nl"
140 CLOSE:END
150 FOR I=1 TO NC%:INPUT#2,CO$(I):NEXT:CO$(0)=STR$(NC%)
160 FOR I=1 TO NL%:INPUT#2,LI$(I):NEXT:LI$(0)=STR$(NL%)
170 CLOSE#2
180 V$(1)="n.rivista "
190 V$(2)="argomento "
200 V$(3)="computer "
210 V$(4)="linguaggio "
220 CLS:PRINT"          menu":PRINT"          =====:PRINT
230 PRINT"  1 - input dati"
240 PRINT"  2 - ricerca"
250 PRINT"  3 - correzione"
260 PRINT"  4 - output generale"
270 PRINT"  0 - fine":PRINT
280 INPUT"scelta";SC$:SC%=ABS(SC%):IF SC%>4 GOTO 280
290 ON SC%+1 GOTO 500,1000,1500,2000,2500
500 OPEN "o",2,"vettori":PRINT"salvataggio vettori..."
510 PRINT#2,NR%,CINT(VAL(CO$(0))),CINT(VAL(LI$(0)))
520 FOR I=1 TO VAL(CO$(0)):PRINT#2,CO$(I):NEXT
530 FOR I=1 TO VAL(LI$(0)):PRINT#2,LI$(I):NEXT:CLOSE:END
600 CLS:PRINTV$(1);:INPUT NZ:LSET MC%=MKI$(NZ):RETURN
700 CLS:PRINTV$(2);:INPUT AR$:LSET A%=AR$:RETURN
800 CLS:C%=VAL(CO$(0))
810 IF C%=0 GOTO 850
820 PRINT"codice computer":FOR I=1 TO C%:PRINTI,CO$(I):NEXT
830 INPUT"codice computer (0=nuovo) ";CO%:CO%=ABS(CO%):IF CO%>C% GOTO 830

```

(continua a pagina 160)

denti e mostrato quando si imposterà il record successivo.

Questo fatto di codificare i nomi dei computer e i linguaggi ha però comportato la necessità di tenere memorizzata da qualche parte la tavola di corrispondenza tra il numero ed il computer e/o il linguaggio: dato che queste informazioni stavano comodissime in due vettori, ecco che si è deciso di creare un apposito file sequenziale di appoggio: badate bene, è sequenziale e non random per il solito motivo che non si conosce a priori la lunghezza del nome del computer e del linguaggio.

Tra l'altro bisognava registrare su disco anche altre tre informazioni numeriche, che non potevano essere poste in un record del file random: innanzitutto il numero di record del file random nonché il numero di computer e di linguaggi usati nella codifica, questi ultimi due valori essendo ovviamente la dimensione dei due vettori, che nel programma si chiamano COS(.) e LIS(.). Trascurando quelle che sono le parti di "abbellimento" dell'output su video, andiamo ad analizzare il programma.

Il programma

Innanzitutto nelle righe 20 e 30 vengono poste rispettivamente a 30 e 10 le dimensioni dei due vettori di cui sopra: questo per dare un valore all'interno della successiva DIM.

Se per caso si avesse sentore che tali valori fossero troppo piccoli basterà semplicemente aumentarli nelle due linee citate.

Alle linee 50 e 60 abbiamo rispettivamente l'apertura del file "archivio" e la definizione dei suoi campi.

Alla linea successiva vediamo un utilissimo trucco programmatico. In particolare si va a leggere il primo record e se il campo MCS (cioè il numero della rivista) viene trovato a 0 si salta alla linea 180, e cioè si salta una parte di programma.

Vediamo il significato di questo trucco: all'inizio, subito dopo aver digitato il programma, il file "archivio" non esisterà ancora, così come il file "vettori".

Con l'istruzione OPEN della linea 50 inizieremo tale file che conterrà byte nulli in ogni campo di ogni suo record. Ecco che il campo MCS conterrà perciò un valore nullo, ben differente da qualsiasi valore a regime (diverso da zero). Questo fatto è dunque sfruttato per far riconoscere al programma che il file "archivio" è vuoto. Attenzione però a non mettere deliberatamente nel primo record un numero di rivista pari a 0, in quanto comporterebbe la cancellazione di tutti i record registrati quando rielaboreremo il programma in un secondo tempo.

Tutto questo perché quando il file è vuoto sono vuoti anche i vettori e sono nulle le tre quantità numeriche già viste prima: in definitiva il file "vettori", come già visto, non esiste nemmeno.

