

Il software per il Commodore 64 che vi proponiamo questo mese comprende due distinte routine (o applicazioni) riguardanti la gestione delle directory e la gestione in L.M. dei blocchi di istruzioni DATA in Basic. Al ritorno dalle vacanze saranno molti coloro che non vedranno l'ora di ricominciare a programmare il loro caro, vecchio e fido home computer. Una buona occasione è rappresentata proprio dal software in questione che è comunque disponibile secondo le solite modalità anche su supporto magnetico

## Diskpack: gestione della directory

di Roberto Morassi - Pistoia

Diskpack è una raccolta di brevi utility-disco un po' insolite, che possono servire quando si ha a che fare con delle directory variamente alterate o manipolate.

### (1) Controllo lunghezza file

I byte 28-29 di ciascun file-header riportano la lunghezza del file in formato basso/alto byte, ed è questo il valore che si legge nel listato della directory. In certi dischetti, tuttavia, capita di trovare dei valori alterati (ad esempio, con tutti i file a «lunghezza zero»); la stessa BAM (Block Availability Map) potrebbe essere stata manipolata in modo tale che il

È disponibile, presso la redazione, il disco con i programmi pubblicati in questa rubrica. Le istruzioni per l'acquisto e l'elenco degli altri programmi disponibili sono a pag. 295.

### Diskpack

```

10 REM *** DISKPACK - BY R.MORASSI ***
20 :
30 DIM X,A,Fs,Zs,Ps,Cs,B,Bs,As,EN,EMS,ET,ES,BF,T,S,N,S1,S2,TT,SS,D,AA,I$
40 DIM TF,DF,TK,SK,NS,L1,E,A1$,B1$,L2,BB
50 DIM Ms(6),Fs(4)
60 FORX=1TO80:READA:Fs=Fs+CHR$(A):NEXT
70 FORX=1TO6:READMs(X):NEXT
80 Zs=CHR$(0):Ps=CHR$(160)
90 Cs="":FORX=1TO17:Cs=Cs+Ps:NEXT
100 Fs(1)="SEQ":Fs(2)="PRG":Fs(3)="USR":Fs(4)="REL"
110 :
120 REM ** MENU PRINCIPALE **
130 :
140 SYS65409:PRINT"(CLR) (WHT) ***** DISKPACK - R.MORASSI *****"
150 PRINT:PRINT"(CYN) 1) CONTROLLA LUNGHEZZA FILES
160 PRINT" 2) CAMBIA NOME DISCO E NORMALIZZA ID
170 PRINT" 3) AZZERA LA B.A.M.
180 PRINT" 4) PROTEGGE/SPROTEGGE DA SCRITTURA
190 PRINT" 5) CAMBIA ADDRESS DI FILES/PRG
200 PRINT" 6) DISPLAY DIRECTORY
210 PRINT" 7) FINE LAVORO":PRINT:PRINT
220 X=3:GOSUB2330:IFAs<"1"ORAs>"7"THEN220
230 ONVAL(As)GOSUB280,960,1200,1400,2020,2420,2770
240 GOTO140
250 :
260 REM ** CONTROLLA LUNGHEZZA FILES **
270 :
280 PRINT"(CLR) (WHT) ** CONTROLLA LUNGHEZZA FILES **:PRINT
290 X=1:GOSUB2330:IFAs="0"THENRETURN
300 OPEN15,8,15,"I":GOSUB2550:GOSUB770:OPEN2,8,2,"#0":GOSUB900
310 PRINT:PRINT"(DOWN) (YEL)DISCO (RVS) N$
320 PRINT:PRINT"(CYN) FILE NOME"TAB(23)"TIPO BLKS BLKS
330 PRINT" N."TAB(29)"DIR VERI":PRINT"(WHT)"
340 T=18:S=1:N=0:S1=0:S2=0
350 PRINT#15,"U1:"2:0:T:S:GOSUB2550:TT-T:SS-S
360 GET#2,As,Bs:T=ASC(As+Zs):S=ASC(Bs+Zs)
370 D=0:AA=0:FORX=1TO8:POKE781,2:SYS65478:I$=""
380 GETAs:TF=ASC(As+Zs):DF=TFAND7
390 A=TFAND64:IFA=64THENI$="<"
400 A=TFAND128:IFA=0THENI$=""
410 GETAs,Bs:TK=ASC(As+Zs):SK=ASC(Bs+Zs)
420 IFTF=0ANDTK=0ANDSK=0THENND=1:X=8:N=N+1:GOTO570
430 Ns="":FORA=1TO16
440 GETAs:IFAs<>P$THENNs=Ns+As
450 NEXT
460 FORA=1TO9:GETAs:NEXT
470 GETAs,Bs:IFDF>0THENL1=ASC(As+Zs)+256*ASC(Bs+Zs):S1=S1+L1
480 IFX<8THENGETAs,As
490 SYS65484:IFDF=0THEN570
500 N=N+1:PRINTTAB(1)N;TAB(6)Ns;TAB(24)Fs(DP)I$;TAB(28)L1;TAB(33):
510 GOSUB790:IFE>1THEN690
520 BB=0:IFL2>1024THENBB=1:L2=0
530 PRINTL2
540 IFL2=0ANDBB=0THENPRINT"(YEL) LINK ILLEGALE ! (WHT) "
550 IFL2=0ANDBB=1THENPRINT"(YEL) FILE FITTIZIO (TRACCIA 18) ! (WHT) "
560 S2=S2+L2:IFL2<>L1ANDL2>0THENGOSUB850
570 NEXT
580 IFAATHENPRINT#15,"U2:"2:0:TT:SS:GOSUB2550:IFEN=26THEN730
590 IFDOR(T=0)THEN730
600 GOTO350
610 PRINT:PRINT"(YEL) BLOCCHI OCCUPATI (DIRECTORY):"S1
620 PRINT" BLOCCHI OCCUPATI (VERI): "S2:PRINT
630 PRINT" BLOCCHI LIBERI (DIRECTORY): "664-S1
640 PRINT" BLOCCHI LIBERI (VERI): "664-S2
650 PRINT" BLOCCHI LIBERI (B.A.M.): "BF:PRINT
660 IFBF=(664-S2)THEN690
670 PRINT
680 X=4:GOSUB2330:IFAs="S"THEN750
690 CLOSE15:PRINT
700 X=5:GOSUB2330:IFAs="B"THEN280
710 RETURN
720 :
730 CLOSE2:SYS65484:GOTO610
740 :
750 PRINT#15,"V":GOSUB2550:PRINT#15,"I":GOSUB900:GOTO650
760 :
770 OPEN3,8,3,"#2":PRINT#15,"B-P"3;0:PRINT#3,Fs:CLOSE3:RETURN
780 :
790 PRINT#15,"U3":CHR$(TK)CHR$(SK):GOSUB2550
800 PRINT#15,"M-R"CHR$(1)CHR$(0):GET#15,As:E=ASC(As+Zs)
810 IFE>1THENPRINT:PRINT"INTERRUZIONE; ERRORE DISCO N."E+18:CL0SE2:RETURN
820 :
830 PRINT#15,"M-R"CHR$(20)CHR$(0)CHR$(2):GET#15,A1$,B1$
840 L2=ASC(A1$+Zs)+256*ASC(B1$+Zs):RETURN
850 PRINT"(CYN) AGGIORNO LUNGHEZZA ? (S/N)";:POKE198,0:WAIT198,1:GETAs
860 IFAs<>"S"THENPRINT"(WHT)":RETURN
870 PRINT#15,"M-W"CHR$(30+32*(X-1))CHR$(3)CHR$(2)A1$B1$:
880 AA=1:PRINT"-OK! (WHT)":RETURN
890 :
900 PRINT#15,"M-R"CHR$(144)CHR$(7)CHR$(24):INPUT#15,Ns
910 PRINT#15,"M-R"CHR$(250)CHR$(2)CHR$(3):GET#15,As,Bs,B$
920 BF=ASC(As+Zs)+256*ASC(Bs+Zs):RETURN
930 :
940 REM ** CAMBIA NOME - AGGIORNA ID **
950 :
960 SYS65409:POKE53280,9:PRINT"(CLR) (SWLC) (LBLU)"TAB(30)"ID VERO:
970 PRINT" | | | | |
980 PRINT"(DOWN) /OME DEL DISCO ID 2#
990 PRINT
1000 X=1:GOSUB2330:IFAs="0"THENRETURN
1010 OPEN15,8,15,"I":OPEN2,8,2,"#":GOSUB2550:GOSUB1120
1020 Ns="":INPUT"(DOWN) /UOVO NOME":Ns:IFNs=""THEN1080

