



di Francesco Petroni

Grafica senza ... input

seconda parte

Con il numero scorso abbiamo iniziato la pubblicazione di una serie di articoli intitolati "ricominciamo da capo", tendenti a ripercorrere da capo l'argomento Computer Grafica, da sempre trattato su MC.

Lo scopo è sia quello di coordinare la materia in un unico discorso a puntate per la comprensione del quale non deve essere necessaria alcuna conoscenza preliminare, sia quello di recuperare all'interesse per la Computer Grafica anche gli ultimi arrivati, neofiti della Computermania.

In questo numero parleremo di quelle applicazioni di C.G. che non richiedono operazioni di input.

Se vogliamo disegnare una figura con il computer, per esempio sul video o sul plotter, dobbiamo necessariamente immettere i dati nel programma di visualizzazione e dobbiamo controllare che i dati immessi siano corretti.

Ad esempio il disegno di una piantina di un appartamento richiede un grosso volume di dati in input e basta che uno di essi sia sbagliato per pregiudicare l'output sia su video dove è teoricamente possibile "cancellare" sia, a maggior ragione, su plotter.

Viceversa il disegno di una superficie spaziale derivata da una formula matematica non richiede input in quanto basta definire in qualche parte del programma la formula e i limiti entro i quali calcolarla.

Sono inoltre disegni molto complessi per il calcolatore in quanto necessitano di un gran volume di calcoli e, in fase di output, di un gran volume di elementi.

La semplicità del programma e la complessità dell'output spiegano la loro diffusione in sede di dimostrazione di capacità grafiche di una macchina. Al punto che qualcuna di queste funzioni è ormai "diventata famosa" solo per la sua "fotogenia" e non per la sua effettiva utilità in questa o quella applicazione.

Un po' di teoria

A questo punto occorre fare un po' di

teoria. Ci riferiamo alla tabellina di figura 1, nella quale sono codificati i quattro casi canonici di curve/superfici sul piano e nello spazio e di sistemi di riferimento cartesiani e polari.

Il discorso sui sistemi di riferimento è stato fatto più volte e quindi lo sintetizziamo.

Esistono due sistemi di riferimento, quello polare e quello cartesiano. Il computer lavora sempre in un sistema di riferimento cartesiano e quindi anche se la curva è riferita ad un sistema polare occorre fare una trasformazione in cartesiano secondo le formule della tabellina 2.

In generale tutte le curve legate al concetto di rotazione (anche la circonferenza)

conviene esprimerle in coordinate polari in quanto la variabile è proprio l'angolo di rotazione e questo semplifica la formula.

L'inconveniente principale è che nelle formule di traduzione polare/cartesiane intervengono le espressioni trigonometriche (seno, coseno, tangente e inverse come arcotangente, ecc.) e di questo non si può fare a meno.

Un po' di Trigonometria

Chi vuole realizzare grafica con il computer per visualizzare funzioni sul piano e nello spazio deve saper maneggiare un po' le espressioni trigonometriche e questo fortunatamente per lui non vuol dire che deve conoscere la trigonometria.

tab. 1	CURVE NELLO SPAZIO BIDIMENSIONALE	SUPERFICI NELLO SPAZIO TRIDIMENSIONALE
COORDINATE CARTESIANE	punto $P(x, y)$ funz. $Y = f(X)$	punto $P(x, y, z)$ funz. $Z = f(X, Y)$
COORDINATE POLARI	punto $P(R, \alpha)$ funz. $R = f(\alpha)$	punto $P(R, \alpha, \beta)$ funz. $R = f(\alpha, \beta)$

Figura 1 - Tabella 1. Casistica delle funzioni rappresentabili sul piano e nello spazio espresse in coordinate cartesiane e polari.