Invece nella linea 80 troviamo una OPEN relativa a tale file: va da sé che, se non

(segue da pagina 159)

```
840 IF COZ<>0 GOTO 860
850 CZ=LX+1:CO$(0)=STR$(CZ):PRINTV$(3);:INPUT CO$(CZ):COZ=CZ
860 LSET C$=MKI$(COZ):RETURN
900 CLS:LX=VAL(LI$(0))
910 IF LX=0 GOTO 950
920 PRINT"codice linguaggio":FOR I=1 TO LX:PRINTI,LI$(I):NEXT
930 INPUT"codice linguaggio (0=nuovo) ";LIX:LIX=ABS(LIX):IF LIX)LX GOTO 930
940 IF LIX<>0 GOTO 960
950 LX=LX+1:LI$(0)=STR$(LX):PRINTV$(4);:INPUT LI$(LX):LIX=LX
960 LSET L$=MKI$(LIX):RETURN
1000 GOSUB 600:GOSUB 700:GOSUB 800:GOSUB 900
1010 CLS:PRINT"numero record";NRX
1020 PRINT"1 - ";V$(1);NZ
1030 PRINT"2 - ";V$(2);AR$
1040 PRINT"3 - ";V$(3);CO$(COZ)
1050 PRINT"4 - ";V$(4);LI$(LIX)
1060 INPUT"ok (s/n)";R$:IF R$="s" GOTO 1090
1070 IF R$<"n" GOTO 1060
1080 GOSUB 1200
1090 NRX=NRX+1:PUT#1,NRX
1100 INPUT"ancora record (s/n)";R$:IF R$="s" GOTO 1000
1110 IF R$<"n" THEN 1100
1120 GOTO 220
1200 INPUT"n.campo errato";CEZ:CEZ=ABS(CEZ):IF (CEZ=0) OR (CEZ)4 GOTO 1200
1210 ON CEZ GOSUB 600,700,800,900
1220 RETURN
1300 PRINT:PRINT"premi un tasto per continuare"
1310 R$=INKEY$:IF R$="" THEN GOTO 1310 ELSE RETURN
1400 CLS:RIZ=0:FOR J=1 TO 4:PRINTV$(J),:NEXT:PRINT:RETURN
1500 CLS:PRINT" ricerca"
1510 FOR J=1 TO 4:PRINT" ";J;" - per ";V$(J):NEXT
1520 PRINT" 0 - ritorno al menu"
1530 PRINT
1540 INPUT"scelta";SZ:SZ=ABS(SZ):IF SZ>4 GOTO 1540
1550 IF SZ=0 GOTO 220
1552 IF SZ=3 THEN PRINT"codice computer":FOR I=1 TO VAL(CO$(0)):PRINTI,CO$(I):NEXT
1554 IF SZ=4 THEN PRINT"codice linguaggio":FOR I=1 TO VAL(LI$(0)):PRINTI,LI$(I):NEXT
1560 PRINTV$(SZ);:INPUT S$
1570 GOSUB 1400
1580 FOR I=1 TO NRX:GET#1,I:ON SZ GOTO 1590,1600,1610,1620
1590 IF CUI(MC$)=CINT(VAL(S$)) THEN GOTO 1630 ELSE GOTO 1650
1600 IF INSTR(A$,S$) THEN GOTO 1630 ELSE GOTO 1650
1610 IF CUI(C$)=CINT(VAL(S$)) THEN GOTO 1630 ELSE GOTO 1650
1620 IF CUI(L$)<>CINT(VAL(S$)) GOTO 1650
1630 PRINTCUI(MC$),A$,CO$(CUI(C$)),LI$(CUI(L$))
1640 RIZ=RIZ+1:IF RIZ=20 THEN GOSUB 1300:GOSUB 1400
1650 NEXT:GOSUB 1300:GOTO 220
2000 CLS:PRINT" correzione"
2010 INPUT"n.record";REZ:GET#1,REZ
2020 PRINT"1 - ";V$(1);CUI(MC$)
2030 PRINT"2 - ";V$(2);A$
2040 PRINT"3 - ";V$(3);CO$(CUI(C$))
2050 PRINT"4 - ";V$(4);LI$(CUI(L$))
2060 GOSUB 1200:INPUT"ancora correzioni (s/n)";R$:IF R$="s" CLS:GOTO 2020
2070 PUT#1,REZ
2080 GOSUB 1300:GOTO 220
2500 CLS
2510 GOSUB 1400
2520 FOR I=1 TO NRX:GET#1,I:PRINTCUI(MC$),A$,LEFT$(CO$(CUI(C$)),15),LI$(CUI(L$))
2530 RIZ=RIZ+1:IF RIZ=20 THEN GOSUB 1300:GOSUB 1400
2540 NEXT:GOSUB 1300:GOTO 220
```

Listing del programma che utilizza un file random per l'archivio ed uno sequenziale per i vettori ausiliari.

