

software

C-64

a cura di Tommaso Pantuso

H COPY 802

di Giorgio Moretti (Fo)

L'unico difetto della 802 è quello di non essere una stampante grafica, ovvero di non permettere la copia su carta di una schermata in altra risoluzione ma solo la stampa dei caratteri del C64.

Questa limitazione non è dovuta all'hardware della macchina ma al programma che la gestisce. Ciò significa che la stampante è praticamente in grado di copiare disegni e grafici in alta risoluzione ma non è stata predisposta per farlo.

Una conferma di questo fatto è la possibilità di definire un carattere, tramite una matrice di 8×8 punti, e di stamparlo anche se non appartiene al set di caratteri del computer.

La pagina grafica che vogliamo stampare è costituita da 320×200 punti ma può essere vista anche come un rettangolo formato da 40×25 quadratini ciascuno dei quali è formato da 8×8 punti (infatti $40 \times 8 = 320$ e $25 \times 8 = 200$).

Possiamo quindi definire il carattere visto prima in modo che sia uguale al primo quadratino di 8×8 punti, stamparlo, ridefinirlo in modo che sia uguale al secondo quadratino, stamparlo dopo il primo e così via per tutti gli altri fino alla fine della schermata (ovvero dei quadratini).

Il problema sarebbe risolto abbastanza agevolmente se non fosse che la stampante permette di definire un solo carattere per riga mentre noi dobbiamo definirne ben 40.

Il problema viene risolto con un piccolo trucco: la stampante si accorge che è finita una riga quando gli arriva un segnale di a capo (codice ASCII 13); questo codice fa sì che la testina vada a capo (ovvio!) e avanzi alla riga

successiva. A questo punto è possibile definire un altro carattere ma, per il nostro scopo, è troppo tardi in quanto la testina della stampante è avanzata di una riga.

Se noi, invece, dopo aver stampato il primo carattere definito inviamo alla stampante un codice che la faccia andare a capo ma senza avanzare alla riga successiva (codice ASCII 141), sarà poi possibile definire un altro carattere e stamparlo di seguito al primo.

Per fare questo occorre però riposizionare la testina di stampa che, andando a capo, è tornata all'inizio della riga. Questo viene fatto inviando una serie di spazi vuoti.

Questo continuo tornare all'inizio

della riga da parte della testina si traduce inevitabilmente in una lentezza di stampa incredibile; fortunatamente però la 802 è una macchina «intelligente», in quanto conosce la colonna a cui si trova ed evita di tornare indietro se vengono inviati gli spazi necessari a farla tornare nella posizione a cui si trovava prima di andare a capo. La stampa risulta quindi continua e abbastanza veloce: circa 5 minuti.

L'immagine che si ottiene in questo modo non è però molto grande (40×25 caratteri normali); ho pensato quindi di aggiungere la possibilità di raddoppiarne la lunghezza (80×25 caratteri), l'altezza (40×50 caratteri) o entrambe le dimensioni (80×50 carat-

Errata corrige

A causa di un disguido nella spedizione dei dischetti contenenti il programma Geo-Convert 1.2, pubblicato sul numero scorso, l'autore ha inviato una versione non completa.

Così come è stato pubblicato, il programma NON converte le maiuscole, causando dei problemi quando si tenta, nel file GeoWrite, di cambiare il font di un blocco di caratteri.

Quando infatti si evidenzia un blocco di testo e si cambia il font dei caratteri, tutti i caratteri all'interno del blocco vengono cambiati, ma le maiuscole presentano delle anomalie (sono troppo strette o occupano lo spazio di molti caratteri). Ciò avviene perché la sub in LM della versione pubblicata non converte le maiuscole, e, anziché portarle in un range \$41-\$5A (come previsto) le lascia in \$C1 e \$DA (cioè con codici del SuperScript).

L'errore sta nella routine LM allocata in \$C000-\$C038, cioè quella incaricata di leggere e convertire il file SuperScript. Le modifiche da fare sono:

```
C016 30 13 bmi $c02b ----> 30 09 bmi $c021
```

```
C029 e9 80 sbc #$80 ----> e9 7f sbc #$7f
```

Dunque, vanno corrette le seguenti linee del listato GEO CONVERT L.M.:

```
220 if ck<>41455 then print "errore nei data!":end
      ↑↑↑↑↑
```

```
290 data 49, 9, 201, 91, 16, 5, 105, 32, 76, 43, 192
      ↑
```

```
300 data 201, 192, 48, 6 201, 219, 16, 2, 233, 127, 141
      ↑↑↑
```