tab. 2.A	TRASFORMAZIONI DA COORDINATE POLARI A CARTESIANE BIDIMENSIONALI	$X = R * \cos(\alpha)$ $Y = R * \sin(\alpha)$
tab. 2.B	TRASFORMAZIONI DA COORDINATE POLARI A CARTESIANE BIDIMENSIONALI	$X = R * \cos(\alpha) * \sin(\beta)$ $Y = R * \sin(\alpha) * \sin(\beta)$ $Z = R * \cos(\beta)$
tab. 2.C	TRASFORMAZIONI DA COORDINATE CARTESIANE A POLARI BIDIMENSIONALI	$R = \sqrt{X^2 + Y^2}$ $\alpha = \text{ATN}(X, Y)$

Figura 2 - Tabella 2. Formule per il passaggio di coordinate da polari a cartesiane e viceversa.

prg.	radian.	gradi	sin	cos
0	0.00000	0.00	0.00000	1.00000
1	0.26180	15.00	0.25882	0.96593
2	0.52360	30.00	0.50000	0.86603
3	0.78540	45.00	0.70711	0.70711
4	1.04720	60.00	0.86603	0.50000
5	1.30900	75.00	0.96593	0.25882
6	1.57080	90.00	1.00000	0.00000
7	1.83260	105.00	0.96593	-0.25882
8	2.09439	120.00	0.86603	-0.50000
9	2.35619	135.00	0.70711	-0.70711
10	2.61799	150.00	0.50000	-0.86603
11	2.87979	165.00	0.25882	-0.96593
12	3.14159	180.00	0.00000	-1.00000
13	3.40339	195.00	-0.25882	-0.96593
14	3.66519	210.00	-0.50000	-0.86603
15	3.92699	225.00	-0.70711	-0.70711
16	4.18879	240.00	-0.86603	-0.50000
17	4.45059	255.00	-0.96593	-0.25882
18	4.71239	270.00	-1.00000	-0.00000
19	4.97419	285.00	-0.96593	0.25882
20	5.23599	300.00	-0.86603	0.50000
21	5.49779	315.00	-0.70711	0.70711
22	5.75959	330.00	-0.50000	0.86603
23	6.02139	345.00	-0.25882	0.96593
24	6.28319	360.00	-0.00000	1.00000

Figura 3 - Tabella Corrispondenza. La tabella mette in corrispondenza angoli espressi in radianti e in gradi. Inoltre il programma carica una tabella, in questo caso di 24 elementi, dove sono precalcolati seno e coseno per una loro ricerca rapida.

Deve sapere che nelle espressioni $Y = \text{SIN}(X)$ e $Y = \text{COS}(X)$ la X rappresenta un angolo, che questo angolo va espresso in radianti e che dato un valore alla X il corrispondente valore della Y è sempre compreso tra -1 e $+1$.

Questo significa pure che non vi è corrispondenza biunivoca tra X e Y , cioè dato un valore alla X vi è un solo valore di Y , ma

non vale il viceversa. Ovvero data la Y (che come detto deve essere compresa tra -1 e $+1$) esistono infiniti valori di X che differiscono tra di loro di un multiplo di 360 gradi; ed è valido sempre che $\sin(x) = \sin(\pi - x)$ e $\cos(x) = \cos(2\pi - x)$.

In generale tutti i Basic accettano nelle funzioni trigonometriche gli angoli espressi in radianti mentre la nostra cultura "sco-

lastica" ci porta a ragionare in gradi sessagesimali. L'angolo di 90° altrimenti noto come angolo retto corrisponde a 1.570796 radianti, $\pi/2$.

L'esecuzione delle funzioni trigonometriche è una delle operazioni più lente del Basic, quindi se in un programma occorre eseguire più volte tali operazioni sempre sugli stessi angoli è opportuno prendere provvedimenti.

Il provvedimento più classico è quello di precalcolare ed inserire in una tabella una serie completa di seni e coseni. Tanto più lunga è questa tabella tanto più sarà preciso il calcolo di funzioni trigonometriche che faccia ricorso a questa tabella.

