

di Francesco Petroni

La grafica di presentazione

Seconda parte

Nel numero scorso abbiamo cominciato a trattare l'argomento «Grafica di Presentazione» intendendo con tale dizione le due attività di predisposizione di immagini e di organizzazione delle stesse in una sequenza logica.

Abbiamo visto come la periferica o lo strumento previsto per la presentazione condizioni tutta l'attività. Se occorrono diapositive o output su stampa le varie immagini possono essere realizzate con strumenti software differenti e in qualsiasi ordine logico.

Se la presentazione va eseguita su computer è necessario organizzare via software l'esposizione delle immagini decidendo innanzitutto l'ordine logico e poi le modalità di avanzamento, comandato via tastiera o via routine temporizzatrice.

In questo articolo vedremo alcune problematiche inerenti la composizione del disegno e vedremo anche come utilizzare nella composizione del disegno stesso farina non del nostro sacco, ovvero pezzi di immagini prelevate qua e là nel software che ci capita di maneggiare.

L'ambito hardware in cui opereremo è quello IBM o IBM compatibile.

Utilizzo del joystick

Il fenomeno della diffusione dello standard IBM e IBM compatibili sta avendo una serie di conseguenze indirette. Un esempio è la contemporanea diffusione di periferiche, schede, accessori più o meno originali che ha come mercato l'insieme degli utenti IBM e degli utenti compatibili.

Tra gli accessori che hanno avuto successo citiamo la famosa scheda multifunction, non presente nei cataloghi IBM, ma che riassume le capacità di almeno quattro schede monofunzione. Alcune di queste sono fondamentali, come le interfacce parallele e seriali, e le espansioni di memoria.

In tale «nobile» compagnia risulta frivola la presenza della porta giochi, che in casa IBM necessita di una scheda specifica e che per questo non è mai stata un accessorio diffuso, quasi fosse poco professionale.

Come utilizzare nei propri programmi il joystick

Il Basic IBM possiede due funzioni

che identificano parametri legati al joystick. Esse sono:

X1 = STICK(0) per il primo joystick
 Y1 = STICK(1)
 X2 = STICK(2) per il secondo joystick
 Y2 = STICK(3)

È chiaro che i range entro i quali variano i valori ritornati dalle funzioni dipendono dai particolari joystick usati, e quindi in ogni programma applicativo occorrerà una procedura iniziale di taratura.

La seconda funzione è:

F = STRIG(n)

che fornisce una serie di valori in funzione dello stato dei quattro tasti, due per joystick, riconosciuti.

Tale tasto comporta una modalità di attivazione e di disattivazione che sono:

STRIG (n) ON
 STRIG (n) OFF
 STRIG (n) STOP

Le combinazioni con due joystick e quattro tasti sono numerosissime e sono ovviamente destinate alla realizzazione di giochi per due antagonisti, non essendo possibile da parte di un unico utilizzatore tenere sotto controllo tanti comandi.

Anche in un semplice programma

```

100 REM inizializzazione schermo
110 SCREEN 2:CLS:F=-1:S=8:T=S/2
120 PRINT "Punta in Alto A Sinistra":GOSUB 900
130 GOSUB 900:X1=X:Y1=Y:FOR K=1 TO 800:NEXT K
140 PRINT "Punta in Basso A Destra":GOSUB 900
150 GOSUB 900:X2=X:Y2=Y:FOR K=1 TO 800:NEXT K
160 X3=(X1+X2)/2:Y3=(Y1+Y2)/2:R=(Y2-Y1)*.3
170 CLS:WINDOW (X1,Y1)-(X2,Y2):LINE (X1,Y1)-(X2,Y2),B
180 REM dati numerici
190 LOCATE 20,2 : PRINT " Da ";X1; " , ";Y1
200 LOCATE 21,2 : PRINT " A ";X2; " , ";Y2
210 LOCATE 22,2 : PRINT " Centro ";X3; " , ";Y3
220 END
900 REM vertici
910 STRIG ON
920 X=STICK(0):Y=STICK(1)
930 IF STRIG(0)=0 THEN 920 ELSE STRIG OFF:RETURN
  
