

di Francesco Petroni

## Il disegno architettonico

Nel numero 20 di MCmicrocomputer (giugno 83) abbiamo trattato l'argomento Archivio Dati per Grafica di tipo Edile Architettonico. Lo riprendiamo in quanto rappresenta un'area applicativa di largo interesse pratico e in quanto è una materia nota a tutti, e oltretutto è relativamente facile da trattare essendo basata su una grafica di tipo bidimensionale e non legata a complessi problemi di matematica.

Riprendiamo l'argomento anche perché oggi la diffusione dello standard IBM e Olivetti mette a disposizione di questa categoria di Personal Computer prestazioni grafiche più spinte e con le quali, e lo dimostrano le illustrazioni a corredo dell'articolo, si ottengono risultati notevoli che vale la pena valutare.

La produzione di disegni architettonici tramite computer rappresenta una problematica la cui unica complessità risiede nell'organizzazione logica dei dati. Mentre ad esempio nel disegno di una superficie nello spazio gli unici dati in gioco sono la formula della superficie e i parametri di visualizzazione (coordinate del punto di osservazione, dello schermo, ecc.) e questo semplifica l'organizzazione del programma anche se le formule matematiche usate sono molto complesse, al contrario se si vuol disegnare una pianta architettonica di un edificio i dati sono moltissimi e quindi bisogna soprattutto creare dei criteri logici di raggruppamento e di organizzazione.

Detto in altre parole si sa esattamente cosa disegnare (la piantina dell'edificio), si sa esattamente quali sono gli strumenti con i quali disegnare (le istruzioni grafiche del Basic), l'unico problema è come collegare, nella maniera più brillante ed economica, le due cose.

Va poi separata la fase visualizzazione dei dati dalla fase caricamento dei dati. Confesso che mi interessa più la prima fase (che logicamente viene dopo) in quanto rappresenta l'essenza del problema, infatti una volta realizzato il programma di visualizzazione dei dati, i programmi di caricamento possono essere centinaia e le problematiche relative non sono specifiche della materia Computer Grafica, ma riguardano la gestione dei dati, problematica presente in tutti i settori applicativi e, inoltre, tali problematiche dipendono anche dall'eventuale uso di particolari strumenti di input di dati grafici (joystick, mouse, tavolette, ecc.).

Quindi perdonate la maniera brutale usata per l'immissione dei dati (direttamente

nel programma di visualizzazione tramite READ/DATA). Vedremo di realizzare in seguito un programma di caricamento dati, di organizzazione di questi in file, e di collegamento con un programma di visualizzazione a sé stante.

### Come organizzare il lavoro

Qualsiasi disegno architettonico può essere considerato come la rappresentazione di una serie di elementi raggruppabili in

categorie logiche (muri, finestre, porte, arredamento, ecc.) comprendenti elementi fondamentalmente simili, o addirittura uguali tra di loro, ma che possono apparire più volte in varie posizioni o con vario formato nel disegno.

In questo caso, anzi in tutti i casi dove è possibile, occorre rendere modulare il programma realizzando per ogni categoria di elementi (ad esempio le porte), una specifica routine di disegno di una singola porta, che accetti in entrata i parametri di localiz-

```
100 REM inizializzazioni e lettura data
110 KEY OFF:SCREEN 0:WIDTH 80:F=1:0=1
120 N=19:P=3.14159:KP=P/180:DIM F$(N):FOR I=1 TO N:READ F$(I):NEXT I
130 DIM A$(999):SCREEN 0:CLS:K=0:LOCATE 20,60:PRINT "attendere"
140 K=1:READ A$(K):IF A$(K)=32000 THEN GOTO 200:ELSE GOTO 140
150 REM
200 REM inizializzazione tipo output
210 ON 0 GOTO 220,230,240,250
220 XS=1:YS=1: : :SCREEN 0: :GOTO 310
230 XS=.5:YS=.5:SC=1:RS=.5:SS=.7:SCREEN 1:COLOR 1,0:GOTO 310
240 XS=1:YS=.5:SC=2:RS=1:SS=.35:SCREEN 2:GOTO 310
250 XS=1:YS=1:SC=3:RS=1:SS=.7:SCREEN 3:GOTO 310
260 REM
300 REM visualizzazione
310 REM reticolo
320 CLS:IF F THEN 400
330 FOR I=0 TO 639 STEP 50: FOR L=0 TO 399 STEP 9:LINE (I,L)-(I,L):NEXT L,I
340 FOR I=0 TO 399 STEP 50: FOR L=0 TO 639 STEP 9:LINE (L,I)-(L,I):NEXT L,I
350 REM
400 REM main routine
410 K=1
420 K=K+1:IF F=1 THEN PRINT:PRINT ,A$(K-1),F$(A$(K-1)-10000):PRINT
430 ON A$(K-1)-10000 GOSUB 1100,1200,1300,1400,1500,1600,1700,1800,1900,2000,2100,2200,2300,2400,2500,2600,2700,2800,2900
440 IS=INKEY$:IF IS="" THEN 440
450 IF A$(K)<30000 THEN 420
460 IS=INKEY$:IF IS="" THEN 460 ELSE SCREEN 0:CLS:END
470 REM
500 REM routine plottaggio di primo livello
510 IF F THEN PRINT "X",X#YS:RETURN:ELSE PSET (X#YS):RETURN
520 IF F THEN PRINT "-",X#YS:RETURN:ELSE LINE -(X#YS):RETURN
530 IF F THEN PRINT "X",Y#X,Y#X,Y#Y:RETURN:ELSE LINE (X#YS),(X#YS):RETURN
540 IF F THEN PRINT "X",Y#X,Y#X,Y#Y:RETURN:ELSE LINE (X#YS),(X#YS),(X#YS),(X#YS),B:RETURN
550 IF F THEN PRINT "X",Y#R,R#R:RETURN:ELSE CIRCLE (X#YS),R#R,,,SS:RETURN
560 IF F THEN PRINT "X",Y#R,R#R,A1,A2:RETURN:ELSE CIRCLE (X#YS),R#R,,A1,A2,SS:RETURN
570 IF F THEN PRINT S#:RETURN:ELSE LOCATE V,H#X:PRINT S#:RETURN
580 IF F THEN PRINT "X",Y#D,D#R:RETURN:ELSE PAINT (X#YS),C#:RETURN
590 REM
1000 REM routine di plottaggio
1010 REM
1100 DATA "spezzata chiusa":REM 10001
1110 X=A$(K):Y=A$(K+1):GOSUB 510:B1=A$(K):B2=A$(K+1)
1120 X=A$(K+2):Y=A$(K+1):GOSUB 520
1130 K=K+3:IF A$(K)>10000 THEN X=B1:Y=B2:GOSUB 520:RETURN
1140 X=A$(K-1):Y=A$(K):GOSUB 520
1150 K=K-1:GOTO 1120
1160 REM
1200 DATA "spezzata aperta":REM 10002
1210 X=A$(K):Y=A$(K+1):GOSUB 510:K=K+2
1220 X=A$(K):GOSUB 520:K=K+1:IF A$(K+1)>10000 THEN K=K+1:RETURN
1230 Y=A$(K):GOSUB 520:K=K+1:IF A$(K+1)>10000 THEN K=K+1:RETURN
1240 GOTO 1220
1250 REM
1300 DATA "finestra orizzontale":REM 10003
1310 X=A$(K):Y=A$(K+1):X1=A$(K+2):Y1=Y:GOSUB 530
1320 Y=A$(K+3)-4:Y1=Y:GOSUB 530
1330 Y=Y+6:Y1=Y:K=K+4:GOSUB 530:RETURN
1340 REM
1400 DATA "finestra verticale":REM 10004
1410 X=A$(K):Y=A$(K+1):X1=X:Y1=A$(K+3):GOSUB 530
```

