

Campo elettrico

di Vincenzo Calamia - Alcamo

Il programma simula il campo elettrico generato da «n» cariche puntiformi comunque disposte sul piano.

Le linee di forza vengono tracciate per punti sullo schermo grafico a 40 colonne, in scala, ed ogni carica viene rappresentata come un cerchietto contenente il segno. Lo schema di calcolo è abbastanza semplice: partendo da un certo numero di punti posti ad una distanza prefissata da ogni carica, il programma calcola il valore del campo elettrico come somma vettoriale dei campi dovuti a tutte le cariche presenti nello spazio, spostandosi quindi di 5 pixel lungo la direzione del campo e ripetendo l'operazione; ovviamente il programma deve verificare ogni volta se il punto considerato è prossimo ai limiti del campo, se è vicino ad un'altra carica e se, completata una linea di forza, deve tracciarne altre con origine nella stessa carica o passare ad un'altra carica.

La procedura seguita dal programma comporta alcune inevitabili approssimazioni:

1) Il punto della linea di forza successivo a quello corrente viene infatti individuato lungo la tangente alla linea di forza e non lungo la linea stessa; l'approssimazione sarebbe migliore se si riducesse la distanza tra due punti successivi, ma questo accorgimento avrebbe come conseguenza una diminuzione della velocità di esecuzione ed un aspetto più frastagliato della linea di forza.

2) In alcuni casi si può notare una sovrapposizione o una intersezione tra linee di forza, circostanza che teoricamente non dovrebbe mai verificarsi.

La distanza di 5 pixel tra un punto ed il successivo rappresenta il valore più adatto agli scopi del programma.

Teoricamente il programma può simulare il campo generato da un numero qualsiasi di cariche ma ogni carica in più rende il programma molto più lento e quindi è opportuno limitarne il numero.

Ultimata la rappresentazione del campo elettrico, viene visualizzata, nella parte inferiore dello schermo, una linea che propone le opzioni possibili premendo i tasti indicati: «A» per elaborare un altro esempio; «C» per calcolare il campo elettrico in un punto qualsiasi dello spazio a meno della costante del mezzo «k»; «P» per stampare lo schermo gra-

fico; «E» per uscire dal programma.

Con l'opzione «C» il programma calcola il modulo del C.E. in un punto qualsiasi e, se il punto ricade entro lo schermo, viene tracciato anche il versore del campo.

Con l'opzione «P», che utilizza una delle tante routine di pubblico dominio in L.M., viene effettuata una copia su carta dello schermo grafico, ultimata la quale, vengono trascritti i moduli e le coordinate delle singole cariche. Poiché il programma è stato elaborato per un output su Star NL-10, la scala del grafico su carta potrebbe essere diversa da quella rilevabile su altre stampanti per le quali peraltro potrebbe essere necessario modificare qualche codice.

DESCRIZIONE DELLE LINEE DI PROGRAMMA

10-190
Schermata di presentazione.
200-280
Descrizione sintetica delle funzioni svolte dal programma.
285-420
Input dati.
430-460
Possibilità di attivazione modo fast con la conseguente disattivazione dello schermo grafico fino al completamento delle operazioni.
470
Disabilita il tasto run/stop.
480
Inizio calcoli.
530-560
Controllo posizione cariche e definizione della scala del grafico su schermo.
590-750
Posizionamento cariche sullo schermo grafico e scelta del numero delle linee di forza.
760-980
Calcolo del campo elettrico in ciascun punto dello spazio e verifiche della posizione del punto rispetto ai limiti del campo ed alla posizione delle cariche.
990-1080
Scelta delle opzioni; la linea 1080 verifica lo stato della stampante.
1090-1230
Predisposizione dello schermo grafico per la stampa e operazioni di output su stampante.
1240-1520
Subroutine di calcolo.
1530
Routine di screencopy.