```

(continua a pag. 292)

numero dei blocchi effettivamente allocati non corrisponda né alla somma delle lunghezze dei file, né al valore risultante dalla BAM (cioè 664 meno i «blocks free»). Questa routine effettua

una scansione di tutti i file, tramite il «link», all'inizio di ciascun blocco, così da ricavare il vero numero di blocchi occupati dai file. La scansione è molto veloce, essendo gestita da una sub-

routine in linguaggio macchina (sembrato) che viene caricata in memoria dalla RAM del drive e messa in esecuzione con un comando di diretto (User command). Qualto

(segue da pag. 291)

```

1030 IFLEN(N$)>16 THEN POKE 781,8:SYS59903:PRINT"(UP)(UP)(UP)":GOTO1020
1040 AS=LEFT$(NS+C$,18)+IS+PS+"2A"+PS+PS
1050 PRINT#15;"M-R"CHR$(1)CHR$(1)CHR$(65)
1060 PRINT#15;"B-P"2;144:PRINT#2;AS$
1070 PRINT#15;"U2";2;0;18;0;GOSUB2550:GOSUB1120
1080 CLOSE2:CLOSE15:PRINT
1090 X=5:GOSUB2330:IFAS="S" THEN 900
1100 RETURN
1110 :
1120 PRINT#15;"U1";2;0;18;0;PRINT#15;"B-P"2;144
1130 FORX=0TO24:GET#2,AS:POKE1024+X,ASC(AS+Z$):NEXT
1140 PRINT#15;"M-R"CHR$(18)CHR$(0)CHR$(2):GET#15,AS,BS:IS=AS+BS
1150 POKE1062,ASC(AS+Z$):POKE1063,ASC(BS+Z$):RETURN
1160 :
1170 :
1180 REM ** AZZERA LA B.A.M. **
1190 :
1200 PRINT"(CLR) *** AZZERA LA B.A.M. ***
1210 PRINT:PRINT"(CYN) ATTENZIONE: QUESTA UTILITY AZZERA
1220 PRINT" LA B.A.M. DEL DISCO (0 BLOCKS FREE)
1230 PRINT" E IMPEDISCE LE OPERAZIONI DI SCRITTURA
1240 PRINT" SUL DISCO (DISK FULL), A MENO CHE LA
1250 PRINT" B.A.M. NON VENGA RIAGGIORNATA CON
1260 PRINT" UN "VALIDATE" (YEL)":PRINT
1270 PRINT:X=1:GOSUB2330:IFAS="0" THEN RETURN
1280 OPEN15,8,15,"I":GOSUB2550
1290 OPEN2,8,2,"#":PRINT#15;"U1";2;0;18;0
1300 PRINT#15;"B-P"2;3;FORX=1TO141
1310 PRINT#2,CHR$(0):NEXT:GOSUB2550
1320 PRINT#15;"U2";2;0;18;0;GOSUB2550
1330 IFEN=26 THEN 1350
1340 PRINT:PRINT" B.A.M. AZZERATA - OK !"
1350 PRINT#15;"I":CLOSE2:CLOSE15:PRINT:X=5:GOSUB2330:IFAS="B" THEN 1200
1360 RETURN
1370 :
1380 REM ** PROTEGGE/SPROTEGGE **
1390 :
1400 PRINT"(CLR) ** PROTEGGE/SPROTEGGE DA SCRITTURA **
1410 PRINT:PRINT"(CYN) 1) PROTEGGE UN FILE
1420 PRINT"(CYN) 2) SPROTEGGE UN FILE
1430 PRINT"(CYN) 3) PROTEGGE TUTTI I FILES
1440 PRINT"(CYN) 4) SPROTEGGE TUTTI I FILES
1450 PRINT"(CYN) 5) PROTEGGE DISCO
1460 PRINT"(CYN) 6) SPROTEGGE DISCO
1470 PRINT"(CYN) 7) DISPLAY DIRECTORY
1480 PRINT"(YEL) 0) TORNA AL MENU PRINCIPALE":PRINT
1490 X=3:GOSUB2330:IFAS<"0"ORAS>"7" THEN 1490
1500 IFAS="0" THEN RETURN
1510 ONVAL(AS)GOSUB1540,1540,1700,1700,1870,1870,2420
1520 GOTO1400
1530 :
1540 D=64+64*(AS="2")
1550 X=1:GOSUB2330:IFAS="0" THEN RETURN
1560 PRINT:PRINT:INPUT"NOME DEL FILE ";NS
1570 OPEN15,8,15,"I":OPEN3,8,3,NS:CLOSE3:GOSUB2550
1580 IFEN=62 THEN PRINT"(DOWN) QUESTO FILE NON C'E !":CLOSE15:X=2:GOSUB2330:RETUR
N
1590 PRINT#15;"M-R"CHR$(144)CHR$(2)CHR$(5)
1600 GET#15,AS,BS,BS,BS:AS=ASC(AS+Z$):B=ASC(BS+Z$)
1610 OPEN2,8,2,"#":PRINT#15;"U1";2;0;18;A
1620 PRINT#15;"B-P"2;B:GET#2,AS:E=ASC(AS+Z$):E=(EAND191)ORD
1630 PRINT#15;"B-P"2;B:PRINT#2,CHR$(E):PRINT#15;"U2";2;0;18;A:GOSUB2550
1640 IFEN=26 THEN 1660
1650 PRINT:PRINT" ESEGUITO - OK"
1660 CLOSE2:CLOSE15:PRINT
1670 X=6:GOSUB2330:IFAS="S" THEN 1560
1680 RETURN
1690 :
1700 E--(AS="3")
1710 PRINT:X=1:GOSUB2330:IFAS="0" THEN RETURN
1720 OPEN15,8,15,"I":GOSUB2550:OPEN2,8,2,"#":GOSUB2550
1730 T=18:S=1
1740 PRINT#15;"U1";2;0;T;S
1750 PRINT#15;"B-P"2;0;GET#2,AS,BS:TT=ASC(AS+Z$):SS=ASC(BS+Z$)
1760 FORX=2TO226STEP32
1770 PRINT#15;"B-P"2;X:GET#2,AS:N=ASC(AS+Z$):D=N AND64
1780 IF(D)AND(E=0)AND(N<192) THEN PRINT#15;"B-P"2;X:PRINT#2,CHR$(NAND191)::GOTO18
00
1790 IFNAND(E=1) THEN PRINT#15;"B-P"2;X:PRINT#2,CHR$(N OR64):
1800 NEXT:PRINT#15;"U2";2;0;T;S:GOSUB2550
1810 IFEN=26 THEN 1840
1820 IF1 THEN IT="I":S="S":GOTO1740
1830 PRINT:PRINT" ESEGUITO - OK"
1840 CLOSE2:CLOSE15:X=5:GOSUB2330:IFAS="S" THEN 1710
1850 RETURN
1860 :
1870 D=65-(AS="5")