Figura 1 - Listato del programma. Al programma, che è composto di sei gruppi di istruzioni, vanno accodati i dati numerici, letti dalla routine di lettura e caricati nel vettore di interi.

zazione, di orientamento e di scaling all'interno del disegno, e che provveda ad eseguire il disegno stesso nella posizione e nel formato voluto.

Ma poiché in nessun Basic, neanche nel più avanzato come grafica, esiste una istruzione di disegno di una porta, occorrerà inserire nella nostra routine una serie di istruzioni grafiche che, in base agli elementi numerici ricevuti, provvedano a disegnare la sospirata porta. In realtà le istruzioni Basic sono poche e nella costruzione di varie routine (del tipo Porta) ricorrerebbero più volte e quindi conviene ricorrere ad una nuova serie di subroutine a livello inferiore, che eseguano ciascuna una delle istruzioni di Grafica Basic, presente nella versione che si sta usando.

Ovvero esisteranno una serie di funzioni superiori, di una certa complessità, richiamate per ciascun elemento che compone il disegno (ad esempio Porta, Finestra, Mu-

```

4000 REM data relativi al campionario
4010 REM
4020 DATA 10001,6,6,85,50,20,20,6
4030 DATA 10002,150,6,240,70,200,40,0
4040 DATA 10003,285,5,310,12
4050 DATA 10004,384,6,391,36
4060 DATA 10005,455,6,432
4070 DATA 10006,559,50,531
4080 DATA 10007,16,122,122,142
4090 DATA 10008,180,95,205,155
4100 DATA 10009,281,115,349,135
4110 DATA 10010,420,180,40
4120 DATA 10011,500,160,40,0,120
4130 DATA 10012,5,230,0,200,0,17
4140 DATA 10013,200,260,40,120,240
4150 DATA 10014,350,200,420,250,440,210,500,240
4160 DATA 10018,20,3,80,82,75,86,65
4170 DATA 10019,190,260,230,187
4180 DATA 31999
4190 REM

```

Figura 2 - Dati relativi al campionario. Vengono eseguite, una sola volta ed in posizione ben differenziata l'una dalle altre, tutte le 16 routine presenti nel programma.

```

1420 I=AZ(K+2)-4:XI=X:GOSUB 530
1430 X=X+6:XI=X:K=K+4:GOSUB 530:RETURN
1440 REM
1500 DATA "porta in su":REM 10005
1510 DI=ABS(AZ(K+2)-AZ(K)):I=AZ(K):Y=AZ(K+1):GOSUB 510:I=AZ(K+2):GOSUB 520
1520 X=AZ(K):Y=AZ(K+1)+DI:GOSUB 520:Y=AZ(K+1):GOSUB 520:K=K+3:RETURN
1530 REM
1600 DATA "porta in giù":REM 10006
1610 DI=ABS(AZ(K+2)-AZ(K)):I=AZ(K):Y=AZ(K+1):GOSUB 510:I=AZ(K+2):GOSUB 520
1620 X=AZ(K):Y=AZ(K+1)-DI:GOSUB 520:Y=AZ(K+1):GOSUB 520:K=K+3:RETURN
1630 REM
1700 DATA "armadio":REM 10007
1710 I=AZ(K):Y=AZ(K+1):XI=AZ(K+2):YI=AZ(K+3):GOSUB 530:GOSUB 540:K=K+4:RETURN
1720 REM
1800 DATA "letto":REM 10008
1810 I=AZ(K):Y=AZ(K+1):XI=AZ(K+2):YI=AZ(K+3):GOSUB 540
1820 XI=AZ(K)+10:K=K+4:GOSUB 520:RETURN
1830 REM
1900 DATA "rettangolo generico":REM 10009
1910 I=AZ(K):Y=AZ(K+1):XI=AZ(K+2):YI=AZ(K+3):GOSUB 540:K=K+4:RETURN
1920 REM
2000 DATA "cerchio generico":REM 10010
2010 I=AZ(K):YI=AZ(K+1):R=AZ(K+2):GOSUB 550:K=K+3:RETURN
2020 REM
2100 DATA "arco generico":REM 10011
2110 I=AZ(K):Y=AZ(K+1):R=AZ(K+2):A1=AZ(K+3):#P:AZ=AZ(K+4):#P
2120 GOSUB 560:K=K+5:RETURN
2130 REM
2200 DATA "box pieno":REM 10012
2210 I=AZ(K):Y=AZ(K+1):XI=AZ(K+2):YI=AZ(K+3):GOSUB 540
2220 X=(AZ(K)+AZ(K+2))/2:Y=(AZ(K+1)+AZ(K+3))/2
2230 C=CHR$(AZ(K+4))+CHR$(AZ(K+5)):GOSUB 580:K=K+6:RETURN
2240 REM
2300 DATA "settore circ.":REM 10013
2310 I=AZ(K):Y=AZ(K+1):R=AZ(K+2):A1=AZ(K+3):#P:AZ=AZ(K+4):#P:GOSUB 560
2320 XI=X+R*COS(A1):YI=Y+R*SIN(A1):#.7:GOSUB 530
2330 XI=X+R*COS(A2):YI=Y+R*SIN(A2):#.7:GOSUB 530
2340 K=K+5:RETURN
2350 REM
2400 DATA "spezz. obliqua":REM 10014
2410 I=AZ(K):Y=AZ(K+1):XI=AZ(K+2):YI=AZ(K+3):GOSUB 530:K=K+4
2420 IF AZ(K)>10000 THEN RETURN
2430 I=AZ(K):Y=AZ(K+1):GOSUB 520:K=K+2:GOTO 2420
2440 REM
2500 DATA "a disposizione":REM 10015
2510 REM
2600 DATA "a disposizione":REM 10016
2610 REM
2700 DATA "a disposizione":REM 10017
2710 REM
2800 DATA "text":REM 10018
2810 S="":V=AZ(K):H=AZ(K+1):K=K+2
2820 S=S+CHR$(AZ(K))
2830 K=K+1:IF AZ(K)>10000 THEN GOSUB 570:RETURN
2840 GOTO 2820
2850 REM
2900 DATA "fill":REM 10019
2910 I=AZ(K):Y=AZ(K+1):C=CHR$(AZ(K+2))+CHR$(AZ(K+3)):GOSUB 580:K=K+4:RETURN
2920 K=K+1:IF AZ(K)>10000 THEN GOSUB 580:RETURN
2930 GOTO 2920
2940 REM