VARIABILI PRINCIPALI

-nq% numero cariche.
-q() vettore intensità cariche.
-xq(), yq() vettori coordinate logiche cariche.
-oq() vettore ordinata di visualizzazione.
-sr fattore di scala di rappresentazione.
-sw, pw scala video e stampante.
-nl% numero linee di forza per ogni carica.
-xp, yp coordinate punto iniziale linea di forza.
-xa, ya coordinate punto corrente della linea di forza.
-xb, yb coordinate punto successivo al precedente.
-e c.e. della i-esima carica nel punto corrente.
-tt c.e. risultante

Tasti speciali 128

di Davide Rinnone e Gaetano Minardi Niscemi (CL)

Chi programma in modo 128, sentirà sicuramente la mancanza (in modo 64) di tutti i tasti aggiunti dalla Commodore per facilitare la programmazione. Infatti per alcuni programmi scritti per il C64 (quali quelli di contabilità ed altri), non si può usufruire del tastierino numerico, che in alcune situazioni risulta insostituibile. Per non parlare poi delle lunghe sequenze di numeri che seguono le frasi DATA, dove per digitare più velocemente e correttamente i dati si sente il bisogno del tastierino numerico. Ma come tutti sanno, «mamma Commodore» non ha provveduto il «buon C64» del tastierino numerico, per assicurare la perfetta compatibilità con il vecchio C64. Ma da oggi i possessori del Commodore 128 possono usufruire del tastierino numerico pure in modo 64 grazie a questa routine, scritta interamente in L.M., che abilita oltre al tastierino anche alcuni tasti «speciali», quali i quattro tasti cursore posti accanto ai tasti funzione e il tasto NO SCROLL (per fermare il listato). Inoltre abbiamo abilitato la sequenza FAST/SLOW per fare girare pure il C64 a 2 MHz. Sì, proprio così, con il vantaggio di fare eseguire alcuni programmi in Basic al-