```

```

1930 PRINT#15;"U2";2;0;18;0;GOSUB2550
1940 IFEN=26 THEN 1960
1950 PRINTTAB(92)"ESEGUITO - OK !":PRINT
1960 CLOSE2:CLOSE15:PRINT
1970 X=5:GOSUB2330:IFAS="S" THEN 1880
1980 RETURN
1990 :
2000 REM ** CAMBIA ADDRESS **
2010 :
2020 PRINT"(CLR)(WHT) *** CAMBIA ADDRESS DI FILES/PRG ***
2030 PRINT"(DOWN)(DOWN)(CYN) (1) CAMBIA LOAD ADDRESS
2040 PRINT"(DOWN) (2) LEGGE DIRECTORY
2050 PRINT"(DOWN) (0) TORNA AL MENU PRINCIPALE
2060 PRINT:X=2:GOSUB2330:IFAS<"0"ORAS>"2" THEN 2060
2070 IFAS="0" THEN RETURN
2080 ONVAL(AS)GOSUB2110,2420
2090 GOTO2020
2100 :
2110 PRINT"(CLR)"
2120 X=1:GOSUB2330:IFAS="0" THEN RETURN
2130 OPEN15,8,15,"I":GOSUB2550:PRINT"(WHT)(DOWN)(DOWN) NOME DEL FILE DI CUI VUOI
CAMBIARE
2140 PRINT" L'ADDRESS":POKE631,158:POKE198,1:INPUTNS
2150 OPEN2,8,2,NS+"P,R":GOSUB2550
2160 IFEN=62 THEN PRINT"(DOWN)(DOWN) QUESTO FILE NON C'E !":CLOSE15:GOTO2280
2170 IFEN=64 THEN PRINT"(DOWN)(DOWN) NON E' UN FILE-PRG !":CLOSE15:GOTO2280
2180 PRINT#15;"M-R"CHR$(24)CHR$(0)CHR$(2):GET#15,AS,BS
2190 A=ASC(AS+Z$):B=ASC(BS+Z$)
2200 GET#2,AS,BS:PRINT"(CYN)(DOWN) ADDRESS ATTUALE:(YEL)"ASC(AS+Z$)+256*ASC(BS+Z
$)
2210 INPUT"(WHT)(DOWN)(DOWN) NUOVO ADDRESS":N:D=INT(N/256):E=N-256*D
2220 CLOSE2:OPEN3,8,3,"#":GOSUB2550
2230 PRINT#15;"U1";3;0;A;B:GOSUB2550
2240 GET#3,AS,AS:PRINT#3,CHR$(E)CHR$(D)
2250 PRINT#15;"U2";3;0;A;B:GOSUB2550
2260 IFEN=26 THEN 2280
2270 PRINT:PRINTSPC(12)"(RVS) ESEGUITO - OK ! (DOWN)
2280 PRINT:X=2:GOSUB2330
2290 CLOSE2:CLOSE3:CLOSE15:RETURN
2300 :
2310 REM ** MESSAGGI **
2320 :
2330 B=20-LEN(M$(X))/2
2340 IFX=1 THEN PRINT"(YEL) SE VUOI TORNARE AL MENU, PREMI {0} !":PRINT
2350 BS=CHR$(18+128*(PEEK(162)/16)AND1))
2360 PRINTTAB(B)BS"(WHT)"M$(X)"(UP)"
2370 GETAS:IFAS="0" THEN 2350
2380 PRINT"(OFF)":RETURN
2390 :
2400 REM ** DIRECTORY **
2410 :
2420 PRINT"(CLR)(RVS)SHIFT(OFF) FERMA LO SCROLLING
2430 PRINT"(RVS)COMMODORE(OFF) INTERRUPE IL LIST
2440 PRINT"(DOWN)(CYN)":OPEN15,8,15,"I":GOSUB2550:OPEN1,8,0,"$":GOSUB2550
2450 POKE781,1:SYS65478
2460 GETNS,NS
2470 GETNS,NS,AS,BS,BS,BS:Y=ST:IFX=0 THEN PRINTASC(AS+Z$)+256*ASC(BS+Z$):
2480 IF(ST)OR(PEEK(653)=2) THEN 2510
2490 GETNS:IFNS="0" THEN PRINT:WAIT653,1,1:GOTO2470
2500 PRINTNS::GOTO2490
2510 PRINT:CLOSE1:CLOSE15:SYS65484:X=2:GOSUB2330:RETURN
2520 :
2530 REM ** ERRORI DISCO **
2540 :
2550 INPUT#15,EN,EM$,ET,ES:IFEN=20 THEN 2090:IFEN=62 THEN 2090:IFEN=64 THEN 2090
2560 IFEN=26 THEN PRINT:PRINT"NON ESEGUITIBILE: TACCA CHIUSA!":RETURN
2570 PRINT:PRINT"ERRORE DISCO:":PRINTEN,EM$,ET,ES:STOP
2580 :
2590 REM ** DATA **
2600 :
2610 DATA 169,255,133,020,133,021,174,003
2620 DATA 002,172,004,002,230,020,208,002
2630 DATA 230,021,134,059,132,060,138,240
2640 DATA 018,201,018,208,004,169,004,208
2650 DATA 008,201,036,144,007,169,000,133
2660 DATA 020,133,021,096,032,075,242,197
2670 DATA 060,144,242,165,059,133,008,165
2680 DATA 060,133,009,169,128,133,001,165
2690 DATA 001,048,252,201,002,176,228,174
2700 DATA 000,004,172,001,004,076,012,005
2710 :
2720 DATA "INSERISCI IL DISCO E PREMI UN TASTO","PREMI UN TASTO","SCEGLI !"
2730 DATA "ESEGUO IL VALIDATE ? (S/N)"
2740 DATA "RIPETO CON UN ALTRO DISCO ? (S/N)"
2750 DATA "RIPETO CON UN ALTRO FILE ? (S/N)"
2760 :
2770 STOP
2780 :