```

Figura 1 - Programma di Inizializzazione dello Schermo di Lavoro per il joystick. Utilizza le due funzioni del Basic IBM che leggono lo stato del joystick e dei suoi pulsanti per definire il loro range di lavoro e per stabilire di conseguenza la finestra di lavoro.

```

100 REM fidae
110 SCREEN 2:CLS:F=-1
120 PRINT "Punta in Alto A Sinistra ":GOSUB 900:X1=X:Y1=Y
130 PRINT "Punta in Basso A Destra ":GOSUB 900:X2=X:Y2=Y
140 CLS:K=40:WINDOW SCREEN (X1,Y1)-(X2,Y2)
150 LINE (X1,Y1)-(X2,Y2),B
200 REM posizionamento
210 GOSUB 950:IF F=1 THEN 210 ELSE PSET (X,Y)
220 LINE -(X,Y):GOSUB 950:IF F=1 THEN 210 ELSE 220
900 REM lettura con attesa
910 STRIG ON
920 X=STICK(0):Y=STICK(1)
930 IF STRIG(0)=0 THEN 920 ELSE STRIG OFF
940 FOR K=1 TO 200:NEXT K:RETURN
950 REM lettura senza attesa
960 STRIG ON
970 X=STICK(0):Y=STICK(1)
980 IF STRIG(0)<>0 THEN F=F-1:STRIG OFF
990 LOCATE 23,3:PRINT F:FOR K=1 TO 40:NEXT K:RETURN
  
```

Figura 2 - Programma di Utilizzazione del joystick - Uno. Una volta definito lo schermo di lavoro occorre individuare le istruzioni che permettono di gestire il movimento di un cursore sul video. Cursore che è lo strumento fondamentale nei programmi grafici.

```

100 REM fitre
110 SCREEN 2:CLS:F=-1:S=0:T=S/2
200 REM vertici
210 PRINT "Punta in Alto A Sinistra ":GOSUB 500:X1=X:Y1=Y
220 FOR K=1 TO 800:NEXT K
230 PRINT "Punta in Basso A Destra ":GOSUB 500:X2=X:Y2=Y
240 FOR K=1 TO 800:NEXT K
250 SCREEN 2:CLS
260 WINDOW SCREEN (X1,Y1)-(X2,Y2):LINE (X1,Y1)-(X2,Y2),,B
270 X3=(X1+X2)/2:Y3=(Y1+Y2)/2:R=(Y2-Y1)*.3
300 REM posizionamento
310 GOSUB 550:IF F=-1 THEN 350
320 X4=X+RND*S-T:Y4=Y+RND*S-T
330 D=SQR((X4-X3)^2+(Y4-Y3)^2)
340 IF D<R THEN PSET(X4,Y4)
350 GOTO 310
500 REM lettura con attesa
510 STRIG ON
520 X=STICK(0):Y=STICK(1)
530 IF STRIG(0)=0 THEN 520 ELSE STRIG OFF
540 FOR K=1 TO 200:NEXT K:RETURN
550 REM lettura senza attesa
560 STRIG ON
570 X=STICK(0):Y=STICK(1)
580 IF STRIG(0)<>0 THEN F=F*-1:STRIG OFF
590 LOCATE 23,3:PRINT F:FOR K=1 TO 40:NEXT K:RETURN

```

Figura 3 - Programma di Utilizzazione del joystick - Due. Una volta creato il rapporto tra il movimento del joystick e il cursore sul video lo si applica alle varie funzionalità, in questo caso il Brush.