```

ro, Rettangolo pieno, Scritta, ecc.) e che a loro volta richiameranno delle routine primitive corrispondenti in pratica alle Istruzioni Grafiche del Basic. Realizzando il programma con questa logica risulta estremamente semplice l'organizzazione delle routine, che possono essere aggiunte e tolte senza alterare la funzionalità globale.

Nel caso poi si usasse un Basic "povero" di istruzioni grafiche si può ricorrere ad un livello intermedio di routine. Ad esempio se non esiste l'istruzione ARCO si deve realizzare una routine richiamata da ciascuna funzione di tipo Architettonico e che a sua volta richiami una o più routine che eseguano una singola istruzione Basic.

Rimane da risolvere il problema di come collegare i dati, in qualsiasi modo siano stati caricati, alle routine alle quali si riferiscono. È un problema tipico di programmazione che quindi si risolve in maniera tipica e cioè tramite una codifica. Cioè i dati costituenti un elemento della figura sono preceduti nella loro organizzazione sequenziale da un codice, che per semplificare al massimo l'archiviazione ha lo stesso formato dei dati grafici.

Il programma di visualizzazione leggerà il codice, ne riconoscerà il significato, richiamerà la relativa routine, che a sua volta richiamerà le routine di base. Quindi quando si rende necessario inserire una nuova funzione gli si attribuisce un nuovo codice e gli si costruisce la relativa routine che a sua volta si appoggerà alle istruzioni Basic, poste sotto forma di routine di base.

Se, come conviene per ragioni di memoria e come facciamo noi, i dati sono memorizzati in un unico vettore di interi, i codici potranno assumere valori numerici maggiori di 10000, mentre i dati grafici veri e propri assumeranno valori numerici inferiori. Il range 0/10000 è sufficiente per qualsiasi output su video e su plotter di dimensioni medio basse. Infatti se il plotter accetta coordinate espresse in decimi di millimetro, 10000 decimi di millimetro corrispondono a un metro...

Per tornare alle funzioni definibili possono essere di due tipi fondamentali, quelle che necessitano di un numero fisso di elementi, e quelle in cui il numero di elementi può variare. Ad esempio se si attiva la funzione TEXT, che è quella che appone delle scritte sul disegno, non è possibile sapere a priori quanto è lunga la scritta. In tal caso il programma di lettura individuerà la fine di una funzione riconoscendo l'inizio della successiva leggendo il relativo codice di individuazione.

La sequenza dei dati sarà quindi codice1, dato, dato, dato, dato, codice2, dato, ecc., numericamente 100xx, 200, 33, 432, 100, 32, 100xx, 433. I dati possono essere o codice (abbiamo detto i valori maggiori di 10000) oppure dati grafici (coordinate cartesiane, oppure angoli, raggi, caratteri ASCII, ecc.) questo dipende dalla funzione in cui sono inseriti e che quindi li riconosce.