Calcolo di una funzione

Questo paragrafo è destinato esclusivamente ai non esperti.

Grafica senza input significa che i dati da visualizzare non sono immessi dall'operatore, cosa che deve sempre avvenire se si disegna una figura qualsiasi, come ad esempio una piantina di una casa, oppure un pupazzo o un viso, ma sono direttamente calcolati dal computer che dopo averli calcolati provvede anche a visualizzarli in forma grafica, secondo le specifiche che abbiamo dato al programma.

Ma perché il computer possa calcolare e poi disegnare bisogna comunque immettere qualcosa. Definiremo questo qualcosa FUNZIONE, in più occorre immettere i limiti entro i quali eseguire i calcoli.

Una funzione si esprime nella forma $Y = f(X)$, il che vuol dire che dato un valore alla X , calcolando la funzione $f(X)$, si trova il corrispondente valore della Y . L'altra informazione necessaria è per quali valori di X vogliamo calcolare e visualizzare la funzione.

Facciamo un esempio con la funzione $Y = 2 \cdot X + 5$. I più esperti sanno che questa è l'equazione di una retta sul piano. Per giocare un po' con questa funzione si può definire un loop sulla X , all'interno di questo loop calcolare la Y corrispondente e poi stamparne i valori in forma numerica. Il programma sarà:

```
10 FOR X = -10 TO 10
20 Y = 2 * X + 5
30 PRINT X,Y
40 NEXT X
```

Espresso in questa forma il programma visualizzerà tutti i valori X compresi tra -10 e $+10$ e tutti i corrispondenti valori di Y . E farà la stessa cosa qualunque sia la formula, anche la più complessa, immessa in riga 20.

La stessa cosa vale per una funzione di due variabili del tipo $Z = f(X,Y)$. Ovviamente i loop saranno due, uno su X e uno su Y .

Il problema di visualizzare in forma grafica i dati calcolati è un problema successivo, isolabile dal calcolo vero e proprio. Consigliamo i principianti di iniziare creando delle funzioni anche complesse e di vi-

```
100 REM tabella seno, coseno
110 P=3.1415926#;PP=12;PA=PP*2:PQ=P/PP;PG=P/180;GG=360
120 REM P      Pigreco
130 REM PA/PP  Numero Elementi nella Tabella Completa e Meta
140 REM PQ    Corrispondenza tra Tabella e Radianti
150 REM PG    Corrispondenza tra Radianti e Gradi
160 REM S(),C() Tabella Seno e Coseno
170 DIM S(PA),C(PA):FOR I=0 TO PA:S(I)=SIN(I*PQ):C(I)=COS(I*PQ):NEXT I
180 CLS
190 PRINT "prg. radian. gradi sin cos":PRINT
200 FOR I=0 TO PA:PRINT USING "###";I;
210 PRINT USING "#####.#####";I*PQ;:PRINT USING "#####.###";I*GG/PA;
220 PRINT USING "#####.#####";S(I);:PRINT USING "#####.#####";C(I):NEXT I
230 CLS
240 INPUT "Angolo in Gradi " ,AG:IF AG=999 THEN 290
250 AG=AG-INT(AG/GG)*GG:PRINT "Angolo in Gradi " ,AG
260 AR=PP*AG:PRINT "Angolo in Radianti " ,AR
270 PRINT "Elementi in Tabella " ,PA
280 NE=INT(.5+AR/PQ):PRINT "Elemento della Tabella " ,NE:PRINT:GOTO 240
290 CLS
300 DEF FNA(A)=INT(.5+PG*(A-INT(A/GG)*GG)/PQ)
310 INPUT "Angolo in Gradi " ,AG:IF AG=999 THEN 350
320 PRINT "Seno " ,USING "###.#####";S(FNA(AG))
330 PRINT "Coseno " ,USING "###.#####";C(FNA(AG))
340 PRINT :GOTO 310
350 CLS:END
```

Figura 4 - Programma TABTRIG - Listato. Il programma genera la tabella precedente. Inoltre contiene due routine per la trasformazione degli angoli immessi via input.

sualizzarne i valori numerici prima di affrontare la parte grafica.