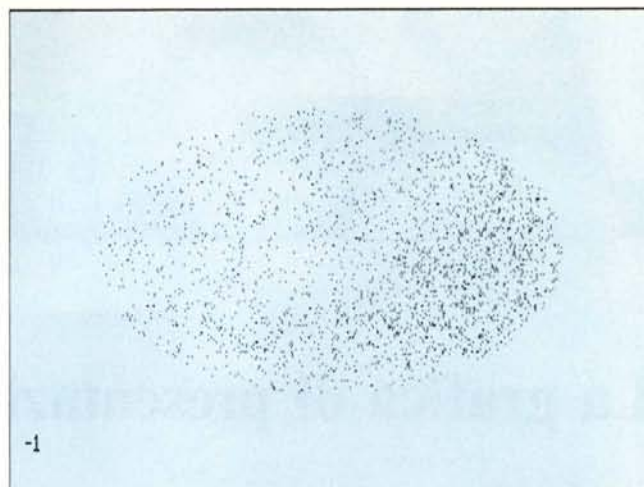


Figura 4 - Out del Programma joystick - Due. Il Brush riproduce l'effetto che in grafica tradizionale si ottiene con l'aerografo. Con il computer si calcolano tanti punti random attorno al cursore mosso dal joystick.

grafico i problemi da risolvere, e da risolvere contemporaneamente, sono vari.

Il primo è quello di collegare il movimento del joystick al movimento di un cursore sul video. Il secondo è quello di permettere di «vedere» la posizione del cursore indipendentemente dall'uso che si fa del cursore

stesso. Il terzo è quello di attivare varie funzionalità scelte tramite un'ulteriore funzionalità di scelta. Il quarto è quella di gestire la funzionalità pre-scelta.

Per esemplificare l'uso del joystick presentiamo tre programmi DEMO. Il primo FIUNO (listato in fig. 1) serve per inizializzare il joystick e quindi

rendere il suo range di lavoro identico al formato video.

Va notato che la routine di riconoscimento tasto premuto è critica in quanto il tasto va sempre interpretato come un interruttore, che va premuto per ottenere un cambio di stato. Se viene premuto troppo succede che lo stato ritorna alla situazione iniziale, e quindi la routine di riconoscimento non deve essere continua ma deve attivare o disattivare un flag in funzione dello stato precedente dell'interruttore.

Al contrario la routine di riconoscimento posizione del cursore deve fornire le coordinate più velocemente possibile, in modo da renderle utilizzabili subito. Questo serve, in particolare, nelle funzionalità a mano libera, dove al movimento della mano corrisponde una immediata conseguenza sul video.

Con il secondo programma, listato figura 2, si fa un passo in avanti. Dopo la necessaria configurazione iniziale, il joystick viene utilizzato per disegnare una linea continua sul video. Il disegno si può interrompere o riprendere accendendo o spegnendo il tasto e la situazione del tasto viene visualizzata in basso a sinistra.

Non è stata inserita, per non complicare il listato, la routine di visualizzazione posizione del cursore.

In figura 3 presentiamo un'altra utilizzazione del joystick per creare l'effetto «spruzzo», ma ne parliamo più tardi.