Con questa logica di organizzazione oltre al vantaggio di poter aggiungere o to-

```

5000 REM data relativi all' applicazione
5010 REM
5020 REM 10001 spezzata chiusa
5030 DATA 10001,6,6,185,14,180,120,160,115,175,14,14,60,6
5040 DATA 10001,6,90,14,115,130,145,125,120,14,150,6
5050 DATA 10001,6,190,14,225,125,170,130,250,125,230,14,245,6
5060 DATA 10001,6,265,14,280,140,285,14,330,6
5070 DATA 10001,6,365,14,385,172,265,180,393,6
5080 DATA 10001,210,6,260,14,255,120,210,115,250,14,210
5090 DATA 10001,280,6,340,14,315,115,340,120,280,115,310,14,280
5100 DATA 10001,370,6,430,14,390,35,385,14,370,10001,470,6,530,14,470
5110 DATA 10001,570,6,633,290,570,282,625,14,570,10001,470,290,530,282,470
5120 DATA 10001,172,232,200,240,180,245,172
5130 DATA 10001,228,232,235,160,240,262,242,270,230,240,228
5140 DATA 10001,270,165,385,115,390,170,270
5150 DATA 10001,270,262,285,230,360,235,290,322,320,330,280,270,270
5160 DATA 10001,390,230,410,282,430,290,413,330,360,322,405,235,390
5170 REM 10002 spezzata aperta
5180 DATA 10002,315,34,365,85,390,35,0
5190 REM 10003 finestra orizzontale
5200 DATA 10003,185,5,210,12,10003,260,5,200,12,10003,340,5,370,12
5210 DATA 10003,430,5,470,12,10003,530,5,570,12,10003,430,282,470,289
5220 DATA 10003,530,282,570,289,10003,320,322,360,329
5230 REM 10004 finestra verticale
5240 DATA 10004,14,60,9,90,10004,14,150,9,190
5250 DATA 10004,14,245,9,265,10004,14,330,9,365,10004,172,245,178,265
5260 REM 10005 porta in su
5270 DATA 10005,125,145,102,10005,170,285,142
5280 DATA 10005,242,170,270,10005,390,235,360
5290 REM 10006 porta in giù
5300 DATA 10006,159,115,131,10006,182,115,209,10006,257,115,279
5310 DATA 10006,125,270,97,10006,228,232,200,10006,242,265,274
5320 REM 10007 armadio
5330 DATA 10007,16,122,122,142,10007,16,366,170,382
5340 DATA 10007,16,113,100,93,10007,132,170,150,210
5350 DATA 10007,363,120,384,163,10007,308,16,280,113
5360 DATA 10007,235,80,249,113,10007,392,237,403,320
5370 DATA 10007,123,232,90,247,10007,292,237,312,320
5380 REM 10008 letto
5390 DATA 10008,40,15,65,75,10008,85,15,110,75
5400 DATA 10008,40,286,65,346,10008,66,286,91,346,10008,28,225,90,285
5410 REM 10009 rettangolo generico
5420 DATA 10009,181,15,249,35,10009,183,17,240,33
5430 DATA 10009,181,45,200,65,10009,186,47,198,63
5440 DATA 10009,249,45,229,55,10009,249,63,227,75
5450 DATA 10009,245,65,229,73,10009,15,231,35,279
5460 DATA 10009,17,233,33,270,10009,45,231,57,251
5470 DATA 10009,47,235,55,249,10009,68,231,80,251
5480 DATA 10009,50,279,70,265,10009,52,275,68,267
5490 DATA 10009,316,35,336,66,10009,316,67,336,88
5500 DATA 10009,316,89,336,114,10009,320,38,332,49,10009,320,52,332,63
5510 REM 10019 fill
5520 DATA 10019,10,10,238,187,10019,10,100,238,187
5530 DATA 10019,10,220,238,187,10019,10,300,238,187
5540 DATA 10019,10,380,238,187,10019,220,10,238,187
5550 DATA 10019,400,10,238,187,10019,500,10,238,187
5560 DATA 10019,600,10,238,187,10019,500,284,238,187
5570 DATA 10019,175,235,238,187,10019,285,284,238,187
5580 DATA 10019,420,288,238,187,10019,380,158,238,187
5590 DATA 10019,300,10,238,187,10019,235,250,238,187
5600 DATA 10019,206,50,0,17,10019,50,255,0,17
5610 REM 10018 text
5620 DATA 10018,11,60,83,65,76,79,78,69,10018,6,43,67,85,67,73,78,65
5630 DATA 10018,6,25,66,65,71,78,79,10018,17,6,66,65,71,78,79
5640 DATA 10018,13,22,68,73,83,10018,10,34,68,73,83,73,77,80,69,71,78,79
5650 DATA 10018,6,6,76,69,84,84,79,10018,12,6,76,69,84,84,79
5660 DATA 10018,23,6,76,69,84,84,79,10018,18,42,83,84,85,68,73,79
5670 DATA 10018,13,34,73,78,71,82,69,83,83,79,32000

```

Figura 3 - Dati relativi all'applicazione. I dati con i quali viene disegnata la piantina dell'appartamento sono circa 650, per una occupazione di memoria di soli 1.3 KB.

gliere in qualsiasi momento funzioni che disegnano un singolo elemento, si ha il vantaggio di una estrema libertà nell'immissione dei dati, che possono essere messi in sequenza con qualsiasi ordine, l'unico vincolo essendo quello di essere preceduti dal proprio codice di funzione. Ovvero ad un disegno potrà in ogni momento essere aggiunto o tolto un elemento senza dover modificare nulla del programma e senza dover alterare l'ordine di immissione dei dati. È comunque sempre consigliabile, anche se come in questo caso non obbligatorio, seguire una certa logica nella immissione dei dati soprattutto per facilitare il DEBUG del disegno, nella fase di "creazione".

L'altra logica fondamentale è quella di svincolarsi da problemi di scaling, ovvero l'esecuzione del programma deve essere indipendente dal formato in cui vengono immessi i dati. E cioè il programma deve contenere funzioni di scaling, estese alle funzioni particolari come TEXT, FILL, ARC, ecc., che permettano di realizzare uscite di vario formato, partendo dagli stessi dati fondamentali. Nel programma che presentiamo è possibile scegliere tre tipi di uscite grafiche differenti corrispondenti ai tre formati SCREEN, permessi dal GWBASIC dell'Olivetti M24.

Passiamo ora ad esaminare il programma presentato nel quale si ritrovano esemplificati tutti i concetti espressi in maniera discorsiva. Per quanto riguarda i dati da

visualizzare ne forniamo due liste, una che rappresenta un campionario delle funzioni presenti nel programma, l'altra che permette il disegno di una piantina di appartamento arredato di una certa complessità (salone, quattro letto, pluriservizi).

### Analisi della struttura del programma

Nella figura 1 è riportato il listato del programma (vale sia per l'IBM PC che per l'Olivetti M24) pronto a ricevere dati. Nelle figure 2 e 3 sono invece fornite due liste di dati "mergiabili" al programma, la prima delle quali produce un campionario delle funzioni attivate (sono 16) e riprodotte nelle figure. 4 e 5 e la seconda produce la piantina di cui alle figure successive.