Campo elettrico

```

800 GOSUB890
810 IFF1=1THEN870
820 XB=XA+5*SR*TX/TT:YB=YA+5*SR*TY/TT:TX=0:TY=0:DRAW1,XA,YA
830 XC=ABS(XD-XB):YC=ABS(YD-YB):IFXC<3*SRANDYC<3*SRTHEN880
840 XD=XA-XB:YD=YA-YB:IFXA<10*SRORXA>310*SRGOTO870
850 YD=YA-YB:IFYA<15*SRORYA>195*SRGOTO870
860 GOTO880
870 F1=0
880 NEXTX:NEXTY
890 FORJ=1TONUM:A=(XA-XQ(J)):B=(YA-YQ(J)):D2=(A^2+B^2):D=SQR(D2)
900 IFSSTHEN920
910 F1=1:GOTO890
920 E=Q(J)/D2:TX=TX+E*A/D:TY=TY+E*B/D:TT=SQR(TX^2+TY^2)
930 IFQ(1)>0THEN950
940 TT=TT
950 NEXTJ
960 IFTT<>0THEN980
970 F1=1
980 IFL=NO%ANDK>0THEN1000
990 RETURN
1000 SLOW:CHAR1,0,24,*(SWLC)A=ALTRO ESEMPIO*C=CALCOLI*P=STAMPA*E=
END
1010 GETA$:IFAS=""THEN1010
1020 IFAS="P"THEN1090
1030 IFAS="E"THEN1080
1040 IFAS="A"THEN1070
1050 IFAS="C"THEN1390
1060 GOTO1010
1070 GRAPHIC0:GOTO300
1080 GRAPHIC0:CLR:PRINTCHR$(142)"(CLR)":POKE808,110:END
1090 IFSSTHEN1090 ELSEGOTO1120
1100 CHAR1,0,24,*(SWLC)ACCENDI LA STAMPANTE E PREMI UN TASTO*
1110 GETA$:IFAS=""THEN1110
1120 CHAR1,0,24,*(SWLC)
1130 CHAR1,0,1,*(SWLC)
1140 OPEN4,4:PRINT*4,CHR$(27):CHR$(108):CHR$(13):CLOSE4:IFSSTHEN
1100
1150 FAST:SYSS120:SLOW
1160 CHAR1,0,1,SW5,0:OPEN4,4
1170 PRINT*4,CHR$(17):" CARICA N. MODULO [C] X [CM] Y [CM] S
CALA GRAFICO"
1180 PRINT*4,
1190 FORI=1TONUM:PRINT*4,USING"*****":I:
1200 PRINT*4,USING"*****"*(Q(I)):
1210 PRINT*4,USING"*****"*(XQ(I)):
1220 PRINT*4,USING"*****"*(YQ(I)):
1230 IFF1=1THENPRINT*4,PWS:
1240 PRINT*4,*(NEXTJ):CLOSE4
1250 CHAR1,0,1,SW5,0:GOTO1000
1260 GRAPHIC0:PRINT"(CLR)PUNTO TROPPO VICINO AD UNA CARICA!!!!!!"
1270 PRINT"PREMI UN TASTO E RIBATTI LE COORDINATE"
1280 GETA$:IFAS=""THEN1280
1290 GRAPHIC0:PRINT"(CLR)(SWLC)(RVS)CALCOLO DEL CAMPO ELETTRICO
IN UN PUNTO(OFF)"
1300 DB=20*SR:DHS=STR$(DB):RNS="(RVS)":RFS="CM(OFF)":WXS=RNS*X
*DB+RFS:WYS=RNS*Y*DB+RFS
1310 PRINT"(DOWN)(DOWN)(DOWN)(DOWN)COORDINATE PUNTO:WXS:INPUTXP:
KD=XP
1320 PRINT"
1330 GRAPHIC0:CHAR1,0,1,V25,0:CHAR1,0,0,*(VETTORE CAMPO ELETTR
ICO IN UN PUNTO**):SCALE1,320*SR,200*SR
1340 TX=0:TY=0
1350 FORJ=1TONUM:A=(XD-XQ(J)):B=(YD-YQ(J)):D2=A^2+B^2:D=SQR(D2)
1360 IFD<4*SRTHEN1260
1370 E=Q(J)/D2:TX=TX+E*A/D:TY=TY+E*B/D:TT=SQR(TX^2+TY^2):NEXTJ
1380 IFTT<>0THEN1500
1390 XX=3*SR*TX/TT:YY=3*SR*TY/TT:XP=XD+4*SQR(2)*XX:YF=YD+4*SQR(2)
*YY
1400 XG=XF-XX+YY:YG=YF-YY-XX:XH=XF-XX-YY:YH=YF-YY+XX
1410 IFXD<5*SRORXD>315*SRTHEN1500
1420 IFXF<5*SRORXF>315*SRTHEN1500
1430 IFXG<5*SRORXG>315*SRTHEN1500
1440 IFXH<5*SRORXH>315*SRTHEN1500
1450 IFYD<5*SRORYD>190*SRTHEN1500
1460 IFYF<5*SRORYF>190*SRTHEN1500
1470 IFYG<5*SRORYG>190*SRTHEN1500
1480 IFYH<5*SRORYH>190*SRTHEN1500
1490 DRAW1,XD,YD,XP,YF,XT,XG,YG,DRAW1,XF,YF,TOXH,YH
1500 TT=TT*(1+4*(TT-SQR(TT)*INT((ABS(TT)*10^4)+5)/10^4):TT=SQR(TT)
XP=STR$(XF):YF=STR$(YF)
1510 M1=(SWLC)IN P1:V1S="(M2S)"E="(M3S)"K(V/M):V1S="
V2S+V1S+V1S+V1S+V1S
1520 MMS=M1S+XP+V1S+YF+M2S+TT+M3S
1530 CHAR1,0,1,MMS,0
1540 SLEEP5:GOTO1000
1550 DATA 76,217,20,32,33,20,169,96,162,4,160,0
1560 DATA 32,186,255,169,0,32,189,255,32,192,255,162
1570 DATA 96,76,201,255,169,96,32,204,255,169,96,76
1580 DATA 195,255,169,15,32,210,255,169,13,32,210,255
1590 DATA 76,28,20,169,0,141,73,21,173,75,21,240
1600 DATA 16,201,1,240,5,238,73,21,24,96,173,74
1610 DATA 21,201,64,176,244,173,77,21,208,239,173,76
1620 DATA 21,201,200,176,232,173,76,21,41,7,133,251
1630 DATA 173,76,21,74,74,41,254,168,185,135,20,56
1640 DATA 229,251,133,251,185,136,20,24,109,75,21,133
1650 DATA 252,173,74,21,41,648,169,173,74,21,41,7
1660 DATA 170,24,96,7,62,199,60,135,59,71,58,7
1670 DATA 57,199,55,135,54,71,53,7,52,199,50,135
1680 DATA 49,71,48,7,47,199,45,135,44,71,43,7
1690 DATA 42,199,40,135,39,71,38,7,37,199,35,135
1700 DATA 34,71,33,7,32,234,32,0,32,32,139,32
1710 DATA 51,20,173,73,21,208,8,189,209,20,49,251
1720 DATA 56,208,1,24,96,128,64,32,16,8,4,2
1730 DATA 1,32,3,20,169,8,32,210,255,169,199,141
1740 DATA 78,21,169,0,141,74,21,141,75,21,141,77
1750 DATA 21,162,7,134,253,173,78,21,141,76,21,32
1760 DATA 191,20,110,79,21,206,76,21,198,253,208,243
1770 DATA 110,79,21,173,79,21,9,128,32,210,255,238
1780 DATA 74,21,208,3,238,75,21,173,75,21,240,209
1790 DATA 173,74,21,201,64,208,202,169,13,32,210,255
1800 DATA 173,78,21,56,233,7,141,78,21,201,252,208
1810 DATA 173,169,15,32,210,255,169,13,32,210,255,76
1820 DATA 28,20,76,38,20,0