Basic Compilatore

L'ultima release del compilatore Basic per IBM, ci riferiamo al Basic Compiler Version 2.0, è compatibile con tutte le istruzioni grafiche Basic e quindi risulta particolarmente interes-

```

100 REM test_0
110 DEFINT I,L
120 TIME#="00:00:00":PRINT TIME#
130 FOR L=1 TO 10
140 FOR I=1 TO 30000
150 NEXT I:PRINT TIME#
170 NEXT L
180 INPUT A
190 END

100 REM test_1
110 KEY OFF:SCREEN 2:CLS:TIME#="00:00:00"
120 WINDOW (-20,-14)-(20,14):LINE (-20,-14)-(20,14),,B
130 RH=30:D=20:TH=.6:PH=.9:S1=SIN(TH):C1=COS(TH):S2=SIN(PH):C2=COS(PH)
140 A=12:B=8:C=12:M=(C^2)/(A^2):N=(C^2)/(B^2)
150 DEF FN XX(X)=SQR(B^2*(1+(X^2)/(A^2)))
160 DEF FN YY(Y)=SQR(A^2*(1+(Y^2)/(B^2)))
170 DEF FN ZZ(X,Y)=SQR(C^2+M*X^2+N*Y^2)
200 REM assi
210 I=20:L=0:Z=0:GOSUB 400:PSET(SX,SY):I=-30:GOSUB 400:LINE-(SX,SY)
220 I=0:L=20:Z=0:GOSUB 400:PSET(SX,SY):L=-30:GOSUB 400:LINE-(SX,SY)
230 I=0:L=0:Z=20:GOSUB 400:PSET(SX,SY):Z=-30:GOSUB 400:LINE-(SX,SY)
300 REM main inizio
310 FOR K=-1 TO 1 STEP 2
320 FOR I=-A TO A:X0=FN XX(I):FOR L=-X0 TO X0:Z=K*(FN ZZ(I,L))
330 GOSUB 400:IF L=-X0 THEN PSET(SX,SY):ELSE LINE-(SX,SY)
340 NEXT L:NEXT I
350 FOR L=-B TO B:Y0=FN YY(L):FOR I=-Y0 TO Y0:Z=K*(FN ZZ(I,L))
360 GOSUB 400:IF I=-Y0 THEN PSET(SX,SY):ELSE LINE-(SX,SY)
370 NEXT I:NEXT L
380 NEXT K:PRINT TIME#:INPUT A#:END
400 REM da tridi a bidi
410 XE=-I*S1+L*C1:YE=-I*C1*C2-L*S1*C2+Z*S2:ZE=-I*S2*C1-L*S2*S1-Z*C2+RH
420 SX=D*XE/ZE:SY=D*YE/ZE:RETURN

100 REM test_2
110 SCREEN 2:OPTION BASE 1
120 OPEN "sfinge.dat" FOR INPUT AS 1
130 INPUT #1,D:DIM A%(D)
140 FOR I=1 TO D:INPUT #1,A%(I):NEXT I:CLOSE #1
150 TIME#="00:00:00"
160 CLS:FOR X=0 TO 350:PUT (X,0),A%,PSET:NEXT X
170 LOCATE 20,1:PRINT TIME#:INPUT A#
180 END

```

Figura 5 - Insieme dei Programmi Demo. Tali programmi riproducono campi estremi di utilizzazione del compilatore Basic, e infatti le prestazioni in termini di tempo risultano assolutamente variabili.

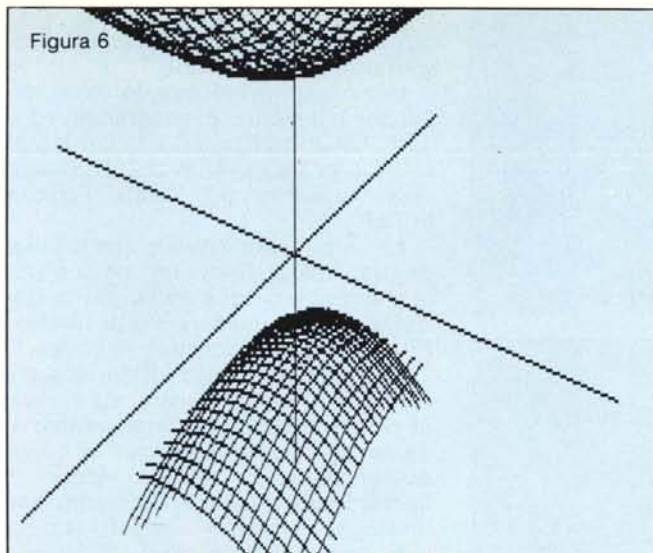


Figura 6

Figura 6 - HardCopy del Secondo Programma Demo. Il sistema più semplice per riversare su carta un grafico prodotto su video è quello di eseguirne una hardcopy, che come noto si ottiene caricando il programma GRAPHICS presente nel dischetto DOS e poi semplicemente premendo il tasto Shift PrtSc.

sante testarne le prestazioni in questo particolare campo di utilizzo.

Va detto che questo prodotto è compatibile anche con le ultime novità hardware di casa IBM e cioè la PC Network e le schede Grafiche avanzate.

In generale si può affermare che il miglioramento delle prestazioni, in termini di tempo, permesse da un compilatore interprete sono tali da rendere praticabili alcune soluzioni non adottabili con un linguaggio lento e questa circostanza si presenta spesso nelle applicazioni grafiche, anche se le istruzioni grafiche, dipendendo in larga parte dall'hardware, sono quelle che meno si dovrebbero avvantaggiare della compilazione.