```

```

..0500 a9 ff lda #fff ; pone #255
..0502 85 14 sta $14 ; in $14
..0504 85 15 sta $15 ; e $15
..0506 ae 03 02 idx $0203 ; carica in X il n. di traccia
..0509 ac 04 02 ldy $0204 ; e in Y il n. di settore
..050c e6 14 inc $14 ; aumenta puntatore
..050e d0 02 bne $0512 ; in $14
..0510 e6 15 inc $15 ; e in $15 se overflow
..0512 86 3b stx $3b ; copia n. traccia in $3B
..0514 84 3c sty $3c ; e n. settore in $3C
..0516 8a txa ; controlla: n. traccia = 0 ?
..0517 f0 12 beq $052b ; si': esce dalla routine
..0519 c9 12 cmp #12 ; no: e' traccia 18 ?
..051b d0 04 bne $0521 ; se no, salta a $0521
..051d a9 04 lda #04 ; se si', carica #04.
..051f d0 08 bne $0529 ; lo conserva in $15 ed esce
..0521 c9 24 cmp #24 ; traccia >= 35 ?
..0523 90 07 bcc $052c ; no: salta a $052c
..0525 a9 00 lda #00 ; si' (illegal link): pone zero
..0527 85 14 sta $14 ; in $14
..0529 85 15 sta $15 ; e $15
..052b 60 rts ; ed esce dalla routine
..052c 20 4b f2 jsr $f24b ; calcola il max. settore ammesso:
..052f c5 3c cmp #3c ; se il settore richiesto
..0531 90 f2 bcc $0525 ; e' > del massimo, salta a $0525
..0533 a5 3b lda $3b ; altrimenti, scrive il n. di traccia
..0535 85 08 sta $08 ; in $08
..0537 a5 3c lda $3c ; e il n. di settore
..0539 85 09 sta $09 ; in $09
..053b a9 80 lda #80 ; mette il codice di comando (#128)
..053d 85 01 sta $01 ; in $01 (job queue buffer 1)
..053f a5 01 lda $01 ; attende finche' il settore
..0541 30 fc bmi $053f ; non e' stato copiato nel buffer 1
..0543 c9 02 cmp #02 ; controlla codice errore
..0545 b0 e4 bcs $052b ; se >= 2, esce dalla routine
..0547 ae 00 04 ldx $0400 ; altrimenti, carica nuova traccia in X
..054a ac 01 04 ldy $0401 ; e nuovo settore in Y.
..054d 4c 0c 05 jmp $050c ; e ripete il ciclo da $050c

```

Disassemblato dalla routine di scansione dei file che viene allocata nel buffer \$0500 del disk drive.

## Datamaster: gestione L.M. dei Data Basic

di Roberto Morassi - Pistoia

Datamaster è una raccolta di due routine distinte per la creazione di DATA in un programma Basic. La prima (Dataenter) di utilizza quando si debba digitare un listato Basic molto lungo, o si abbia una serie di valori esadecimali relativi a routine in linguaggio macchina. La seconda (Datamemory) converte in DATA un blocco di memoria, sia ROM che RAM nascosta, molto più rapidamente di altri programmi analoghi dato che non ricorre mai al Basic né al buffer di tastiera.

Dopo il RUN, se i DATA sono stati introdotti correttamente, potete digitare RUN 1000 per salvare, col nome Datamaster/MC, la zona di memoria contenente le routine. In seguito, sarà sufficiente caricarla con LOAD «Datamaster/MC», 8,1 (seguito da NEW) per averle immediatamente disponibili.

All'occorrenza, Datamaster è completamente rilocabile senza modifiche in qualunque altra zona di memoria, così da poter coesistere con altre routine personali e non sovrapporsi alla zona che si intende convertire in DATA. È sufficiente cambiare l'indirizzo di partenza alla riga 850, e ricalcolare i nuovi valori dei SYS.