La struttura del programma è molto semplice e consta di 6 parti successive:

#### 1 - Inizializzazioni generali e lettura dei dati (da riga 100).

Comprende il settaggio delle condizioni iniziali di lavoro, il dimensionamento dei vettori, la definizione delle costanti e la lettura di tutti i dati grafici. La condizione di fine lettura è riconosciuta da un "tappo" nei dati, costituito dal valore 32000. Con questo sistema, immettendo subito il tappo, si possono verificare i dati grafici via via che si immettono semplificando la scoperta di quelli sbagliati.

#### 2 - Inizializzazione del tipo Output (da riga 200).

Per non legare rigidamente il formato

dei dati al particolare output scelto occorre definire dei parametri di scaling che vanno settati inizialmente e che non debbono entrare in fase di lettura, ma in fase di visualizzazione come fattori presenti nelle routine di plottaggio di primo livello. Nel nostro caso, come detto, abbiamo tre formati più una uscita alfanumerica per il debug numerico dei dati.

SCREEN 0 - uscita alfanumerica

SCREEN 1 - uscita grafica - 320 per 200 pixel - 4 colori

SCREEN 2 - uscita grafica - 640 per 200 pixel - 2 colori

SCREEN 3 - uscita grafica - 640 per 400 pixel - 2 colori

Quest'ultima modalità non è possibile con l'IBM in configurazione standard.

Poiché le tre modalità grafiche non sono legate da un rapporto di proporzionalità tra di loro, non è possibile utilizzare un solo fattore di scaling, ma ne occorrono due per le due direzioni principali, due per i parametri delle istruzioni CIRCLE e che servono per correggere lo schiacciamento degli archi e delle circonferenze. Occorre inoltre differenziare le uscite TEXT a seconda che si esca in SCREEN 1 (40 colonne) o in SCREEN 2 e SCREEN 3 (80 colonne).

#### 3 - Visualizzazione (da riga 300)

Nel listato è presente una routine di "reticolo", utilissima in fase di predisposizione dei dati, ma da saltare o da eliminare quando i dati sono stati caricati e sono

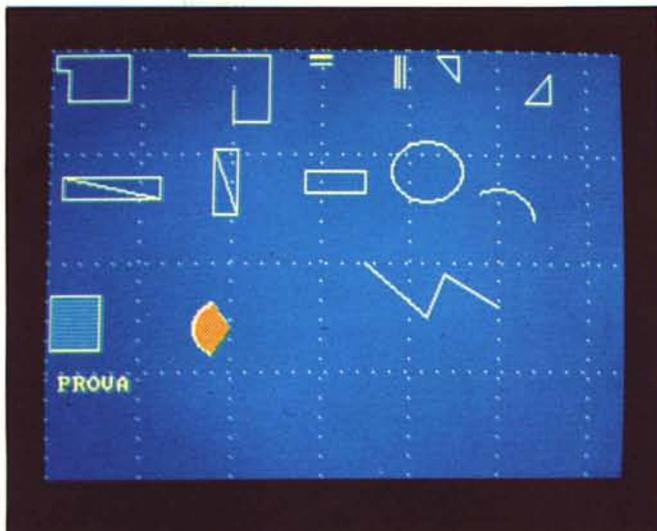


Figura 4 - Campionario (eseguito in SCREEN 1). È stata scelta l'opzione di output 2, che comporta l'uscita con definizione 320 x 200 pixel a 4 colori, le scritte, occupando una matrice di 40 righe per 24 colonne, appaiono molto grosse.

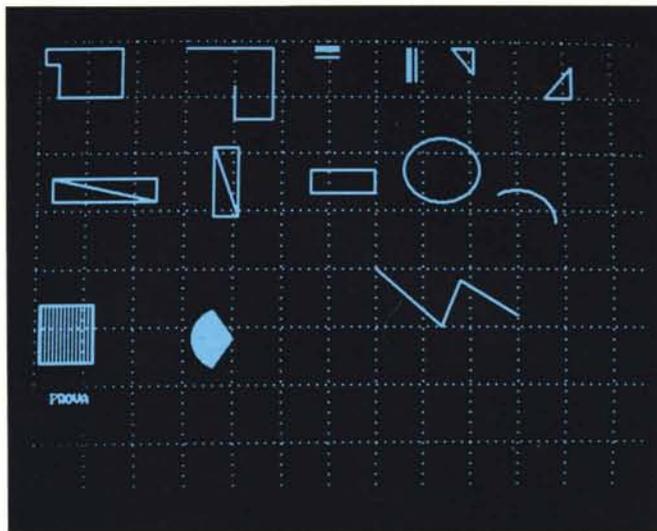


Figura 5 - Campionario (eseguito in SCREEN 3). È l'output che utilizza la definizione più spinta (640 x 400 pixel in bianco e nero). I vari fattori di scala permettono l'adeguamento di tutte le routine ai vari formati di uscita.

corretti. Il vantaggio di avere un reticolo sarà visto dopo quando forniremo consigli pratici per l'uso del programma.

La MAIN routine si fonda sul lavoro del contatore K che permette di scorrere tutto il vettore dei dati, che come detto è un vettore di interi. All'interno della MAIN vengono riconosciuti i codici funzione (quelli > 10000) e quindi avviene il rimando alla specifica routine di plottaggio. È all'interno della routine di plottaggio che il contatore viene incrementato (di una quantità fissa o variabile a seconda della funzione) in modo tale che il rientro al MAIN avvenga con il contatore posizionato sul codice successivo. Se il codice successivo è superiore o uguale a 32000 l'esecuzione termina.

Nel caso si scelga la modalità SCREEN 0, quella alfanumerica utile per il debug, viene settato a 1 il flag F e quindi le varie

routine grafiche vengono sostituite da routine di PRINT. Se si dispone di dati esatti e si vuole alleggerire il programma vanno eliminate tutte le istruzioni o parte delle istruzioni dove appare IF F THEN...