H COPY 802

```

5 PRINT"(CLR)(YEL)(DOWN) ATTENDE UN ATTIMO. STO' CONTANDO(DOWN)(DOWN)(DOWN)
(DOWN)(DOWN)(DOWN)(DOWN) :PRINTTAB(16)"HCOPY"
10 FORA=49184TO49833:READB:POKEA,B
15 PRINT"(HOME)(DOWN)(RIGHT) (LEFT)(LEFT)(LEFT) "49833-A:
20 H=B:B-NEXT
30 IFH<86637THENPRINT"(CLR)(DOWN) DATA ERROR":STOP
40 PRINT"(CLR)(DOWN)OK HO FINITO. IL PROGRAMMA E' IN MEMORIA"
100 END
60000 REM DATA
60005 DATA 32, 138, 173, 76, 247, 183, 234, 234, 76, 72, 178, 234, 234
60010 DATA 234, 234, 234, 32, 253, 174, 32, 32, 192, 165, 21, 208, 238
60015 DATA 165, 20, 201, 4, 176, 232, 141, 0, 192, 32, 253, 174, 32
60020 DATA 32, 192, 165, 21, 208, 219, 173, 0, 192, 106, 144, 5, 169
60025 DATA 0, 76, 94, 192, 165, 20, 201, 41, 176, 202, 141, 1, 192
60030 DATA 32, 253, 174, 32, 32, 192, 165, 20, 133, 251, 24, 105, 64
60035 DATA 133, 253, 165, 21, 133, 252, 105, 31, 133, 254, 32, 253, 174
60040 DATA 32, 32, 192, 165, 21, 208, 166, 165, 20, 240, 34, 201, 201
60045 DATA 176, 158, 141, 8, 192, 165, 251, 133, 253, 165, 252, 133, 254
60050 DATA 174, 8, 192, 24, 165, 253, 105, 64, 133, 253, 165, 254, 105
60055 DATA 1, 133, 254, 192, 208, 240, 169, 0, 141, 6, 192, 141, 7
60060 DATA 192, 141, 5, 192, 141, 4, 192, 165, 254, 201, 160, 144, 3
60065 DATA 238, 5, 192, 32, 204, 255, 32, 231, 255, 169, 4, 162, 4
60070 DATA 160, 0, 32, 242, 192, 169, 6, 162, 4, 160, 6, 32, 242
60075 DATA 192, 169, 5, 162, 4, 160, 5, 32, 242, 192, 162, 6, 32
60080 DATA 201, 255, 169, 24, 32, 210, 255, 169, 13, 32, 210, 255, 76
60085 DATA 254, 192, 32, 186, 255, 169, 0, 32, 189, 255, 32, 192, 252, 65
60090 DATA 96, 32, 46, 193, 32, 243, 193, 32, 9, 194, 32, 31, 194
60095 DATA 32, 55, 194, 32, 88, 194, 32, 99, 194, 165, 252, 197, 254
60100 DATA 208, 229, 165, 251, 197, 253, 208, 223, 169, 13, 32, 210, 255
60105 DATA 32, 210, 255, 32, 204, 255, 32, 231, 255, 96, 173, 5, 192
60110 DATA 240, 10, 165, 1, 141, 5, 192, 41, 248, 120, 133, 1, 173
60115 DATA 0, 192, 74, 176, 22, 160, 0, 177, 251, 153, 16, 192, 200
60120 DATA 192, 8, 208, 246, 173, 0, 192, 201, 2, 240, 100, 76, 234
60125 DATA 193, 160, 0, 173, 6, 192, 208, 18, 162, 4, 177, 251, 32
60130 DATA 160, 193, 200, 192, 8, 208, 244, 238, 6, 192, 76, 133, 193
60135 DATA 162, 4, 177, 251, 10, 10, 10, 10, 32, 160, 193, 200, 192
60140 DATA 8, 208, 240, 206, 6, 192, 162, 0, 189, 16, 192, 74, 29
60145 DATA 16, 192, 157, 16, 192, 232, 224, 8, 208, 241, 173, 0, 192
60150 DATA 201, 3, 240, 29, 76, 234, 193, 10, 142, 2, 192, 140, 3
60155 DATA 192, 174, 3, 192, 62, 16, 192, 30, 16, 192, 174, 2, 192
60160 DATA 172, 3, 192, 202, 208, 231, 96, 173, 7, 192, 208, 23, 162
60165 DATA 3, 160, 8, 189, 16, 192, 136, 153, 16, 192, 136, 153, 16
60170 DATA 192, 202, 224, 255, 208, 240, 76, 234, 193, 162, 0, 160, 4
60175 DATA 185, 16, 192, 157, 16, 192, 232, 157, 16, 192, 232, 200, 192
60180 DATA 8, 208, 240, 173, 5, 192, 240, 3, 133, 1, 88, 96, 160
60185 DATA 8, 136, 162, 8, 185, 16, 192, 202, 74, 126, 24, 192, 224
60190 DATA 0, 208, 247, 192, 0, 208, 237, 96, 238, 4, 192, 173, 6
60195 DATA 192, 208, 13, 24, 169, 8, 101, 251, 133, 251, 169, 0, 101
60200 DATA 252, 133, 252, 96, 162, 5, 32, 201, 255, 162, 0, 189, 24
60205 DATA 192, 32, 210, 255, 232, 224, 8, 208, 245, 169, 13, 32, 210
60210 DATA 255, 96, 162, 4, 32, 201, 255, 169, 32, 174, 1, 192, 240
60215 DATA 7, 32, 210, 255, 202, 76, 65, 194, 174, 4, 192, 202, 240
60220 DATA 7, 32, 210, 255, 202, 76, 78, 194, 96, 169, 254, 32, 210
60225 DATA 255, 169, 141, 32, 210, 255, 96, 173, 0, 192, 74, 176, 8
60230 DATA 173, 4, 192, 201, 40, 240, 9, 96, 173, 4, 192, 201, 80
60235 DATA 240, 1, 96, 169, 13, 32, 210, 255, 169, 0, 141, 4, 192
60240 DATA 173, 0, 192, 201, 2, 176, 1, 96, 173, 7, 192, 240, 4
60245 DATA 206, 7, 192, 96, 238, 7, 192, 24, 165, 251, 233, 63, 133
60250 DATA 251, 165, 252, 233, 1, 133, 252, 96, 0, 0, 0, 0