### Routine Dataenter

La sintassi da usare per attivare Dataenter è la seguente:

```
SYS 49152, LINE, INCR, FLAG
```

con il seguente significato:

LINE: numero di linea Basic di inizio dei DATA

INCR: incremento di linea

FLAG: ponendo FLAG uguale a zero, la routine opera nel modo «decimale»; con FLAG maggiore di zero, si opera invece nel modo «esadecimale».

### Modo «decimale»

Lo schermo verrà cancellato, e comparirà in alto il primo numero di linea, seguito da «DATA». Potete adesso digitare i valori decimali da inserire, tutti di seguito: se sono a tre cifre, la virgola verrà inserita automaticamente (lo stesso vale per i formati 001, 076, ecc.). Per valori con meno di tre cifre, dovrete inserire la virgola premendo il tasto «←» o il tasto «+» a piacere, così da

ghezza del file riportata in directory non corrisponda a quella vera, è possibile aggiornarla al valore corretto. Vengono segnalati i «files» fittizi che puntano alla traccia 18 (in genere linee di separazione o commenti), nonché quelli difettosi che presentano un «link» illegale (traccia o settore al di fuori dei limiti consentiti). Alla fine, viene riportato il numero totale dei blocchi occupati, e quelli liberi risultanti dalla directory, dalla scansione effettuata e dalla BAM. Se quest'ultimo valore è discordante, si può effettuare un «validate» per aggiornarlo, a meno che non si presuma la presenza di «blocchi nascosti» ad accesso diretto, che siano stati allocati per proteggerli da sovrascrittura.

### (2) Cambia nome e aggiorna ID

È opportuno ricordare che l'ID di due caratteri che viene assegnato al dischetto al momento della formattazione deve essere diverso da disco a disco, pena la possibile sovrascrittura di blocchi occupati. La cosiddetta «modifica dell'ID» registrato nella directory, incautamente effettuata da certe utility, è a questo riguardo del tutto inutile e molto pericolosa, dato che poi non si saprà più qual è l'ID reale. Con questa routine, oltre a cambiare il nome del disco, viene aggiornato l'ID nella directory riportandolo al codice «vero» che è formattato all'inizio di ogni traccia. Premendo RETURN a vuoto, l'operazione viene annullata.

### (3) Azzerare la BAM

Se si vuole proteggere un disco dalla scrittura di nuovi file, si possono allocare con questa routine tutti i blocchi residui, azzerando completamente la BAM («0 blocks free»). Volendo, è comunque possibile ricostruire la BAM

tramite un «validate».

### (4) Protegge/sprotegge da scrittura

La routine permette di proteggere dalla cancellazione (scratch) alcuni o anche tutti i file di un disco, nonché di effettuare l'operazione inversa (sprotezione). Ricordiamo che i file così protetti appaiono contrassegnati, nella lista della directory, con un simbolo «<» dopo il tipo di file.

È possibile infine proteggere l'intero disco da qualunque operazione di scrittura (scratch, rename, Save, ecc.) modificando il codice di formato «A» (CHR\$(65)) all'inizio del settore 18,0. Ripristinando tale codice, la protezione viene eliminata.

### (5) Cambia address

L'indirizzo di memoria da cui è stato salvato un file-programma (PRG) è registrato all'inizio di esso, nel formato base/alto byte, e viene utilizzato da Sistema Operativo per riallocare il file nella zona di memoria originale quando lo si carica con LOAD «FILE», 8,1. Con questa routine è possibile modificare tale indirizzo, così da rilocalizzare automaticamente il file in una zona di memoria diversa. Ciò può servire, ad esempio, per convertire schermate grafiche, set di caratteri ecc., da una utility grafica ad un'altra, o per rilocalizzare routine in linguaggio macchina.

### (6) Directory

Breve routine di lettura della directory, che può essere richiamata da vari punti del programma. Per fermare il listato ed esaminarlo, si tiene premuto il tasto SHIFT; per interromperlo prima della fine si preme il tasto COMMODE.

```

..c000 20 f1 b7 jsr $b7f1 ; raccoglie un parametro
..c003 8a txa ; dal testo (n. sprite)
..c004 29 07 and #$07 ; lo normalizza modulo 7
..c006 85 fb sta $fb ; e deposita in $FB
..c008 20 fd ae jsr $ae fd ; salta la virgola
..c00b 20 8a ad jsr $ad 8a ; raccoglie un parametro (coord. x)
..c00e 20 f7 b7 jsr $b7 f7 ; lo converte in intero a 2 bytes
..c011 a9 00 lda #$00 ; azzerà A
..c013 38 sec ; setta il carry
..c014 a6 fb ldx $fb ; contatore = n. sprite
..c016 2a rol ; sposta il carry in A (bit = 1)
..c017 ca dex ; per tante volte
..c018 10 fc bpl $c016 ; quant'è il n. sprite (in $FB)
..c01a 85 fc sta $fc ; deposita in $FC (bit a l= n. sprite)
..c01c 49 ff eor #$ff ; inverte tutti i bits in A
..c01e 2d 10 d0 and $d010 ; AND con $d010 (copia gli altri bits)
..c021 8d 10 d0 sta $d010 ; e ricopia A in $d010
..c024 a5 fc lda $fc ; recupera il valore in $FC
..c026 a6 15 ldx $15 ; controlla hi-byte della coord. x
..c028 f0 06 beq $c030 ; e' zero (coord. < 256) ?
..c02a 0d 10 d0 ora $d010 ; no: setta il bit corrispondente
..c02d 8d 10 d0 sta $d010 ; in $d010
..c030 18 cbc ; si': moltiplica per due
..c031 06 fb asl $fb ; il contatore in $FB
..c033 a5 14 lda $14 ; carica in A il lo-byte della coord. x
..c035 a6 fb ldx $fb ; usa X come indice al registro-x
..c037 9d 00 d0 sta $d000,x ; dello sprite: vi copia il lo-byte,
..c03a 20 f1 b7 jsr $b7f1 ; legge un parametro (coord. y).
..c03d 8a txa ; lo copia in A, poi
..c03e a6 fb ldx $fb ; usa X come indice al registro-y
..c040 9d 01 d0 sta $d001,x ; dello sprite: vi copia la coord. y
..c043 60 rts ; ed esce dalla routine