In riga 440 c'è la tipica istruzione "premi un tasto per continuare", anche questa utile in fase di debug in quanto isola l'esecuzione di una routine di plottaggio dalla successiva permettendo l'individuazione di eventuali errori. Routine da eliminare nella versione finale del disegno.

#### 4 - Routine di Plot di Primo Livello (da riga 500)

Sono otto e corrispondono alle istruzioni BASICA:

- PSET - plottaggio di un punto;
- LINE - plottaggio di una linea fino a...
- LINE - plottaggio di una linea da ... fino a...
- LINE B - plottaggio di un rettangolo

- CIRCLE - plottaggio di una circonferenza
- CIRCLE A - plottaggio di un arco
- PRINT - apposizione di una scritta
- PAINT - riempimento di un'area

Sono tutte affette dalla condizione che individua l'uscita alfanumerica (IF F THEN) che, se attivata, produce il PRINT dei valori anziché il loro PLOT. Inoltre comprendono ciascuna i citati fattori di scala che traducono i valori numerici originari (quelli letti con i DATA) in valori numerici compatibili con il formato di uscita prescelto.

#### 5 - Routine di Plottaggio di Primo Livello (da riga 1100)

La numerazione è studiata in modo da mettere in corrispondenza diretta il codice della funzione con la routine interna al programma. Cioè la routine "finestra orizzontale", il cui codice di riconoscimento è 10003, è posizionata a partire dalla riga

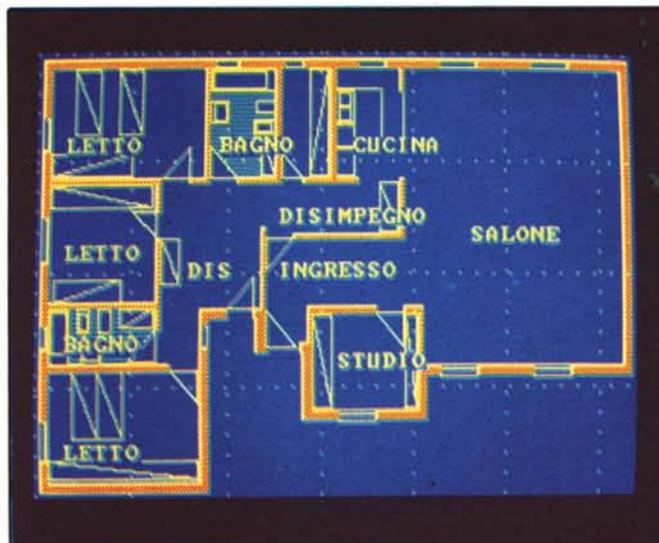


Figura 6 - Piantina (eseguita in SCREEN 1). L'opzione COLORE non offre in questo tipo di applicazioni, particolari vantaggi rispetto al BIANCONERO che offre però una maggiore definizione.

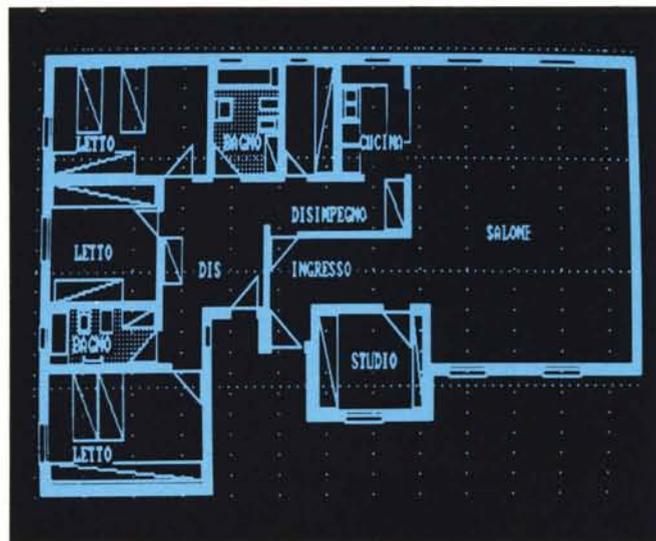


Figura 7 - Piantina (eseguita in SCREEN 2). È la definizione massima permessa dall'IBM dotato di scheda grafica. Rispetto alla definizione superiore, SCREEN 3 offerta dal GWBASIC, viene penalizzata soprattutto la funzione di PRINT.

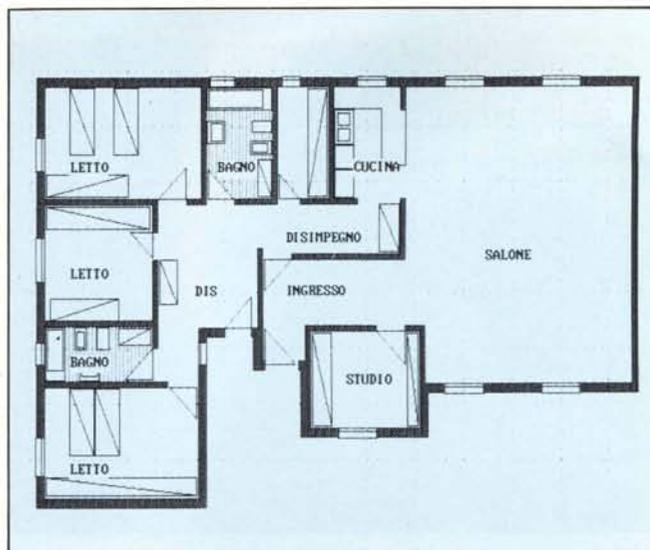
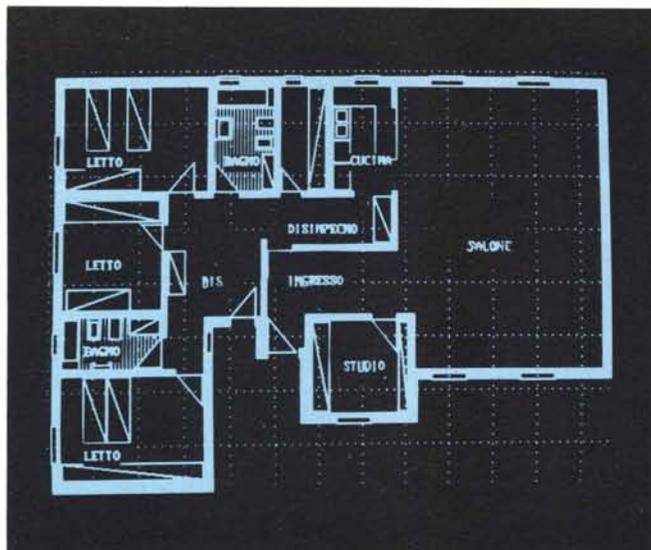


Figure 8/9 - Piantina (eseguita in SCREEN 3). Il risultato è ottimo sia sul monitor del M24 sia sulla stampante Olivetti PR15 che permette l'Hard Copy della pagina SCREEN 3.