vi fosse il test sul campo MC\$, il programma incontrerebbe tale istruzione di apertura di un file inesistente, fatto che comporta il blocco del programma stesso per errore. Invece a regime succede che il valore MC\$ del primo record sarà diverso da 0 ed allora si andrà a leggere il file "vettori". In particolare si leggeranno il numero di record, le dimensioni dei due vettori ed i due vettori stessi senza alcuno spreco di spazio sul dischetto. È inoltre importante chiudere SOLO il file "vettori" subito dopo aver effettuato questa lettura: questo perché si presuppone che la successiva operazione su tale file sarà una scrittura, per il salvataggio dei valori all'atto dell'uscita dal programma.

Successivamente troviamo l'output del "menu" con quattro opzioni più quella di uscita: già a questo punto si possono effettuare le prime migliorie aggiungendo ulteriori scelte.

In particolare scegliendo l'opzione 0 si va alla routine 500 che prevede il salvataggio delle tre famose quantità e dei due altrettanto famosi vettori, dopodiché si vanno a chiudere tutti e due i file e si esce dal programma. Ovviamente per effettuare il salvataggio abbiamo aperto il file "vettori" in output.

L'input dei dati

La fase di input dei vari record consiste

in quattro piccole subroutine relative ad ognuno dei quattro campi di ogni record: non presentano alcuna difficoltà di interpretazione, mentre vi è da notare l'utilizzazione dell'istruzione LSET per la memorizzazione dei vari campi.

La ricerca

È questa una parte del programma molto interessante ed utile in quanto consente di visualizzare, tra i vari record, tutti quelli che soddisfino una condizione a scelta tra:

- il "numero della rivista"
- l'"argomento"
- il "computer" sul quale gira il programma
- il "linguaggio" di programmazione

In particolare per quanto riguarda la ricerca per "argomento", si è sfruttata la potentissima INSTR (substring, stringa): cioè si può indicare come argomento di ricerca ad esempio "sort" e subito avremo visualizzati tutti quei record aventi "sort" come sottostringa del campo "argomento". Ecco che appariranno ad esempio record relativi all'"heapsort", allo "Shell - Metzer sort" come pure un eventuale record avente come argomento il "sorteggio".

Impostando come argomento la lettera "a", ad esempio, è facile capire che verranno mostrati tutti i record aventi nella stringa "argomento" almeno una "a".

La correzione

Per quest'altra fase molto importante, basta dire che si deve inserire il numero del record interessato e subito vi si accederà, secondo lo stile dei file random: effettuata la correzione, il record verrà riscritto con una solita PUT.

L'output generale e le conclusioni

Per ottenere una o più videate di tutti i record basta eseguire un loop di visualizzazione di un record il cui indice va da 1 al numero massimo di record presenti nell'archivio.

Come si è visto, il tutto è stato realizzato molto semplicemente: ripetiamo che non era nostra intenzione creare un ennesimo data - base (che per giunta è alquanto lento, dal momento che è scritto in Basic).

Nella prossima puntata analizzeremo alcune particolarità dell'MBASIC, riguardanti l'istruzione PRINT, che forse non tutti conoscono: anche se esistono nell'interprete originario della Microsoft, in parecchie versioni adattate ai vari personal alcune caratteristiche si sono "perse per strada" oppure, e ciò è ancora peggio, sono presenti, ma "dimenticate" dai manuali.

MC

PROGRAMMA 2000

il bimestrale di 20 programmi su cassetta

NOVITA' PROGRAMMA 2000

Per tutti voi un apposito centro per i vostri problemi telefonate al: "Software Solution Center" 06/5926442
In regalo il catalogo "Hard-Soft"
PROGRAMMA 2000 anche per Commodore C16/4 PLUS
I volumi arretrati a lit. 20.000

SONO USCITI

- P 2000 C 16/4 PLUS vol 1°**
TOTOCALCIO: pronostici e colonne fatti con il computer
SIMULAVOLO: il formidabile completo simulatore di volo
BAR-CHART: la creazione di grafici (barre, torte, linee)
MAT-GRAF: crea ed analizza i grafici per ogni funzione
LA CITTA' DEL FUTURO: nuova adventure game in italiano
... e poi fino a 20 ...