```

Disassemblato della routine «posizione sprite».

```

100 REM ** POSIZIONA SPRITES - BY R.MORASSI **
110 REM" ** ESEMPIO D'USO:SYS 49152, N, X, Y **
120 REM" ** N=NUMERO DELLO SPRITE **
130 REM" ** X=COORDINATA ORIZZONTALE **
140 REM" ** Y=COORDINATA VERTICALE **
150 :
160 FORX=0TO67:READY:A=A+Y:POKE49152+X,Y:NEXT
170 IFA<>8964THENPRINT"ERRORE NEI DATA":STOP
180 STOP
190 :
200 REM ** DIMOSTRAZIONE (RUN 220) **
210 :
220 V=53248:REM INIZIO VIC II
230 FORK=0TO7:POKE2040+K,7:NEXT:REM IMMAGINI SPRITES
240 POKEV+21,255:REM ATTIVA TUTTI GLI SPRITES
250 POKEV+23,255:POKEV+29,255:REM ESPANDE X - Y
260 POKE53281,0:PRINT"(CLR)"
270 FORK=1TO250:SYS49152,K,10+320*RND(1),10+240*RND(1):NEXT
280 END
290 :
300 DATA 032,241,183,138,041,007,133,251
310 DATA 032,253,174,032,138,173,032,247
320 DATA 183,169,000,056,166,251,042,202
330 DATA 016,252,133,252,073,255,045,016
340 DATA 208,141,016,208,165,252,166,021
350 DATA 240,006,013,016,208,141,016,208
360 DATA 024,006,251,165,020,166,251,157
370 DATA 000,208,032,241,183,138,166,251
380 DATA 157,001,208,096

```

READY.

non scendere mai sotto la prima fila di tasti. La pressione accidentale di altri tasti verrà ignorata, salvo il «DELETE» che potrete usare per correggere eventuali errori. La routine provvederà ad aggiornare i numeri di linea, con l'incremento scelto.

Quando avrete introdotto l'ultimo valore, premete RETURN e mantenete premuto finché il cursore non riappare in basso: in tal modo, le linee DATA verranno aggiunte al programma in memoria. Se i valori da inserire occupano più di una schermata, dovrete effettuare l'operazione in più volte: per sicurezza, la routine si blocca alla fine della

penultima riga di schermo, in attesa del RETURN, così da evitare lo scrolling. Notate che non viene effettuato nessun controllo sui dati introdotti: se devono essere POKati in memoria, dovrete fare attenzione a non mettere, per errore, valori > 255 pena un successivo ILLEGAL QUANTITY ERROR.

#### Modo «esadecimale»

Vi sarà certamente capitato di dover convertire in linee Basic di DATA decimali delle routine in linguaggio macchina listate in esadecimale (notazione comunemente usata). In realtà si potrebbero usare anche i DATA esadecimali, purché si inserisca nel programma una

subroutine che li converta, al momento della lettura, nei valori decimali richiesti dal POKE, e ciò a scapito della velocità. La conversione manuale in valori decimali, così come l'uso dei monitor o assembler, sono operazioni lunghe, macchinose e spesso soggette ad errori. Con Dataenter, basterà digitare:

```
SYS 49152, LINE, INCR, FLAG
```

con FLAG uguale ad 1 o più. Il funzionamento è analogo a quello descritto sopra: introducete i valori esadecimali tutti di seguito, senza spazi né altri separatori, come se fosse una stringa unica; per es. «A9808D8A022081FF...». Dataenter calcolerà i corrispondenti valori decimali e li stamperà formattandoli a tre cifre (con degli «0» in testa). Verranno accettati solo i valori da 0 a 9 e da A ad F, più il «DELETE» per la correzione degli errori.

#### Routine Datamemory

Con questa utility si possono convertire i valori di tutti i byte di un determinato blocco di memoria in altrettanti DATA decimali, da utilizzare poi all'interno di un programma Basic. A differenza di analoghe utility in Basic, Datamem esegue la conversione in tempi brevissimi, circa 7 secondi ogni 1000 byte: si tenga tuttavia presente che le linee di DATA così create non si aggiungono ad un eventuale programma Basic già residente in memoria, ma lo sostituiscono del tutto: il resto del programma che utilizza i DATA dovrà essere digitato in un secondo tempo, oppure aggiunto tramite una utility di MERGE o APPEND.

La sintassi da usare è la seguente:

```
SYS 49450, START, END+1, LINE, INCR, FLAG
```

dove i parametri hanno il seguente significato:

START, END: prima ed ultima locazione della zona di memoria da convertire in DATA;

LINE, INCR: numero della prima linea dei DATA, ed incremento di linea;

FLAG: se uguale a zero, nelle eventuali zone di memoria ROM verrà letta la ROM; se FLAG è maggiore di zero, verrà invece letta la RAM sottostante.

I data così creati saranno formattati a tre cifre (con «0» in testa), in gruppi di otto per linea. Alla fine, date un CLR per resettare i puntatori delle variabili, e poi il LIST per controllare il regolare funzionamento della routine. Potete adesso salvare il gruppo di linee-DATA con un normale SAVE, e incorporarle in un programma Basic come indicato in precedenza.

MC

# Elenco del software disponibile su supporto magnetico

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 sotto 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.**