```

ESEMPIO H COPY 802

```

10 REM QUESTO PROGRAMMA E' DA USARE SOLO DOPO AVER CARICATO E LANCIATO
20 REM IL PROGRAMMA H COPY
30 REM
100 PRINT"(CLR)(YEL)(DOWN)(DOWN)(DOWN)(DOWN)(DOWN) GRAFICI DI FUNZIONI (S PER S
TAMPARE)(DOWN)(DOWN)(DOWN)"
110 PRINT"1- Y=SEN(X) :PRINT"2- Y=COS(X) :PRINT"3- Y=X^3-X+5"
120 PRINT"4- Y=SIN(X)*COS(X/2) :PRINT"5- Y=X*COS(X) :PRINT"(DOWN)0- FINE"
130 INPUT"(DOWN)(DOWN)COSA SCEGLI " :S$
140 S=VAL(S$) :IFS<0ORS>5THEN100
145 IF S=0THEN END
150 GOSUB10000:REM ATTIVA PAGINA GRAFICA
160 GOSUB11000:REM CANCELLA PAGINA GRAFICA
170 REM DISEGNA GLI ASSI
180 FORX=0TO319:Y=100:GOSUB20000:Y=0:GOSUB20000:Y=199:GOSUB20000:NEXT
190 FORY=0TO199:X=160:GOSUB20000:X=0:GOSUB20000:X=319:GOSUB20000:NEXT
195 REM DISEGNA GRAFICO
200 FORX1=150TO150
210 ON S GOSUB2000,2100,2200,2300,2400:REM CALCOLA VALORE FUNZIONE IN X1
220 X=X1+160:GOSUB20000:REM PLOT X,Y
230 NEXT
240 GETAS:IFAS=""THEN240
250 POKE53265,PEEK(53265)AND223:POKE53272,21 :REM DISATTIVA PAGINA GRAFICA
260 IFAS<>"S"THEN100
265 PRINT"(CLR)STAMPA"
270 INPUT"(DOWN)(DOWN)(DOWN)(DOWN)TIPO DI STAMPA (0-3) " :T
280 INPUT"(DOWN)COLONNA INIZIO STAMPA (0-40) " :C
290 INPUT"(DOWN)RIGHE DA STAMPARE (25-TUTTO) " :R
300 GOSUB10000:REM ATTIVA PAGINA GRAFICA
310 SYS49200,T,C,8192,R :REM STAMPA
330 AS="" :GOTO250
1999 REM FUNZIONI
2000 Y=100-SIN(X1/25)*60:RETURN
2100 Y=100-COS(X1/25)*60:RETURN
2200 Y=100-(X1/35)^3-X1/35+5:RETURN
2300 Y=100-SIN(X1/16)*COS(X1/32)*60:RETURN
2400 X2=X1/1.7:Y=100-X2*COS(X2/16):RETURN
9999 REM
10000 POKE53265,PEEK(53265)OR32:POKE53272,(PEEK(53272)AND240)OR8
10010 FORA=1024TO2023:POKEA,1:NEXT:RETURN
11000 FORA=8192TO16192:POKEA,0:NEXT:RETURN
20000 REM PLOT X,Y
20010 P=320*INT(Y/8)+8*INT(X/8)+(YAND7)
20020 B=7-(XAND7)
20030 POKEB192+P,PEEK(8192+P)OR21B
20040 RETURN

```

teri). In questo modo aumenta proporzionalmente anche il tempo di stampa che diventa di 10 minuti per il formato 80x25 e 40x50 e di 20 minuti per il formato da 80x50 caratteri.

Per le schermate che vengono stampate con la larghezza di 40 colonne è anche possibile scegliere in quale parte del foglio metterle, specificando la colonna da cui dovranno iniziare.

È inoltre possibile scegliere se stampare tutte le 25 righe di cui è formata la pagina grafica o solo una parte o addirittura di più (questa possibilità può essere utile nel caso si lavori con due schermi grafici messi uno dopo l'altro).

Uso del programma

Caricato il programma e dato il run, esso verrà messo in memoria a partire dalla locazione 49186 in modo da non interferire con i programmi in Basic.

Per usarlo, una volta preparata la pagina grafica da stampare, basta digitare direttamente o da programma:

SYS 49200, TIPO, POSIZIONE, INIZIO, LINEE

Dove:

TIPO indica il tipo di stampa che si vuole ottenere e può essere:

0 per la stampa normale di 40x25 caratteri,

1 per la stampa di larghezza doppia

Nota

I codici di controllo nei listati sono riportati in forma «esplicita», in conseguenza dell'impiego della stampante Star NL-10 e relativa interfaccia per Commodore. Ovviamente, nella digitazione del programma è necessario usare i consueti tasti che corrispondono alle indicazioni fra parentesi: ad esempio cursore destro per (RGHT), CTRL-3 per (RED) eccetera.

(CLR)	=	⏏	(YEL)	=	⏏
(HOME)	=	⏏	(RVS)	=	⏏
(DOWN)	=	⏏	(OFF)	=	⏏
(UP)	=	⏏	(ORNG)	=	⏏
(RGHT)	=	⏏	(BRN)	=	⏏
(LEFT)	=	⏏	(LRED)	=	⏏
(BLK)	=	⏏	(GRY1)	=	⏏
(WHT)	=	⏏	(GRY2)	=	⏏
(RED)	=	⏏	(LGRN)	=	⏏
(CYN)	=	⏏	(LBLU)	=	⏏
(PUR)	=	⏏	(GRY3)	=	⏏
(GRN)	=	⏏	(SWLC)	=	⏏
(BLU)	=	⏏			

cioè di 80 × 25 caratteri,

2 per la stampa di altezza doppia
cioè di 40 × 50 caratteri,

3 per la stampa di 80 × 50 caratteri.