Per verificare praticamente questi concetti abbiamo scritto tre programmi di test (listati in fig. 5), il primo di calcolo puro, il secondo di calcolo e visualizzazione grafica, e il terzo di grafica pura. I risultati rispecchiano in maniera perfetta questa suddivisione logica.

Il primo programma che non fa che

Figure 8 e 9 - Programma di Scorrimento Figurine. Sono due pagine dell'album di figurine, queste sono state prelevate dal prodotto PrintMaster che serve per realizzare biglietti di auguri, decorazioni natalizie, ecc.

Figura 7

```

100 REM
110 SCREEN 2:OPTION BASE 1:DIM AX(1000):H=64:L=104:P=0:CLS
120 READ A$:IF A$="FINE" THEN END
130 A$=A$+".dat":ERASE A$
140 OPEN A$ FOR INPUT AS I
150 INPUT #1,D:DIM AX(D)
160 FOR I=1 TO D:INPUT #1,AX(I):NEXT I:CLOSE #1
170 Y=INT(P/6):X=P-Y*6:X1=X*L:Y1=Y*H:X2=X*13+2:Y2=Y*8+1
180 PUT (X1,Y1),AX,PRESET:P=P+1:IF P>17 THEN P=0
190 LOCATE Y2,X2:Q=Q+1:PRINT Q:GOTO 120
200 REM data
210 DATA "ima01","ima02","ima03","ima04","ima05","ima06"
220 DATA "ima07","ima08","ima09","ima10","ima11","ima12"
230 DATA "ima13","ima14","ima15","ima16","ima17","ima18"
240 DATA "ima19","ima20","ima21","ima22","ima23","ima24"
250 DATA "ima25","ima26","ima27","ima28","ima29","ima30"
260 DATA "ima31","ima32","ima33","ima34","ima35","ima36"
270 DATA "ima37","ima38","ima39","ima40","ima41","ima42"
280 DATA "ima43","ima44","ima45","ima46","ima47","ima48"
310 DATA "ima49","ima50","ima51","ima52","ima53","ima54"
320 DATA "ima55","ima56","ima57","ima58","ima59","ima60"
330 DATA "ima61","ima62","ima63","ima64","ima65","ima66"
340 DATA "ima67","ima68","ima69","ima70","ima71","ima72"
350 DATA "ima73","ima74","ima75","ima76","ima77","ima78"
360 DATA "ima79","ima80","ima81","ima82","ima83","ima84"
370 DATA "ima85","ima86","ima87","ima88","ima89","ima90"
380 DATA "ima91","ima92","ima93","ima94","ima95","ima96"
410 DATA "ima97","ima98","ima99","imaa0","imaa1","imaa2"
420 DATA "imaa3","imaa4","imaa5","imaa6","imaa7","imaa8"
430 DATA "FINE"
    
```

Figura 7 - Programma di Scorrimento Figurine. Quando si dispone di una raccolta di figurine è bene utilizzare un album in cui siano visibili identificabili o con il loro nome o con un numero da utilizzare al momento opportuno.





Figura 10 - Tipico Esempio di Set di Caratteri. Ogni programma grafico dispone di più set di caratteri che vanno richiamati ed utilizzati durante la composizione del disegno. Quello riprodotto appartiene al pacchetto Energraphics.

calcolare 10 volte fino a 30.000 impiega 253 secondi in interprete e 7 secondi in compilatore, con una velocizzazione di 36 volte. Si noti che si è presa la precauzione di dichiarare le variabili in modo da ottimizzare la compilazione.

Il secondo è un classico programma di grafica tridimensionale, apparso più o meno in questa forma su MCmicrocomputer, che calcola e visualizza un solido nello spazio. Risultati della prova 189 secondi l'interprete e 45 il compilato, ovvero oltre 4 volte.