| Codice                 | Titolo Programma                               | MC n. | Prezzo |
|------------------------|--|-------|--------|
| <b>COMMODORE AMIGA</b> |  |       |        |
| DAM/01                 | F-15   | 63    | 15000  |
| DAM/02                 | Gest. lista programmi                          | 64    | 15000  |
| DAM/03                 | Studio di Funzioni                             | 66    | 15000  |
| DAM/04                 | Math Pack                                      | 68    | 15000  |
| DAM/05                 | Redcode & Mars (Core Wars)                     | 68    | 15000  |
| DAM/06                 | Life   | 69    | 15000  |
| DAM/07                 | Rubrica Telefonica                             | 70    | 15000  |
| DAM/08                 | Piramidi                                       | 70    | 15000  |
| DAM/09                 | Regolazione dei colori                         | 71    | 15000  |
| DAM/10                 | Analitica                                      | 71    | 15000  |
| DAM/11                 | Grafici  | 72    | 15000  |
| DAM/12                 | Traduttore                                     | 73    | 15000  |
| DAM/13                 | La Borsa                                       | 74    | 15000  |
| DAM/14                 | DMA Music Compiler                             | 74    | 15000  |
| DAM/15                 | Poker  | 78    | 15000  |
| DAM/16                 | Programmi per il Copper                        | 79    | 15000  |
| DAM/17                 | Mandelbrot mania                               | 81    | 15000  |
| DAM/18                 | SF-Search File                                 | 86    | 15000  |
| DAM/19                 | AMipac   | 87    | 15000  |
| DAM/20                 | Click  | 93    | 15000  |
| DAM/21                 | Editor di Videogame                            | 95    | 15000  |
| DAM/22                 | RandomLand                                     | 98    | 15000  |
| <b>MS-DOS</b>          |  |       |        |
| DMS/01                 | Plotter + Morse                                | 67    | 15000  |
| DMS/02                 | Meloe + Spawn                                  | 68    | 15000  |
| DMS/03                 | Pretty + Scritte scorrevoli + Compute          | 69    | 15000  |
| DMS/04                 | Emulatore CGA per Hercules                     | 70    | 15000  |
| DMS/05                 | Turbo Directory                                | 71    | 15000  |
| DMS/06                 | Math Tool S                                    | 72    | 15000  |
| DMS/07                 | Bionimi + Routine                              | 72    | 15000  |
| DMS/08                 | Salvavideo + Scritte scorrev. + PG151          | 73    | 15000  |
| DMS/09                 | Optimizer + Indenter dBase III                 | 74    | 15000  |
| DMS/10                 | Joystick Controller                            | 75    | 15000  |
| DMS/11                 | BootSlow & SlowDown + Turbo Utility            | 76    | 15000  |
| DMS/12                 | Redcode & Mars (Core Wars)                     | 76    | 15000  |
| DMS/13                 | Gestione Errori Critici Disco + PosCur         | 77    | 15000  |
| DMS/14                 | Finestre & Desk                                | 78    | 15000  |
| DMS/15                 | General Manager                                | 78    | 15000  |
| DMS/16                 | Tool 05  | 79    | 15000  |
| DMS/17                 | Pulidown Menu + Retrace                        | 80    | 15000  |
| DMS/18                 | Righe  | 81    | 15000  |
| DMS/19                 | La spada di Krall                              | 82    | 15000  |
| DMS/20                 | Rappresione                                    | 82    | 15000  |
| DMS/21                 | Tesseract + Charset Editor                     | 83    | 15000  |
| DMS/22                 | Sega-File + Installatore                       | 84    | 15000  |
| DMS/23                 | Shape Tool-kit                                 | 87    | 15000  |
| DMS/24                 | Starfox  | 88    | 15000  |
| DMS/25                 | Inline in Turbo C + Svuota Keyboard Buffer     | 89    | 15000  |
| DMS/26                 | Sistemi parametrici + GIFDIR 1.0               | 90    | 15000  |
| DMS/27                 | QuickMatrix + Stampa Virtuale                  | 91    | 15000  |
| DMS/28                 | Wheris + Bold C + Setmanu                      | 92    | 15000  |
| DMS/29                 | Bobo + MGA 100                                 | 93    | 15000  |
| DMS/30                 | War+LS (2 dischi)                              | 94    | 20000  |
| DMS/31                 | Warbot Arena                                   | 95    | 15000  |
| DMS/32                 | Crouse + Sp-Build + Tratt.Lsp                  | 96    | 15000  |
| DMS/33                 | Il diavoleto di Maxwell + Criminals (2 dischi) | 97    | 20000  |
| DMS/34                 | Num-Unlock & ESC + Math Parser                 | 98    | 15000  |
| DMS/35                 | Convert + Hard Copy                            | 99    | 15000  |
| <b>ATARI ST</b>        |  |       |        |
| DST/01                 | Virus Killer                                   | 74    | 15000  |
| DST/02                 | Mandelbrot + Proiez. Ort. + Bianco             | 78    | 15000  |
| DST/03                 | Diagrammi di Henon                             | 81    | 15000  |
| DST/04                 | Paroliamo                                      | 84    | 15000  |
| DST/05                 | Enalotte                                       | 85    | 15000  |
| DST/06                 | Cariche  | 83    | 15000  |
| DST/07                 | Flow-Chart                                     | 87    | 15000  |
| DST/08                 | Funz Manual                                    | 89    | 15000  |
| DST/09                 | RedCode  | 88    | 15000  |
| DST/10                 | Pattern Generator 2.1                          | 92    | 15000  |
| DST/11                 | ST Hopalong + Moto casuale                     | 93    | 15000  |
| DST/12                 | Speedy Maid                                    | 94    | 15000  |
| DST/13                 | Maed   | 94    | 15000  |
| DST/14                 | Designer                                       | 95    | 15000  |
| DST/15                 | Pro-Function 1.0                               | 96    | 15000  |
| DST/16                 | Tis + Carame                                   | 97    | 15000  |
| DST/17                 | Maed + Print + Multitas + Memmon               | 98    | 15000  |
| DST/18                 | Memoria + Griglia                              | 99    | 15000  |
| <b>COMMODORE 128</b>   |  |       |        |
| D28/01                 | MMCalc   | 53    | 15000  |
| D28/02                 | Hardcopy 128                                   | 55    | 15000  |
| D28/03                 | Sheet II                                       | 57    | 15000  |
| D28/04                 | Star Quest                                     | 58    | 15000  |
| D28/05                 | Family Budget                                  | 60    | 15000  |
| D28/06                 | La Casa Stregata                               | 61    | 15000  |
| D28/07                 | Struttura 80/33                                | 63    | 15000  |
| D28/08                 | Bas 80 V. 2.0a                                 | 64    | 15000  |
| D28/09                 | Paint 80 1.0                                   | 65    | 15000  |
| D28/10                 | Bas 80 V. 2.11                                 | 66    | 15000  |
| D28/11                 | Calendario Perpetuo + Montecarlo               | 67    | 15000  |