```


(\$DC00) e 56321 (\$DC01). Già! Sono gli stessi registri che servono per la lettura delle porte dei joystick. Ed è per questo motivo che quando inserite un joystick in porta 1 e lo muovete, sullo schermo appaiono alcuni numeri e simboli che si potrebbe pensare siano senza senso, in realtà inserendo un joystick in porta 1, lasciando libera la porta 2, e muovendolo viene attivata la linea 8 della quale fanno parte i tasti ←; 1; 2; CTRL; RUN/STOP; COMMODORE; Q e SPACE. Chiamiamo meglio il concetto di linee. Il C64 ha una tastiera composta in totale da 66 tasti. Tuttavia questo numero si riduce a 64 tasti, escludendo il RESTORE (collegato direttamente al 6510) e lo SHIFT LOCK (la cui funzione è quella di bloccare lo SHIFT sinistro). Ma i 64 tasti devono essere interamente scanditi da due soli registri di 8 bit. Affinché ciò sia possibile è necessario dividere la tastiera in 8 linee, composte ciascuna da 8 tasti, in modo da potere assegnare le 8 linee ad un registro (\$DC00 nel nostro caso) e gli 8 tasti di ogni linea ad un altro registro (\$DC01). In questo modo selezionando una delle 8 linee disponibili nel registro \$DC00 potremo andare a leggere nel registro \$DC01 lo stato degli 8 tasti appartenenti a tale linea. Bisogna però stare attenti al modo di attivare le linee in quanto questi due registri funzionano al contrario. In altre parole, se un bit è posto a 1, quindi attivato, la linea corrispondente sarà disattivata; al contrario se tale bit è azzerato la linea corrispondente sarà attivata. Lo stesso vale per il registro \$DC01 e quindi se dopo aver attivato una linea tale registro ci restituisce \$FF, cioè tutti i bit posti ad uno, significa che nella linea selezionata nessun tasto è stato premuto. In tal modo il C64 riesce a gestire tutta la tastiera tramite i due registri \$DC00 e \$DC01 con i quali è possibile ottenere 64 combinazioni. Questo numero di combinazioni però è del tutto insufficiente se si usa il C128, in quanto in questa modalità si hanno 88 tasti a