Di tale programma presentiamo anche una hardcopy (fig. 6) ottenuta attraverso la funzionalità permessa dal programma GRAPHICS, presente sul dischetto DOS, che rende attivo il tasto Shift PrtSc anche in modalità grafica.

L'ultimo programma è tutto grafico,

in quanto carica una figura da un file sequenziale su una matrice, e la sposta per tutto il video con delle PUT, successive (per comprendere il programma si può vedere l'articolo sullo scorso numero di MC). I risultati sono stati identici 72 secondi per tutte e due le versioni, e i tempi sono calcolati al netto della routine di caricamento da disco.

In conclusione si può dire che la programmazione Basic deve tener conto del fatto che il programma, nella versione definitiva, venga compilato o meno. In quanto non solo la compilazione richiede il rispetto di alcune regole all'interno del «sorgente» del programma, ma anche perché la prestazione di ciascuna routine sarà influenzata dalla compilazione e quindi deve essere opportunamente tarata.

Questo discorso vale in particolar

modo nei programmi di grafica interattiva che vanno tarati sui tempi di risposta dell'utente medio.

Per esemplificare quanto detto abbiamo realizzato il programma FITRE, listato in figura 3 e la cui Hard-Copy è in figura 4, in cui il joystick viene utilizzato per creare l'effetto brush.

L'effetto brush è quello che si ottiene con l'areografo ovvero con la pistola a spruzzo, e serve, nella grafica tridimensionale, per ottenere effetti di chiaroscuro oppure sfumature di colore.

Per simulare questo effetto occorre calcolare in modo casuale una «rosa» di pixel attorno alla posizione individuata dal cursore. L'insieme di questi calcoli rende la routine «lenta» e quindi inutilizzabile. La compilazione invece, velocizzando i calcoli, aumenta la «produzione di pixel» di almeno 10 volte e rende il tutto utilizzabile concretamente.

```

100 REM mattonella
110 SCREEN 2:OPTION BASE 1:CLS
130 OPEN "lmaxx.dat" FOR INPUT AS 1
140 INPUT #1,D:DIM AX(D)
150 FOR I=1 TO D:INPUT #1,AX(I):NEXT I:CLOSE #1
160 LINE (0,0)-(622,155),,BF
200 FOR X=0 TO 550 STEP 89
210 FOR Y=0 TO 150 STEP 52
220 PUT (X,Y),AX:NEXT Y:NEXT X
    
```

Figura 12 - Listato Programma Mattonella. Una volta editato il modulo e memorizzato in un file sequenziale, si trasferisce in un vettore e poi con la istruzione PUT si posiziona tramite un loop il cui step corrisponda alla dimensione della mattonella.

Il listato del programma è simile ai due precedenti, varia solo l'uso che viene fatto delle coordinate individuanti la posizione del cursore su video.

Raccolta di immagini

Come abbiamo più volte detto per la realizzazione di immagini si possono seguire tre strade. Quella di servirsi di un pacchetto grafico scelto tra i tantissimi sul mercato. Quella di realizzare in proprio un programma autore che disponga di tutte le funzionalità che ci interessano, e infine la strada intermedia, che è quella di utilizzare il più possibile il buon materiale reperibile, realizzando in proprio le funzionalità che ne permettono l'uso.

L'utilizzazione del materiale reperito può essere semplificata se di tale materiale si ha una completa conoscenza, ovvero si sa cosa permette di fare ad esempio di quali set di caratteri oppure di quali immagini preconfezionate disponga. Per cui nel momento in cui serve una certa immagine ci si ricorda di dove sta e in qualche modo la si preleva.