| Codice   | Titolo Programma                         | MC n. | Prezzo |
|--|--|-------|--------|
| D28/12   | Disegna Circuiti                         | 68    | 15000  |
| D28/13   | Marc's Data Base                         | 70    | 15000  |
| D28/14   | Label Disk + Disk Editor + Dem DOS       | 71    | 15000  |
| D28/15   | Pulidown 128HR + Menu + Drawer           | 72    | 15000  |
| D28/16   | Prospettive                              | 73    | 15000  |
| D28/17   | Char 80 V. 1.0                           | 74    | 15000  |
| D28/18   | Italia 128                               | 75    | 15000  |
| D28/19   | Super Sprite                             | 77    | 15000  |
| D28/20   | Othello                                  | 80    | 15000  |
| D28/21   | Expert System Shell 128                  | 81    | 15000  |
| D28/23   | Caratteri Programmabili                  | 83    | 15000  |
| D28/24   | Brush & Mouse                            | 84    | 15000  |
| D28/25   | Astrolabio                               | 89    | 15000  |
| D28/26   | Sort Vettore Stringa + Input Stringa     | 90    | 15000  |
| D28/27   | Piano Cartesiano + Calendario            | 91    | 15000  |
| D28/28   | Bingoi 90                                | 92    | 15000  |
| D28/29   | Koala View + Clock 128 + File Data Maker | 93    | 15000  |
| D28/30   | Curve Polari                             | 94    | 15000  |
| D28/31   | Biblio 128                               | 95    | 15000  |
| <b>COMMODORE 64</b>  |  |       |        |
| D64/14   | Link-64                                  | 57    | 15000  |
| D64/15   | New Char 2.2                             | 58    | 15000  |
| D64/16   | Music 64                                 | 59    | 15000  |
| D64/17   | TRX-MEM                                  | 59    | 15000  |
| D64/18   | WOS + WBasic                             | 60    | 15000  |
| D64/19   | Strange Basic + Dracula                  | 63    | 15000  |
| D64/20   | File Rescue                              | 64    | 15000  |
| D64/21   | La Casa                                  | 64    | 15000  |
| D64/22   | Digital Voice                            | 65    | 15000  |
| D64/23   | Vita 3D                                  | 65    | 15000  |
| D64/24   | Corso di Linguistica                     | 66    | 15000  |
| D64/25   | Archipius                                | 66    | 15000  |
| D64/26   | Math Pack Plus                           | 66    | 15000  |
| D64/27   | Scroll + Multitask + Classifica          | 67    | 15000  |
| D64/28   | Calend. Perpetuo + Effetto Telecamera    | 68    | 15000  |
| D64/29   | Listing Plus + Utility Data              | 69    | 15000  |
| D64/31   | Trucchi a Routine per programmatori      | 71    | 15000  |
| D64/32   | Flow-Chart + Flower's Love               | 73    | 15000  |
| D64/33   | Sprite Editor                            | 76    | 15000  |
| D64/34   | Portolio 64 + Elimin. bordi schermo      | 77    | 15000  |
| D64/37   | Intonatore                               | 81    | 15000  |
| D64/38   | Gendata 64                               | 82    | 15000  |
| D64/40   | Block Utility                            | 84    | 15000  |
| D64/41   | Sprite 6499                              | 85    | 15000  |
| D64/42   | Mot Mot Labyrinth + Tool 64              | 86    | 15000  |
| D64/43   | Titolatore 64                            | 88    | 15000  |
| D64/44   | Editor Musicale                          | 89    | 15000  |
| D64/45   | Dump Variabili                           | 90    | 15000  |
| D64/46   | FBasic V1.0 + Tod-Basic                  | 91    | 15000  |
| D64/47   | Le sedici carte + Stampa Directory       | 92    | 15000  |
| D64/48   | REL-Arch 64 Statistica 3.0               | 94    | 15000  |
| D64/49   | MegaGame 64                              | 95    | 15000  |
| D64/50   | PK/Pok                                   | 96    | 15000  |
| D64/51   | Sam: Scritte scorrevoli                  | 97    | 15000  |
| D64/52   | Overlay + Easy Format + Editor 160       | 98    | 15000  |
| D64/53   | Diskpack + Datamaster                    | 99    | 15000  |
| <b>MSX</b>   |  |       |        |
| DMX/01   | Toto 13                                  | 60    | 15000  |
| DMX/02   | Painter                                  | 62    | 15000  |
| DMX/03   | MSX Bank                                 | 63    | 15000  |
| DMX/04   | Grafica 3D + Hard Copy                   | 65    | 15000  |
| DMX/05   | Easy Disk                                | 66    | 15000  |
| DMX/06   | Classifiche                              | 67    | 15000  |
| DMX/07   | Magic Paint                              | 67    | 15000  |
| DMX/08   | Autogest                                 | 68    | 15000  |
| DMX/09   | Compilatore v. 1.01                      | 69    | 15000  |
| DMX/10   | Diskmap                                  | 70    | 15000  |
| DMX/11   | Mini dBase MSX                           | 71    | 15000  |
| DMX/12   | Grafica in Turbo Pascal                  | 72    | 15000  |
| DMX/13   | Math Pack Plus 3.20                      | 73    | 15000  |
| DMX/14   | RGB CAD                                  | 75    | 15000  |
| DMX/15   | Simple Desk                              | 76    | 15000  |
| DMX/16   | The MSX2 Super Print                     | 77    | 15000  |
| DMX/17   | Grafica in Turbo Pascal (Graph 1&2)      | 77    | 15000  |
| DMX/18   | Hard Copy                                | 78    | 15000  |
| DMX/19   | HEXDUMP                                  | 79    | 15000  |
| DMX/20   | Utilities in Turbo Pascal                | 80    | 15000  |
| DMX/21   | dBase MSX Plus                           | 81    | 15000  |
| DMX/23   | PutChar + 4010 S                         | 84    | 15000  |
| DMX/24   | Analysis + Lister Basic                  | 85    | 15000  |
| DMX/25   | Aliment MSX                              | 86    | 15000  |
| DMX/26   | SortDir + Hectogonus + 3D Mover          | 87    | 15000  |
| DMX/27   | Tool/Extended Basic + Bindata.Bas        | 88    | 15000  |
| DMX/28   | Chain & Window + Chain Utility           | 90    | 15000  |
| DMX/29   | Latino Pack                              | 91    | 15000  |
| DMX/30   | Anoid                                    | 97    | 15000  |
| DMX/31   | MSX-DOS Utility                          | 98    | 15000  |
| DMX/32   | MSX-Editor Tool Box                      | 99    | 15000  |
| <b>Nota:</b><br>Il supporto è rappresentato da dischi da 5.25" e 3.5" in relazione alle caratteristiche dei disk drive presenti sul computer in oggetto. |  |       |        |