1300. Questo sempre per rendere più maneggevole l'aggiunta di routine o la modifica di quelle già esistenti, senza dover intervenire in altre parti del programma.

Le routine presentate sono 16, ma possono essere molte di più oppure qualcuna di meno. Ad esempio se si combinano tra di loro la CIRCLE + la PAINT si ha una nuova routine non indispensabile in quanto sostituibile dalle altre due. Sarà utile realizzarla solo se è di uso frequente. Oppure supponiamo che nel disegno fossero presenti degli esagoni. Se la routine specifica non esiste si può utilizzare la spezzata obliqua, ma è chiaro che una routine specifica permette di ottimizzare i consumi di memoria e quindi i tempi di esecuzione in quanto per definire un esagono regolare occorrono meno dati che per definire una spezzata obliqua composta da 7 vertici.

Per inserire nuove funzioni si possono utilizzare i codici 10015, 10016, 10017 lasciati a disposizione. Oppure si possono aggiungere in coda, da riga 3000 in poi. In tal caso le righe in più vanno aggiunte pure nell'istruzione di riga 430 che si occupa del rimando alle routine di plottaggio. Senza entrare nel dettaglio di tutte le routine citiamo solo quella il cui codice è 10001, e cioè la "spezzata chiusa" in cui i vari segmenti sono paralleli ai due assi principali. In questa maniera una volta fornite le coordinate del punto iniziale basta fornire, per individuare i segmenti successivi, solo la coordinata che cambia (e cioè la Y se il segmento è verticale e la X se il segmento è orizzontale). La routine 10018 (TEXT) è stata adattata alla logica del vettore di interi. Ovvero la stringa da visualizzare è stata tradotta nei corrispondenti caratteri ASCII. Questo in definitiva oltre a non essere una soluzione elegante comporta l'inconveniente di "consumare" due byte per carattere invece del solo che consumeremmo se memorizzassimo direttamente una stringa. In realtà ne guadagnamo in termini di semplicità di gestione delle routine.

## 6 - I dati (in coda)

I dati non richiedono nessuna organizzazione particolare, gli unici due vincoli sono quelli di mettere il "tappo", ed è bene metterlo subito, e quello che i dati siano raggruppati in codice + dati relativi al codice.

Vedremo subito nel paragrafo Consigli Pratici di Utilizzazione quale sia l'organizzazione più produttiva e gli errori più comuni nella fase di preparazione.

### Elenco delle variabili principali

F	flag per il riconoscimento del tipo di Output
N	numero delle funzioni attivate
FS(N)	lista delle funzioni attivate
P,KP	Pigreco e Pigreco/180
A%	vettore dei dati grafici (in formato intero)
K	contatore del vettore
O	tipo di out scelto (valori possibili 1, 2, 3, 4)

In funzione dell'Out prescelto vengono settati i vari fattori di scaling:

XS,YS	nelle direzioni X e Y
RS	raggio
SS	schacciamento dell'arco

Le variabili passate alle routine di plottaggio di primo livello:

X,Y	coordinate X,Y
R	raggio di arco o circonferenza
V,H	localizzazione delle scritte

### Consigli pratici (ed empirici) di utilizzazione

Per chi sia riuscito ad arrivare fin qui e magari abbia intenzione di provare il programma sulla propria macchina diamo una serie di consigli pratici che lo possono aiutare sia a capire meglio la filosofia di tale programma che ad utilizzarlo praticamente.

Facciamo l'esempio di una applicazione relativa al progetto di una casa. La cosa più opportuna è quella di predisporre una

minuta della pianta con una matita su di un foglio di carta a quadretti in modo tale che le coordinate dei vari componenti siano leggibili direttamente sulla quadrettatura senza dover ricorrere a misurazioni con riga e squadra. Eseguita a mano la minuta, si passa ad immettere i dati come DATA in coda al programma.

Immettendo il "tappo" è possibile eseguire ogni tanto delle prove senza dover aspettare la fine del caricamento.

Un aiuto al debug dei dati grafici è costituito dalla quadrettatura, che sarebbe opportuno rendere proporzionale a quella del foglio su cui si è eseguita la minuta. Altro aiuto è costituito dall'uscita alfanumerica, che, in più rispetto a quella grafica, scrive in chiaro il nome della routine attivata e i valori dei dati numerici ad essa relativi. Tra l'esecuzione di una routine e di quella successiva è bene lasciare in fase di debug un'attesa per meglio individuare lo scorrere dei dati.

Per dare una valutazione dei volumi dei dati necessari alla realizzazione di una piantina di una certa complessità diremo che i DATA del campionario sono 90 mentre quelli dell'applicazione sono 650. Il che tradotto in byte vuol dire solo 1300, trattandosi di un vettore di interi. Questo significa che un qualsiasi Basic con un po' di memoria a disposizione può comunque sopportare disegni molto più complessi.

Abbiamo detto che non è necessario seguire alcun ordine prestabilito nell'immissione dei dati. È vero, ma comunque ci permettiamo di suggerire di raggruppare i dati nello stesso ordine delle routine che richiamano. È invece consigliabile, e questo vale per ogni programma grafico, di apporre le scritte alla fine, eseguendo le routine relative per ultime. Questo in quanto il Print provoca comunque la cancellazione di un formato carattere e questo può provocare delle conseguenze in caso di fill di aree che potrebbero non risultare chiuse.