POSIZIONE indica la colonna da cui dovrà iniziare la stampa del tipo 0 o 2. Essa può essere un numero compreso tra 0 (inizio del margine sinistro del foglio) e 40 (inizio della 40esima colonna e fino al margine destro del foglio).

Se il tipo è 1 o 3 questo parametro non serve in quanto la stampa deve per forza cominciare dalla prima colonna; potrà quindi esservi un valore qualunque.

INIZIO specifica la locazione di inizio della pagina grafica. Valori tipici per questo parametro sono 8192, 16384, 24576, 32768.

LINEE indica il numero di linee che si vogliono stampare. In particolare 25 equivale all'intera pagina grafica (25 × 8 = 200 punti che è l'altezza della schermata) e può essere sostituito anche con 0.

Applicazione del programma

Il programma permette la stampa di immagini in alta risoluzione 320 × 200 di punti con due colori, ovvero degli 8000 byte che la costituiscono.

Cercando di stampare immagini di 160 × 200 punti con 4 colori (in cui le due tonalità supplementari sono ottenute considerando la combinazione di due punti vicini) oppure di 320 × 200 punti ma con più di due colori (ottenuta utilizzando anche la memoria video e quella colore del computer, vedi ad esempio la tavoletta KoalaPainter), si potranno avere delle differenze dall'originale per quanto riguarda le parti del disegno che usano queste caratteristiche.

Per stampare un'immagine realizzata con KoalaPainter occorre, dopo aver caricato e avviato il programma, caricare in memoria il disegno con:

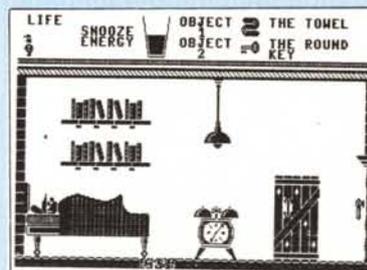
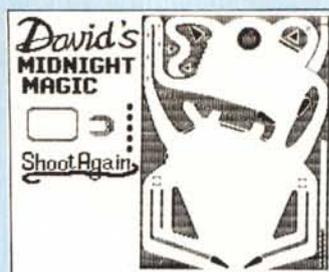
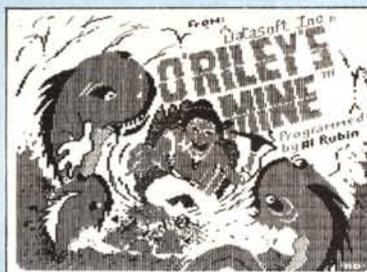
```
LOAD 'chr$(129)PIC+ nome disegno ',8,1
e digitare:
```

```
SYS 49200, TIPO, POSIZIONE, 24576, 25
```

Per stampare un'immagine realizzata con il Simon's Basic bisogna prima avviare il programma che realizza il disegno suddetto, quindi caricare e lanciare il programma Hcopy. A questo punto basterà digitare:

```
SYS 49200, TIPO, POSIZIONE, 57344, 25
```

In pratica il programma Hcopy va caricato in memoria dopo aver realizzato il disegno in quanto il Simon's Basic utilizza l'area di memoria in cui è situata la routine di stampa.

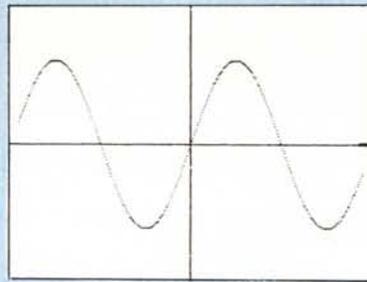


T.R.C.