disposizione. Pertanto ci sono 24 tasti in più che qualcuno dovrà pur gestire. Dopo avere studiato la routine \$FF9F, cioè SCNKEY che effettua la scansione di tastiera, abbiamo notato che il registro in più esiste, ma non appartiene a nessuno dei due CIA, ma al 6566/6567 alias VIC II. Infatti il VIC II del C64 contenuto nel C128 ha due registri in più (47 e 48) rispetto a quelli contenuti nel normale C64. Il primo di questi due registri viene utilizzato per la scansione di tastiera in quanto controlla (tramite i suoi bit meno significativi) le linee 9-11 per un totale di 24 combinazioni. L'altro viene invece utilizzato per settare il funzionamento a 2 MHz o ad 1 MHz: noi li abbiamo utilizzati entrambi per realizzare il programma. Per implementare FAST e SLOW è stato modificato il vettore \$0308-\$0309 facendolo puntare a \$C157 dove avviene un controllo del carattere corrente. Se non si tratta né di F né di S si salta alla routine di esecuzione comando corrente, altrimenti si fa un successivo controllo per le lettere AST e LOW, se sono uguali si eseguono le due routine, altrimenti il controllo va all'interprete. Per fare funzionare il tastierino numerico, i tasti cursore separati e il tasto NO/SCROLL abbiamo deviato il vettore di interrupt alla nostra routine che è perfettamente uguale all'originale tranne che per la chiamata a SCNKEY, in quanto prima veniva effettuata una JSR\$EA87 mentre adesso viene effettuata una JSR\$C000. Da questo indirizzo inizia la nuova SCNKEY che, se nessun tasto «normale» è premuto, salta alla routine per la scansione dei tasti speciali; altrimenti esegue la normale SCNKEY, che non era necessario riscrivere interamente, ma che abbiamo riscritto per completezza e per seguire meglio il flusso del programma. Come già detto prima funziona il NO/SCROLL che serve per bloccare il listing. Al contrario del C128 per ripartire è però necessario ripremere NO/SCROLL.

Trasformazione di numeri in lettere

di Vincenzo Calamia - Alcamo

Il programma realizza una semplice routine che trasforma in lettere i numeri interi < 10¹⁵; è stato creato per essere inserito in un programma di fatturazione su un modello di fattura che prevede che l'importo totale sia trascritto anche in lettere.

Il programma può essere facilmente tradotto per altri sistemi e può essere utilizzato, come routine, in tutti i casi in cui sia necessaria la trascrizione in lettere di un numero espresso in cifre: elaborazione di fatture, parcelle, ricevute, cambiali etc.

Il programma prende inizialmente in considerazione le prime tre cifre di destra, spostandosi poi verso sinistra di tre cifre in tre cifre; ogni pacchetto di tre cifre viene analizzato con la stessa subroutine che provvede anche alla sistemazione delle particolarità linguistiche che si incontrano nella «traduzione».

Descrizione delle linee di programma

10-60

Schermata di presentazione.

70-90

Dimensionamento e trasferimento nel vettore u\$(i) delle unità, decine, centinaia, etc. espresse in lettere nelle linee DATA 420-460

100-180

Input numero e verifiche iniziali.

190-410

Analisi del numero e traduzione in lettere.