Un sistema più elegante e anche più produttivo se si fa spesso grafica di

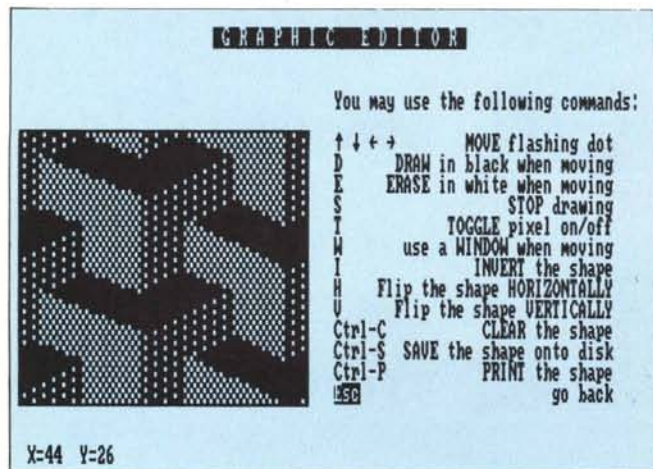


Figura 11
Tipico esempio di Graphic Editor.

Per mezzo di un set di comandi, abbastanza numerosi, si compone un modulo elementare, da utilizzare nella composizione del disegno. Quello riprodotto appartiene al pacchetto PrintMaster.

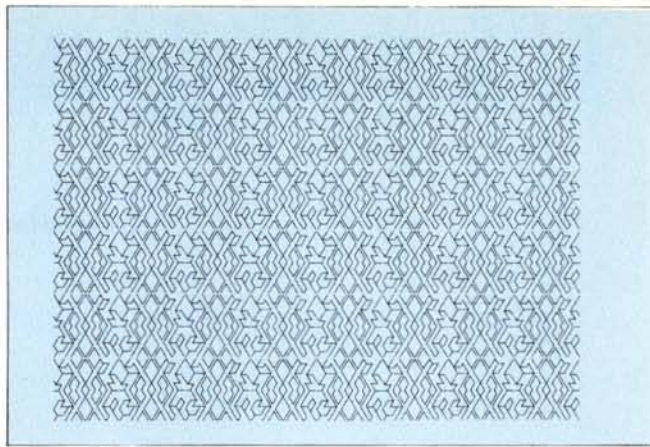
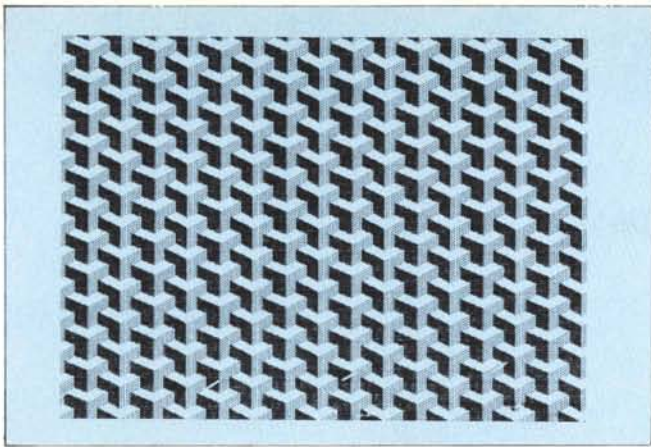


Figure 13 e 14 - Due Output Programma Mattonella. Il concetto è quello di realizzare moduli che conservino la continuità del disegno quando vengono affiancati. L'effetto finale è un disegno senza soluzioni di continuità.

presentazione è quello di raccogliere e catalogare immagini, set di caratteri ecc. e di renderli immediatamente disponibili per il software di nostra produzione. Tale attività diventa molto simile al collezionismo o meglio alla raccolta di figurine che allietava la nostra infanzia.

Per esemplificare tale attività abbiamo utilizzato un prodotto grafico che a sua volta contiene un set di immagini e che le utilizza per la stampa di festoni, cartoncini di auguri, ecc. Il PrintMaster. Le numerose immagini hanno tutte lo stesso formato per cui si prestano ad essere raccolte e catalogate con facilità.

In pratica le abbiamo estratte dallo schermo grafico e le abbiamo immesse in file sequenziali, con le modalità citate nell'articolo pubblicato sul numero scorso, in modo da renderle maneggevoli con comandi grafici del Basic IBM.

Se le immagini sono tante va organizzato un programma di catalogazione che le visualizzi in sequenza e ne mostri il codice di riferimento da richiamare nelle successive utilizzazioni dell'immagine stessa. Per cui si sa che un certo soggetto, ad esempio il tachino che appare nella figura 8, che mostra una pagina del catalogo, è richiamabile tramite il suo codice 7.