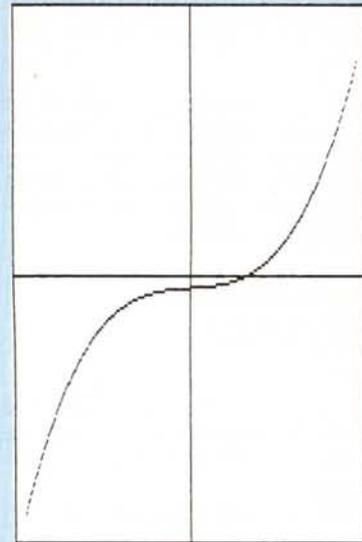
```
1 POKES3280,0:POKES3281,0:DIM X(50),A(50,50),T(50)
3 PRINT"(CLR)(CYN)(DOWN)TEOREMA DI ROUCHE CAPELLI VERSIONE C-64":
4 PRINT"(RED) BY "
5 PRINT" VITTORIO ACCOMAZZI "
10 INPUT"(DOWN)(DOWN)(DOWN)(WHT)NUMERO EQ.":N
20 FOR R=1 TO N
30 FOR C=1 TO N
40 INPUT"(CLR)(DOWN)(DOWN)(DOWN)COEFF.":A(C,R):NEXT C
50 INPUT"(DOWN)(DOWN)TERMINE NOTO":T(R):NEXT R
60 PRINT"(CLR)":GOSUB 1000:PRINT"(DOWN)(DOWN)(DOWN)(DOWN)Y/N"
61 GOSUB1015:IF A$="N"THEN RUN
62 PRINT"(CLR)(DOWN)(DOWN)(DOWN)(DOWN)(DOWN)(DOWN)ELABORO"
70 FOR I=N TO 2 STEP -1
81 FOR R=1 TO I:IF A(I,R)<>0 THEN T(R)=T(R)/A(I,R)
82 NEXT R
85 FOR R=1 TO I:IF A(I,R)=0 THEN P=P+1:X(P)=R:GOTO 121
90 FOR C=1 TO I
100 A(C,R)=A(C,R)/A(I,R)
120 NEXT C
121 NEXT R
122 GOSUB 240
130 FOR R=1 TO I-1
140 FOR C=1 TO I
150 A(C,R)=A(C,R)-A(C,I)
160 NEXT C:NEXT R
161 FOR C=1 TO I-1:T(C)=T(C)-T(I):NEXT C
170 NEXT I
171 IF A(1,1)=0 THEN 400
175 X(1)=T(1)/A(1,1)
176 PRINT"(CLR)(RED)X 1 --->(WHT)":T(1)/"A(1,1)"="X(1)
180 FOR I=2 TO N:X(I)=A(I,1):A=0
200 FOR C=1 TO I-1
210 A=A+A(C,I)*X(C)
220 NEXT C:X(I)=(-1*A+T(I))/X(I)
221 PRINT"(RED)X":I"--->(WHT)":(-1*A+T(I))/"A(I,1)"="X(I)
230 NEXT I:GOSUB 1015:RUN
240 REM
241 REM DENOMINATORE=0
242 REM
250 IF P=0 THEN RETURN
270 FOR Z=1 TO I
280 IF A(I,Z)<>0 THEN 300
290 NEXT Z:T(1)=0:GOTO 400
300 FORT=1TOP:FOR C=1 TO I
310 A(C,X(T))=A(C,X(T))+A(C,Z)
320 NEXT C:T(X(T))=T(X(T))+T(Z):NEXT T
330 FOR T=1 TO P
340 FOR C=1 TO I
350 A(C,X(T))=A(C,X(T))/A(I,X(T))
360 NEXT C
380 T(X(T))=T(X(T))/A(I,X(T))
390 NEXT T:P=0:RETURN
400 IF T(1)<>0 THEN PRINT"(CLR)(WHT) NON ESISTE ALCUNA SOLUZIONE
410 IF T(1)=0 THEN PRINT"(CLR)(WHT) ESISTONO INFINITE SOLUZIONI
430 GOSUB 1015:RUN
1000 PRINT"(CLR)":FOR R=1 TO N
1008 PRINT"(RED)R:--->(WHT)":FOR C=1 TO N
1009 PRINTA(C,R):NEXTC:PRINT"=:T(R):NEXTR:RETURN
1015 GET A$:IF A$="N"THEN 1015
1020 RETURN
```

◀ Esempi di schermate grafiche prese da alcuni giochi del C64.

Tipo 0

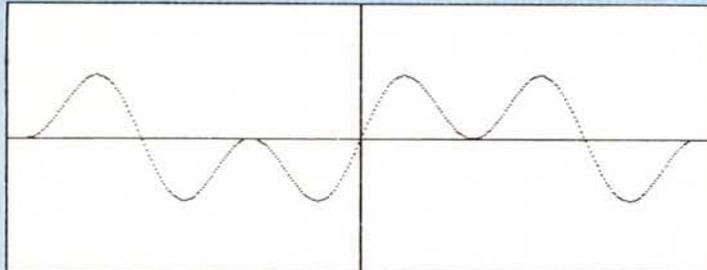


Tipo 2



Esempi di grafica ▶ con MSP 802.

Tipo 1



T.R.C. e Boot 1.2

di Vittorio Accomazzi, Novara

Sono un accanico lettore di MC frequente l'anno integrativo del Liceo Artistico perché intendo provare a seguire informatica a Milano. Studiando la logica matematica, il calcolo matriciale mi ha interessato molto per le sue applicazioni e perciò ho scritto, per il C 64, T.R.C. ovvero il teorema di Rouché Capelli per risolvere sistemi lineari di n equazioni in n incognite. Ho fatto anche un secondo programma che potrebbe interessare tutti coloro che hanno il disk drive. Si chiama *Boot* ed il suo compito è quello di leggere la directory e fare partire qualsiasi programma solo alla pressione di un tasto. Entrambi funzionano perfettamente ed hanno dei tempi di elaborazione brevi; il primo risolve i sistemi di 3 eq in meno di un secondo (compiuto), *Boot* legge una directory piena in 7 secondi (in Basic oltrepasserebbe abbondantemente il minuto).

Istruzioni per l'uso

T.R.C. Appena lanciato, il programma chiede il numero di equazioni del sistema (che deve coincidere con quello delle variabili). Dopo l'utente deve digitare tutte le equazioni, alla conferma dei dati il computer calcolerà le soluzioni.

BOOT 1,2 Lanciato il programma leggerà la directory e la mostrerà sullo schermo. Per far partire un programma nella colonna di destra schiacciare solo la lettera corrispondente, per la colonna di sinistra premere anche lo SHIFT. La differenza tra i due pro-