MC

Trasformazione di numeri in lettere

```

10 PRINT (CLR) (WHT) (DOWN) (DOWN) (DOWN) (DOWN) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (R
20 PRINT (DOWN) (DOWN) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (DI NUM
ERI INTERI <10^15^
30 PRINT (DOWN) (DOWN) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT)
(RIGHT) (RIGHT) (RIGHT) (RIGHT) IN LETTERE^
40 PRINT (DOWN) (DOWN) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT)
(RIGHT) (RIGHT) (RIGHT) (RIGHT) PER C-128^
50 PRINT (WHT) (DOWN) (DOWN) (DOWN) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (
RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) DI V. CALAMIA^
60 SLEEP 3
70 CLR:PRINT (CLR) :DIMUS(40)
80 FOR I=1 TO 40:READ US(I):NEXT
90 FOR J=1 TO 10:READ MS(J):NEXT
100 NLS="" : CFS="" : INPUT (CLR) NUMERO INTERO : NS=N-VAL(NS) : X=LEN(NS) : GOTO 140
110 PRINT (YEL) (DOWN) (DOWN) (DOWN) (DOWN) ALTRO NUMERO? (S=1/N=0)
120 GETKEYS : IFTS="S" OR TS="1" THEN 100
130 IFTS="N" OR TS="0" THEN END : ELSE GOTO 120
140 IF N<1 ELSE 190 : ELSE BEGIN
150 PRINT (CLR) (DOWN) (DOWN) (DOWN) (DOWN) : FOR L=1 TO 3
160 PRINT (UP) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT)
      : SLEEP 4
165 SOUND 1,8000,10
170 PRINT (UP) (BLK) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) (RIGHT) ***NUMERO TR
PO GRANDE*** : SLEEP 1:NEXT:BEND:GOTO 110
180 PRINT (DOWN) (WHT) : IF N<0 THEN 190 : ELSE PRINT (DOWN) ZERO^
190 FOR I=1 TO 5

```

```

200 DD=0:Y=3*I:J=1+S:CMS=RIGHT$(NS,3):GOSUB 270
210 IF VAL(CMS)=0 THEN NLS=CFS+NLS:GOTO 240
220 IF VAL(CMS)=1 THEN NLS=CFS+MS(J)+NLS:GOTO 240
230 NLS=CFS+MS(I)+NLS
240 IF X-Y>0 THEN 250:ELSE PRINT NLS:GOTO 110
250 CSS=LEFT$(NS,X-Y):NS=CSS
260 IF VAL(CSS)<>1 THEN NEXT:ELSE PRINT MS(J+1)+NLS:GOTO 110
270 CM=VAL(CMS):IF CM=1 AND Y=6 THEN CFS="" : RETURN
280 IF CM>19 THEN 290:ELSE CFS=US(CM):RETURN
290 IF CM>99 THEN 300:ELSE GOTO 320
300 CCS=LEFT$(CMS,1):CC=VAL(CCS)
310 IF CC=1 THEN NLS="" : ELSE CLS=US(CC)
320 DUS=RIGHT$(CMS,2):DU=VAL(DUS)
330 IF DU>19 THEN 340:ELSE CFS=US(DU):GOTO 390
340 DDS=LEFT$(DUS,1):DD=VAL(DDS)
350 UUS=RIGHT$(DUS,1):UU=VAL(UUS)
360 IF UU=10 OR UU=8 THEN DLS=US(DD+30):ELSE DLS=US(DD+20)
370 IF UU=0 THEN NLS="" : ELSE UUS=US(UU)
380 CFS=DLS+UUS
390 IF CM>99 THEN 400:ELSE RETURN
400 IF DD=8 THEN CCS="" : ELSE CCS="CENTO"
410 CFS=CLS+CCS+CFS:RETURN
420 DATA UNO,DUE,TRE,QUATTRO,CINQUE,SEI,SETTE,OTTO,NOVE,DIECI,UNDICI,DOBDICI
430 DATA TREDICI,QUATTORDICI,QUINDICI,SEDICI,DISETTE,DIOTTO,DICIANNOV
E,
440 DATA ,VENTI,TRENTA,QUARANTA,CINQUANTA,SESSANTA,SETTANTA,OTTANTA,NOVANTA
,
450 DATA ,VENT,TRENT,QUARANT,CINQUANT,SESSANT,SETTANT,OTTANT,NOVANT,
460 DATA ,MILA,MILIONI,MILIARDI,BILIONI,,MILLE,UNMILIONE,UNMILIARDO,UNBILIO
NE