# NELLE TUE MANI

## tutta la potenza di una grande stampante

### P-40 ideale per home e personal computer

Questa è Epson P-40, la stampante termica ultracompatta, quasi tascabile, la compagna ideale per il tuo personal computer a casa, a scuola e anche nel lavoro.

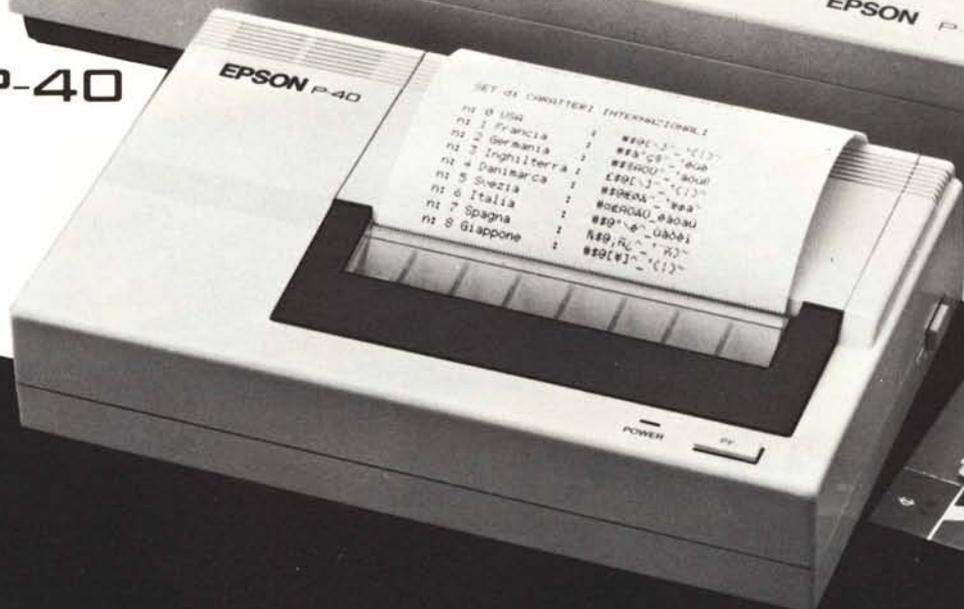
Piccola, robusta, progettata per lavorare a lungo e realizzata con la proverbiale qualità Epson, la P-40 funziona con batterie ricaricabili e stampa grafici e testi su 20, 40 o 80 colonne (modo compresso) a 45 caratteri al secondo.

Regala Epson P-40 al tuo personal. Con la piccola Epson il tuo personal diventa grande!

### P-80



### P-40



EPSON l'informatica portatile, anche nelle periferiche



# PERSONAL ATARI 130XE

MIT

## NUMERI, NON PAROLE.

ATARI 130 XE è un grande personal computer che, ad una enorme capacità di memoria, abbina una maggiore versatilità. Atari 130XE non ha bisogno di molte parole per essere presentato, i suoi numeri parlano da soli.

### 128K RAM

ATARI 130XE utilizza un microprocessore 6502C. Ha una memoria di 128K RAM, 24K ROM (Sistema operativo e linguaggio programmazione Basic). Tanta potenza in più per un maggior numero di informazioni.

### 4 CHIPS SPECIALI

Sono Chips esclusivi Atari: GTIA - Per la visualizzazione grafica; POKEY - Generatore e controllo di suono; AN-TIC - Per la gestione dello schermo e di Input /Output; FREDDY - Sistema di controllo della memoria.

### 256 COLORI

ATARI 130XE ha capacità grafiche nettamente superiori alla media: 11 tipi di modi grafici; 256 colori (128 visualizzati contemporaneamente); grafica 320 x 192 ad alta risoluzione; una visualizzazione di 40 colonne per 24 linee; 5 modi di testo. Tutto quanto ti serve per scatenare al meglio

la tua creatività, senza limiti di alcun tipo.

### 4 CANALI SONORI

Anche per le capacità sonore, Atari 130XE rivela straordinarie e inaspettate possibilità: 4 canali sonori indipendenti su una gamma di 3 ottave e mezza.

### 62 TASTI

Una tastiera formidabile: 62 tasti, tasto Help e quattro tasti per funzioni speciali; caratteri internazionali; 29 tasti con funzione grafica. Linguaggio di programmazione Atari Basic incorporato.

### 41 CONFIGURAZIONI POSSIBILI

Software e periferiche compatibili con i computers della serie XL: Stampante a colori Atari 1020; stampante "Letter Quality" Atari 1027; Stampante ad aghi programmabile Atari 1029; Disk drive Atari 1050 (È possibile l'utilizzo di 4 unità in serie); Joystick e Super Controller.

**L. 380.000+ IVA**



**ATARI® COMPUTERS  
TECNOLOGIA FORTE, PREZZO VINCENTE.**

**I NOSTRI RIVENDITORI:** Lazio Videosuono V.A. Baldovinetti 68/74 - Roma 06/5038525-5035880 - Lombardia + Novara Consolo & Longoni V.le dell'Industria 63 - Paderno Dugnano 02/9183372-9184083 - Piemonte - Valle d'Aosta Norvat C.so Fiume 12 - Torino 011/682171 - Tre Venezie Interservice V.S. Pietro 58/A - Padova 049/655654 - Emilia+S. Marino Telcom Via dei Mille 19 - Bologna 051/274600 - Toscana, Umbria, Marche+La Spezia Telebit c/o Sig. Cattani Via Il Prato 8/R - Firenze 055/262652 - Puglia, Abruzzo, Basilicata, Molise Domina V. Turati 16 - Bari 080/420204 - Sicilia, Calabria Belco V. Mariano d'Amelio 78 - Palermo 091/547566 - Sardegna Sicom Strada Sestu - Elmas Km. 2,1 Sestu - Cagliari 070/22317 - Campania Lada V. Ferrante Imperato 33 S.Giovanni a Teduccio (NA) 081/7527006  
DISTRIBUZIONE NAZIONALE Canale Foto Cine Circe Film Via Pascoli 70/3 - Milano 02/2363556 - 2366410