BOOT 1

```

1 DIMX(16):POKE253,0
2 FORT=49152T049222:READA:POKET.A:NEXT
3 FORT=49223T049279:READA:POKET.A:NEXT
4 FORT=49280T049297:READA:POKET.A:NEXT
10 OPEN 8,8,8,"$.S.R":POKE53280,2:POKE53281,2:POKE253,0:PRINT"(CLR)(WHT)":SYS492
80
20 PRINTTAB(3):SYS 49152:IFPEEK(53281)=240ORPEEK(253)>45THEN CLOSE 8:GOTO 50
21 SYS49223:PRINT"(WHT)":IFPEEK(253)=23THENPRINT"(HOME)(DOWN)":TAB(22):
30 IFPEEK(253)>23 THEN PRINTTAB(22):
31 GOTO 20
50 SYS 62259:PRINT"(HOME)":TAB(23):"(WHT)(RVS) +SHIFT ":POKE53280,0:POKE53281,
0
60 FORT=1 T023:PRINTTAB(19):"(RED)(RVS) ":CHR$(T+64):" ":NEXT:PRINT"(HOME)"
61 FORT=1T023:PRINTTAB(0):"(RED)(RVS) ":CHR$(T+64):" ":NEXT
70 GET AS:IFAS="" THEN70
81 IF AS=CHR$(13) THENOPEN15,8,15,"I":CLOSE15:CLR:GOTO10
82 X=ASC(AS)-64:IFX>23THENX=X-105
83 X=X*40+3:IFX>963THENX=X+19-23*40
84 FORT=0T016:X(T)=PEEK(1024+X+T):IFX(T)=42THEN86
85 NEXTT
86 PRINT"(CLR)(HOME)LOAD":CHR$(34):FORX=0T0T-1:POKE(1029+X),X(X):NEXT
87 PRINT"(HOME)"TAB(5+X):CHR$(34):" ",8,1"
88 POKE631,145:POKE632,145:POKE633,157:POKE634,157:POKE635,131:POKE198,5 :NEW
101 DATA162,8,32,198,255,32,180,255
109 DATA32,165,255,201,130,208,1,96
117 DATA201,194,208,1,96,206,32,208
125 DATA201,199,208,236,32,61,192,32
133 DATA165,255,201,199,208,229,230,251
141 DATA166,251,224,20,208,241,169,0
149 DATA234,234,234,141,32,208,141,33
157 DATA208,96,234,234,234,169,0,133
165 DATA251,238,32,208,96,255,0
166 DATA230,253,32,165,255,32,165,255,32,165,255,201,160,208,4,32,112
167 DATA192,96,201,0,208,7,32,112,192,96,234,234,206,32,208,32,71
168 DATA 171,76,79,192,234,234,169,144,32,71,171,169,42,32,71,171,169,13,32,71
169 DATA 171,96
170 DATA162,8,32,198,255,32,180,255,32
180 DATA165,255,232,224,4,208,248,96,194
    
```

BOOT 2

```

1 POKE55,72:POKE56,30
10 DIMX(16):OPEN 8,8,8,"$.S.R":POKE53280,2:POKE53281,2:POKE253,0:SYS 3203
11 PRINT"(CLR)(WHT)"
20 PRINTTAB(3):SYS 3073:IF PEEK(53281)=240ORPEEK(253)>45THEN CLOSE 8:GOTO 50
21 SYS3144:PRINT"(WHT)":IFPEEK(253)=23THENPRINT"(HOME)(DOWN)":TAB(22):
30 IFPEEK(253)>23 THEN PRINTTAB(22):
31 GOTO 20
50 SYS 62259:PRINT"(HOME)":TAB(23):"(WHT)(RVS) +SHIFT ":POKE53280,0:POKE53281,
0
60 FORT=1 T023:PRINTTAB(19):"(RED)(RVS) ":CHR$(T+64):" ":NEXT:PRINT"(HOME)"
61 FORT=1T023:PRINTTAB(0):"(RED)(RVS) ":CHR$(T+64):" ":NEXT
70 GET AS:IFAS="" THEN70
81 IF AS=CHR$(13) THENOPEN15,8,15,"I":CLOSE15:CLR:GOTO10
82 X=ASC(AS)-64:IFX>23THENX=X-105
83 X=X*40+3:IFX>963THENX=X+19-23*40
84 FORT=0T016:X(T)=PEEK(1024+X+T):IFX(T)=42THEN86
85 NEXTT
86 PRINT"(CLR)(HOME)LOAD":CHR$(34):FORX=0T0T-1:POKE(1029+X),X(X):NEXT
87 PRINT"(HOME)"TAB(5+X):CHR$(34):" ",8,1"
88 POKE631,145:POKE632,145:POKE633,157:POKE634,157:POKE635,131:POKE198,5
89 POKE55,0:POKE56,160:NEW
    
```

t(50) = matrice dei termini noti
 r = riga
 c = colonna
 i = indice secondo il quale avviene la riduzione della matrice

grammi è che nel primo le routine in linguaggio macchina le deve caricare dai data mentre il secondo le ha rilocate al termine del programma Basic con il quale sono salvate (abbassa perciò il top della memoria Basic).

Teorema Rouche Capelli (T.R.C.)

Variabili

x(50) = matrice per memorizzare le linee dove il denominatore è nullo e perciò deve essere sommata una seconda riga

a(50,50) = è la matrice quadrata dei coefficienti

Subroutine

1000 stampa il sistema
 1050 aspetta la pressione di un tasto (messo in a\$)

240 somma una riga a quelle che hanno denominatore zero

1-5 setta i colori, stampa il titolo.

10-62 routine di input; riempie la matrice dei coefficienti e dei termini noti al termine chiede una verifica (linee 60,61).

70-170 routine centrale riduce la matrice da quadrata a triangolare il metodo è il seguente: 81-122 divide una colonna per tutte le altre se il denominatore è 0 allora incrementa p e mette in x(50) riga dove vi è verificato. Al termine di questa routine salta a

240. 130-170 sottrae una riga a tutte le altre (per fare venire fuori gli zero nella riduzione della matrice).