```



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
APPLE II			
DA2/06	Miniset + LevaDOS	37	15000
DA2/07	27 programmi grafici	38	30000
DA2/08	Adventure Editor	38	15000
DA2/09	Animazione Funzioni	42	15000
DA2/12	Routine Grafiche Estese	44	15000
DA2/13	Scroll 300 linee	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
DA2/19	Catalogo	64	15000
DA2/20	Apple Puzzle II	65	15000
DA2/21	Precisione Multipla	66	15000
DA2/22	Sistema 2 + Toto 5.3 IIGS	68	15000
DA2/23	Operazione Apokalypsis	71	30000
DA2/24	Classifiche di Formula 1	72	15000
DA2/25	Programmabile RPN	73	15000
DA2/26	Supercircle + Poigonalie	74	15000
DA2/27	Hard Copy OKI 83/A	76	15000
DA2/28	ProDOS Utility	77	15000
DA2/29	Modulo Base	78	15000

Codice	Titolo Programma	MC n.	Prezzo
COMMODORE AMIGA			
DAM/01	F-15	63	15000
DAM/02	Gest. liste 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

Codice	Titolo Programma	MC n.	Prezzo
MS-DOS			
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	Bioritmi + 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

Codice	Titolo Programma	MC n.	Prezzo
COMMODORE 128			
C28/01	MMCalc	53	17000
C28/02	Mega Bank 128	56	17000
D28/01	MMCalc	53	15000
D28/02	Hardcopy 128	55	15000
D28/03	Shell	57	15000
D28/04	Star Guest	58	15000
D28/05	Family Budget	60	15000
D28/06	La Casa Stregosa	61	15000
D28/07	Strutture 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
D28/12	Disegna Circuiti	68	15000
D28/13	Mark's Data Base	70	15000
D28/14	Label Disk + Disk Editor + Dem DOS	71	15000
D28/15	Pulldown 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

Codice	Titolo Programma	MC n.	Prezzo
ATARI ST			
DST/01	Virus Killer	74	15000
DST/02	Mandelbrot + Proiez. Ort. + Bilancio	78	15000
COMMODORE 64			
C64/06	Bilancio Familiare	35	17000
C64/19	Sistema Enalotto	47	17000
C64/21	RTTY	48	17000
C64/22	Mescola	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	Fib	54	17000
C64/28	Boz's Adventure	57	17000

D64/11	Anno Domini	57	15000
D64/12	The Disk Editor	57	15000
D64/13	Boz's Adventure	54/6/7	15000
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	Archiplus	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
D64/30	TWS Basic	70	15000
D64/31	Trucchi e Routine per programmatori	71	15000
D64/32	Flow-Chart + Flower's Love	73	15000
D64/33	Sprite Editor	76	15000
D64/34	Portfollio 64 + Elim. bordi schermo	77	15000
D64/35	Alfabeto Morse + Locare + Menu/Driver	78	15000

Codice	Titolo Programma	MC n.	Prezzo
MSX			
CMX/04	Monitor Disassembler	45	17000
CMX/05	Video Art	46	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	Omno	58	17000
CMX/18	Toto 13	60	17000
CMX/19	Painter	62	17000
CMX/20	MSX Bank	63	17000
CMX/21	Grafica 3D + Hard Copy	65	17000
CMX/22	Mini dBase MSX	71	17000
CMX/23	Elenco Clienti	74	17000

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	Compiler 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	RGBCAD	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

Nota:
l'iniziale del codice è C per le cassette, D per i floppy.