- P 2000 COMMODORE 64 vol 5°**
BASCOMP: compilatore BASIC P 2000! 15 volte più veloce
P2000 1,2,3: integra foglio elettronico, grafica e D.B.
MUSIC: finalmente puoi comporre musica con il computer
SCACCHI: per giocare seriamente con il vostro computer
REPORT: gestisci 132 colonne sulla tua stampante da 80
... e poi fino a 20 ...
- P 2000 TEXAS 1199/4A vol 3°**
TIWORD: gestisce i testi con giustificazione in stampa
MONITOR: legge in ASSEMBLER tutta la memoria RAM e ROM
MAIL: archivia stampa: etichette, lettere personalizzate
GAMES: un set di cinque video-games in BASIC sul TEXAS
MAGAZZINO: poter gestire ora: scorte, carichi e scarichi
... e poi fino a 20 ...



DA' PIU' FORZA AL TUO COMPUTER

in ogni volume troverai il MANUALE ITALIANO e la cassetta di 20 programmi per il TUO computer un volume 20000 lire

1.000 LIRE A PROGRAMMA

Tutti i programmi sono listabili !!

Spedire in busta chiusa a: **PROGRAMMA 2000**
loc. Le Macere - 00060 Formello (Roma) - tel. 06/5926442

Nome e Cognome

Indirizzo

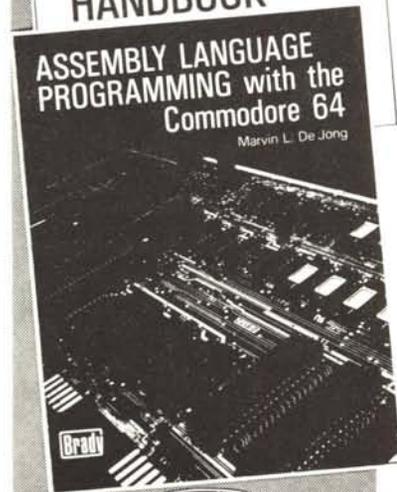
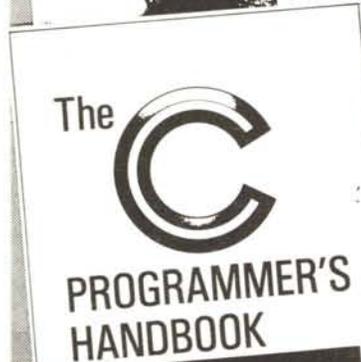
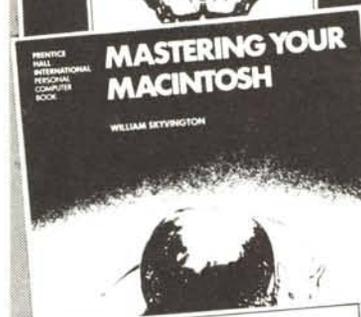
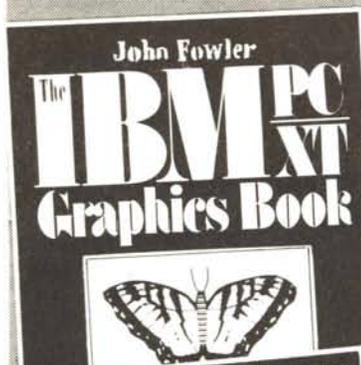
Città CAP

Computer Vol N. copie ...

allego pagamento pagherò alla consegna

New From

Prentice-Hall International



IBM PERSONAL COMPUTER

Assembly Language for the IBM Personal Computer

David Bradley

Beginning with the fundamentals of computer operation, this book completely covers the assembly language instructions of the Intel 8088 and their use in the IBM PC.

416 pages 13-049171-3 Paper \$25.95

Programmer's Guide to MS-DOS

Dennis Jump

For novice or intermediate programmers, traditional character device I/O, non-device Functional File Management, extended function groups and extended memory management are fully explained.