171-230 routine per stampare i risultati. Se il coefficiente della prima incognita è 0 allora salta a 400 dove guarda il termine noto (se è = 0 è una identità, altrimenti il sistema non ha soluzioni).

240-390 cerca una riga dove nella colonna, secondo la quale avviene la riduzione, il valore non sia 0 (linee 270-290) somma in fine questa riga a tutte quelle che avevano denominatore 0 (linee 300-390). Se non riesce a trovare un valore che non sia nullo fa stampare «infinite soluzioni» (linea 290) al termine azzera p.

Boot

Variabili

\$00FD numero programmi letti
 x = puntatore programma scelto
 x(16) = nome del file

Routine (in l.m.)

Subroutine 1 C000-C046: questa routine ha il compito di leggere gli elementi del file affinché non trovi un valore \$82 o \$C2 che precedono nella directory i nomi dei programmi e dei programmi protetti. Per una più dettagliata descrizione si veda il disassemblato.

Subroutine 2 C047-C04F: questa routine ha il compito di leggere il nome del file e scriverlo sullo schermo. Inoltre al termine stampa un asterisco nero per permettere al programma Basic di determinare la fine del nome.

Subroutine 3 C080-C090: questa subroutine prende 251 elementi del file, in pratica legge tutta la BAM dove non sono scritti i nomi dei programmi.

1-10 lettura data (solo in boot 1), apre il file setta i colori e salta alla routine in l.m. numero 3.

20 stampa 3 spazi salta alla subroutine numero 1 e al termine controlla che il file non sia al termine o che il numero di programmi sia massimo.

21 salta alla subroutine 2, stampa inoltre lo spazio per i nomi che devono andare in seconda colonna.

50 risetta i file e i colori.

60-61 sono due routine che stampano le colonne con le lettere.

70-81 attende la pressione di un tasto se è il return inizializza il disco, pulisce le variabili e ritorna al 10.

82-83 calcola il valore di x (= prima locazione dello schermo del nome file scelto).

84-85 memorizza nella matrice x(16) i valori nelle locazioni affinché non trovi l'asterisco.

86-88 lancia il programma mettendo i dovuti valori nel buffer di tastiera.

```

ROUTINE USATE IN BOOT

PRIMA ROUTINE
.. C000 A2 08 LDX #08
.. C002 20 06 FF JSR #FFC6 ;apre un canale di input
.. C005 20 04 FF JSR #FFB4 ;segnala di prepararsi a trasmettere dati
.. C008 20 05 FF JSR #FFB5 ;prende un dato e lo mette nell' accumulatore
.. C00B C9 82 CMP #82 ;lo compara con #82 (=Programma)
.. C00D 00 01 BNE #C010 ;se uguale ritorna al basic
.. C00F 50 RTS
.. C010 C9 C2 CMP #C2 ;compara con #c2 (=Programma Protetto)
.. C012 00 01 BNE #C015 ;se uguale torna al basic
.. C014 50 RTS
.. C015 CE 20 D0 DEC #0020 ;cambia il colore al bordo
.. C018 C9 C7 CMP #C7 ;compara con #C7 carattere di fine file
.. C01A 00 EC BNE #C008 ;se diverso Prende un altro carattere.
.. C01C 20 3D C0 JSR #C000 ;subroutine Per azzerrare #FB
.. C01F 20 A5 FF JSR #FFA5 ;Prende un altro carattere
.. C022 C9 C7 CMP #C7 ;lo compara con #C7
.. C024 00 E5 BNE #C008 ;se Per 20 volte e' uguale
.. C026 E6 FB INC #FB
.. C028 A6 FB LDX #FB
.. C02A E0 14 CPX #14
.. C02C 00 F1 BNE #C01F
.. C02E 00 90 LDA #00
.. C030 EA NOP
.. C031 EA NOP
.. C032 EA NOP
.. C033 8D 20 D0 STA #D020 ;setta i colori e
.. C035 8D 21 D0 STA #D021
.. C037 00 00 NOP
.. C039 EA NOP
.. C03B EA NOP
.. C03C EA NOP
.. C03D A9 00 LDA #00
.. C03F 05 FB STA #FB
.. C041 EE 20 D0 INC #D020
.. C044 50 RTS
.. C045 FF ???
.. C046 00 BRK

SECONDA ROUTINE
.. C047 E5 FD LHC #FD ;FD=numero Programmi
.. C049 20 A5 FF JSR #FFA5 ;legge la traccia
.. C04C 20 A5 FF JSR #FFA5 ;e settore di inizio Programma
.. C04F 20 A5 FF JSR #FFA5 ;Prende un altro carattere
.. C052 C9 A0 CMP #A0 ;lo compara con #A0 (=fine nome)
.. C054 00 04 BNE #C05A
.. C056 20 70 C0 JSR #C070 ;se uguale salta a #C070
.. C059 50 RTS ;ritorno al basic
.. C05A C9 00 CMP #00 ;compara con #00 (=fine nome)
.. C05C 00 07 BNE #C055
.. C05E 20 70 C0 JSR #C070 ;se uguale salta a #C070
.. C061 50 RTS ;ritorno al basic
.. C062 EA NOP
.. C063 EA NOP
.. C064 EA NOP
.. C065 CE 20 D0 DEC #D020 ;cambia colore al bordo
.. C068 20 47 AB JSR #AB47 ;stampa il carattere
.. C06B EA 4F C0 JMP #C04F ;ritorna a Prendere un altro carattere
.. C06E EA NOP
.. C06F EA NOP
.. C070 A9 90 LDA #90 ;mette il colore nero
.. C072 20 47 AB JSR #AB47 ;lo stampa
.. C074 A9 2A LDA #2A ;mette nell' accumulatore '*'
.. C076 20 47 AB JSR #AB47 ;lo stampa
.. C078 A9 8D LDA #8D
.. C07C 20 47 AB JSR #AB47 ;stampa un ritorno di carrello
.. C07F 50 RTS ;ritorna dalla subroutine

TERZA ROUTINE
.. C080 A2 08 LDX #08
.. C082 20 06 FF JSR #FFC6 ;apre un canale di input
.. C085 20 04 FF JSR #FFB4 ;segnala di prepararsi a trasmettere dati
.. C088 20 05 FF JSR #FFB5 ;prende
.. C08B E6 04 INX #04 ;per 251 volte un carattere
.. C08E 00 F8 BNE #C088
.. C090 50 RTS ;ritorna albasic
    
```