Nelle figure 8 e 9 vediamo due pagine del catalogo di figure, prese come detto dal PrintMaster, e in figura 7 vediamo il programma che ne produce lo scorrimento e che si chiama catalogo.

Il programma non presenta alcuna difficoltà in quanto le singole immagini sono di pari formato e quindi la loro collocazione nella videata può essere eseguita con un semplice loop che ne calcola le varie posizioni.

In particolare il loop realizzato (righe 170-190 del listato) ottiene il triplice scopo di battezzare l'immagine, di posizionarla sul video e di permettere

la sovrapposizione progressiva delle immagini, se queste sono più numerose di quelle che appaiono in una singola videata.

Un'altra cosa che vale la pena notare è la necessità di «azzerare» al volo la matrice A%, in quanto è su di essa che vengono caricate via via le varie immagini che possono essere di dimensioni variabili e che quindi richiedono un dimensionamento «su misura». Non è possibile sovradimensionare la matrice in quanto è con essa che vengono eseguiti i comandi PUT e quindi verrebbe visualizzata anche la parte vuota di dati.

Set di caratteri e Templates

Lo stesso discorso fatto per le figurine vale per i set di caratteri, in quanto ogni carattere può essere variamente composto. La differenza sta nel fatto che l'utilizzazione del set di caratteri deve avvenire tramite tastiera, in quanto la maniera più semplice per scrivere un carattere, ad esempio il carattere A in stile gotico, è tuttora quello di premere la A della tastiera.

Ma poiché la A non è che una figurina realizzata in modo grafico, si utilizza lo stesso sistema per quei particolari disegni, ad esempio i Flow Chart, o i disegni Elettronici, composti da tanti piccoli elementi, ciascuno dei quali può essere associato alla tastiera.

Il programma autore dovrà gestire il posizionamento sul video del cursore e l'accoppiamento tra il tasto premuto e il disegno da richiamare.

Il Basic IBM, nel lavoro in modalità grafica, permette di costruire un proprio set di caratteri grafici (che occupano la parte superiore della tabella ASCII). Nel disco DOS è presente un set di caratteri già confezionati che riproducono l'analogo SET del video monocromatico. La limitazione sta nel fatto che tale SET o quello che si può costruire sono sempre limitati ad una

matrice di 8 pixel per 8 pixel. Se si sceglie la strada di utilizzare set di caratteri più sofisticati, autocostruiti o prelevati da qualche pacchetto, è sempre possibile in Basic realizzare una piccola routine che leghi ciascun tasto al relativo carattere.

Tale routine va però integrata con le funzioni legate ai tasti speciali, come il RETURN il BACKSPACE, lo SPACE, per cui diventa alquanto complessa.

In figura 10 presentiamo un tipico set di caratteri grafici prelevato dal prodotto Energraphics e in figura 11 presentiamo un tipico esempio di Graphic Editor, prelevato dal PrintMaster, con il quale si possono confezionare propri caratteri o disegni.

Programma Mattonella

Un'ultima applicazione, realizzata così tanto per divertirsi, è il programma MATTONELLA, listato in figura 12 e con il quale sono stati realizzati i due output di figure 13 e 14.

Il concetto è quello di costruire una figura che consenta di essere accostata in tutte le direzioni ad altre uguali e che costruisca con queste un disegno unico, in cui non si vedano quindi discontinuità.

Il disegno base si può realizzare con un editor tipo quello di figura 11, e poi occorre inserirlo in un programma Basic, che tramite due loop, uno lungo la X e l'altro lungo la Y, calcoli le posizioni esatte in cui sistemare la singola mattonella.

L'esempio è stato realizzato su un IBM dotato di scheda Hercules e l'output è, anche in questo caso, L'HardCopy. Infatti le funzioni PUT e GET, fortunatamente, sono valide nei vari ambienti hardware, PC IBM screen 1, PC IBM screen 2, PC IBM con Hercules. Ed è anche possibile trasferire le picture via file da un ambiente hardware ad un altro.