244 pages 8359-5655-5 Paper \$20.75

Understanding and Using dBASE II Revised and Enlarged Edition

R. Krumm

Concise and easy to use, this software guide shows new users how dBASE II can work for them without going into unnecessary detail about its complexities.

308 pages 89303-916-0 Paper \$25.95

IBM PC/XT Graphics Book

John Fowler

Custom-designed for the IBM PC/XT, this tutorial addresses the practical questions of how graphics work and how to use them, and is full of BASIC programming examples.

Paperback c. 300 pages 13-448408-8 \$19.45
Book and disk 13-448416-9 \$38.95 (non-returnable)

APPLE MACINTOSH

Mastering Your Macintosh

A User's Guide to Apple's Macintosh Computer

William Skyvington

300 screen displays, practical business programs, and full coverage of the Macintosh's "mouse", "window" and software packages make this book ideal for new or potential users of Apple's new computer.

257 pages 13-559527-4 Paper \$13.95

Programming Tips and Techniques for the Apple II and Apple IIe

John Campbell

A guide to the advanced functions of the Apple II and IIe which demonstrates how to solve programming problems.

403 pages 89303-273-5 Paper \$25.95

Book and disk 89303-776-1 \$64.95

Disk only 89303-782-6 \$39.00 (non-returnable)

COMMODORE 64

Assembly Language Programming with the Commodore 64

Marvin L. de Jong

All the important assembly language topics are addressed including data transfer, logic operations, branches and loops, and programming the 6581 sound interface device.

296 pages 89303-319-7 Paper \$14.95

Machine Language for the Commodore 64 and other Commodore Computers

Jim Butterfield

Programmers of all levels are introduced to the principles of Commodore machine language—what it is, how it works, and how to program with it.

326 pages 89303-652-8 Paper \$12.95

PROGRAMMING LANGUAGES

The C Programmer's Handbook

Thom Hogan

Revealed in this definitive desktop reference for C language users is a wealth of information about C compilers—complete with documentation, examples, restrictions and defaults.

262 pages 89303-365-0 Paper \$16.95

FORTH Programming

Steven Vickers

An introductory guide which explains how to use FORTH commands for decision making, repeating, performing, arithmetic, sound, graphics and more.

185 pages 13-326356-8 Paper \$19.45

Prices are correct at the time of going to press, but may be subject to change.



Questi titoli e una vasta scelta di libri in computer science pubblicati dalla Prentice-Hall sono reperibili presso le più importanti librerie internazionali.

Permanentemente in stock presso.

Per ulteriori informazioni, richieste di inserimento nella mailing list ecc., contattare il nostro agente in Italia: Flavio Marcello, Via Vincenza 27/B, 35100 Padova. Tel: (049) 25541.

FIRENZE
LA NOUVA ITALIA
BIBLIOGRAFICA
Via E. Codignola
Scandicci
CAT 50018 Firenze
Tel: 27 98 11
MARZOCCO
Via de Martelli, 22/R
Tel: 26 52 51

GENOVA
CENFOR S R L
Piazza Sabina 2
Tel: 20 38 76
PALERMO
S. F. FLACCOVIO
Via Ruggero Settimo, 37
Tel: 33 42 49
PADOVA
CORTINA
Via F. Marzolo, 2
Tel: 65 08 59

LIBRERIA PROGETTO
AL PORTEO
Via Marzolo, 28
Tel: 66 55 85
BARI
GIUS LATERZA & FIGLI
Via Sparano, 134
Tel: 21 08 29
ROMA
ANGLO-AMERICAN
Via della Vite, 27
Tel: 67 83 890

INGEGNERIA 2000
Via della Polverera, 15
Tel: 47 44 169
GABI
Via Gabi 30
Tel: 77 43 03
VERONA
CORTINA
Via Carlo Cattaneo, 8
Tel: 59 41 77

MILANO
C.L.U.P.
Piazza L. de Vinci, 32
Tel: 23 05 45
HOEPLI
Via U. Hoepli, 5
Tel: 86 54 46
IPSOA
Largo Augusto, 8
Tel: 82 47 6

TORINO
LIBRERIA EDITRICE
UNIVST LEVROTTO
& BELLA
Corso Vittorio
Emanul N28
Tel: 83 25 35
BOLOGNA
D.E.A.
Via Belle Arti 48C
Tel: 23 61 00