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 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	Note
APPLE II				
DA2/00	Shape Tablet	22	15000	:
DA2/01	Motomuro	26	15000	:
DA2/02	DEBUG	28	15000	:
DA2/03	EDIT + INPUT	29	15000	:
DA2/04	Basic modulare	34	15000	:
DA2/05	ANFA 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	:
DA2/13	Scroll 300 righe	46	15000	:
DA2/14	Assembler in Basic	50	15000	:
DA2/15	G-Basic II	53	15000	:
DA2/16	Disk Editor	54	15000	:
DA2/17	Latino	57	15000	:
DA2/18	Battaglia	61	15000	:
COMMODORE 128				
C28/01	MNCalc	53	17000	:
C28/03	Mega Bank 128	56	17000	:
D28/01	MNCalc	53	15000	:
D28/02	Hardcopy 128	55	15000	:
D28/03	Sheetit	57	15000	:
D28/04	Star Quest	58	15000	:
D28/05	Family Budget	60	15000	:
D28/06	La casa stregata	61	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	Paroliamo	39	17000	:
C64/14	Scarabeo	40	17000	:
C64/15	Magazzino	41	17000	:
C64/16	Rubrica	44	17000	:
C64/17	World	45	17000	:
C64/18	P. J. T. Basic	46	17000	:
C64/19	Sistema Enalotto	47	17000	:
C64/20	Simulat.reti logiche	48	17000	:
C64/21	RITTY	48	17000	:
C64/22	Nescola	49	17000	:
C64/23	Othello	51	17000	:
C64/24	Voters	51	17000	:
C64/25	Flashtape	50/51	17000	:
C64/26	Cross Reference	53	17000	:
C64/27	Flib	54	17000	:
C64/28	Boz's Adventure	57	17000	:
D64/01	Spreadsheet	34	15000	:
D64/02	ADP Basic da 35 a	39	15000	:
D64/03	Wordprocessor	38	15000	:
D64/04	Paroliamo	39	15000	:
D64/05	Data base Galileo	40/41	15000	:
D64/06	Magazzino	41	15000	:
D64/07	Gestione biblioteca	46	15000	:
D64/08	P. J. T. Basic	46	15000	:

Codice	Titolo programma	MC n.	Prezzo	Note
D64/09	Simulat.reti logiche	48	15000	:
D64/10	Archprog	50	15000	:
D64/11	Anno Domini	57	15000	:
D64/12	The Disk Editor	54/6/7	15000	:
D64/13	Boz's Adventure	57	15000	:
D64/14	Link-64	57	30000	:
D64/15	New Char 2.2	58	15000	:
D64/16	Music 64	59	15000	:
D64/17	TRX-MEM	59	15000	:
D64/18	VCS + VBasic	60	15000	:
MSX				
CMX/01	Sound editor	42	17000	:
CMX/02	VP Reporter	43	30000	:
CMX/03	Foresta maledetta	44	17000	:
CMX/04	Monitor disassembler	45	17000	:
CMX/05	Video Art	46	17000	:
CMX/06	Othello	47	17000	:
CMX/07	Joe's Chicken	48	17000	:
CMX/08	Planet Hunter	49	17000	:
CMX/09	Dune	50	17000	:
CMX/10	Ramboman	51	17000	:
CMX/11	Worm	52	17000	:
CMX/12	Controparola	53	17000	:
CMX/13	Shape Editor	54	17000	:
CMX/14	Labirinto 3D	55	17000	:
CMX/15	Fred	56	17000	:
CMX/16	Il tesoro dei pirati	57	17000	:
CMX/17	Omino	58	17000	:
CMX/18	Toto 13	60	17000	:
CMX/19	Painter	62	17000	:
DMX/01	Toto 13	60	15000	:
DMX/02	Painter	62	15000	:
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
CSS/19	42 colonne	46	17000	:
CSS/20	3D Pacman	46	17000	48 K RAM
CSS/21	Forza 4	47	17000	48 K RAM
CSS/22	ZX Editor	47	17000	48 K RAM
CSS/23	Va-Tor	48	17000	48 K RAM
CSS/24	Meta	49	17000	:
CSS/25	Graphic Macro Lang.	49	17000	:
CSS/26	Super Monitor	50	17000	48 K RAM
CSS/27	Database 64 colonne	50	17000	48 K RAM
CSS/28	MC Basic	52	17000	48 K RAM
CSS/29	Spectrum LOGO	53	17000	:
CSS/30	Disassembler	54	17000	48 K RAM
CSS/31	Istogrammi	55	17000	48 K RAM
CSS/32	Finestre	56	17000	48 K RAM

Note:
L'iniziale del codice e' C per le cassette, D per i minifloppy