La seconda fase comporta un solo problema, quello del formato, che si può risolvere in varie maniere, a seconda del verificarsi di due condizioni possibili:

- 1 — il calcolatore possiede o non possiede funzioni di scaling
- 2 — i valori calcolati rientrano nel formato video.

In sostanza se i valori numerici calcolati rientrano nel formato video non esiste un problema di visualizzazione.

Se non rientrano possono essere modificati, con una operazione che si chiama scaling, per farceli rientrare.

Oppure, se il computer lo consente, si opera una "compressione od espansione" della finestra video. Il che in pratica equivale all'operazione precedente solo che la fa direttamente il computer.

Programma TABRIG (tabella valori trigonometrici)

Il programma TABRIG (listato in figura 4) esemplifica l'intera problematica. Costruisce una tabella lunga a piacere (occorre modificare il valore PP) che mette in corrispondenza per ogni progressivo, l'angolo in radianti, l'angolo in gradi e i valori

Angolo in Gradi	60
Angolo in Radianti	1.047197
Elementi in Tabella	24
Elemento della Tabella	4
Angolo in Gradi	90
Angolo in Radianti	1.570796
Elementi in Tabella	24
Elemento della Tabella	6
Angolo in Gradi	7
Angolo in Radianti	.122173
Elementi in Tabella	24
Elemento della Tabella	0
Angolo in Gradi	8
Angolo in Radianti	.1396263
Elementi in Tabella	24
Elemento della Tabella	1
Angolo in Gradi	45
Seno	0.70711
Coseno	0.70711
Angolo in Gradi	90
Seno	1.00000
Coseno	0.00000
Angolo in Gradi	0
Seno	0.00000
Coseno	1.00000
Angolo in Gradi	

Figura 5 - Programma TABRIG - Output. È rappresentata l'hard copy delle videate del programma di traduzione angoli. L'angolo va immesso in gradi e viene restituito in radianti, inoltre viene ricercato l'elemento della tabella più vicino.

```

100 REM Campionario di Superfici nello Spazio
110 GOSUB 470:SCREEN 1:COLOR 1,0:CLS:S=2
120 TH=.3:S1=SIN(TH):C1=COS(TH):PH=.4:S2=SIN(PH):C2=COS(PH)
130 REM loop principale
140 FOR Q=1 TO 10:GOSUB 290
150 REM meridiani
160 FOR Y=-20 TO 20 STEP S:FL=0:FOR X=-20 TO 20 STEP S
170 ON Q GOSUB 340,350,360,370,380,390,400,410,420,430
180 GOSUB 270:IF FL=0 THEN FL=1:PSET (SX,SY)
190 LINE -(SX,SY):NEXT X:NEXT Y
200 REM paralleli
210 FOR X=-20 TO 20 STEP S:FL=0:FOR Y=-20 TO 20 STEP S
220 ON Q GOSUB 340,350,360,370,380,390,400,410,420,430
230 GOSUB 270:IF FL=0 THEN FL=1:PSET (SX,SY)
240 LINE -(SX,SY):NEXT Y:NEXT X:NEXT Q
250 GOSUB 440:END
260 REM routine
270 REM da tridim. a bidim.
280 SX=X1*C1-Y1*C2:SY=X1*S1+Y1*S2+Z1:RETURN
290 REM pulizia schermo e tracciamento riferimenti
300 PRINT CHR$(7)
310 I$=INKEY$:IF I$="" THEN 310
320 CLS:WINDOW(-10,-6)-(10,6):LINE (-10,-6)-(10,6),2,BF
322 LOCATE 2,2:PRINT Q:RETURN
330 REM funzioni
340 X1=X/5:Y1=Y/5:Z1=(1-COS(X1))*(1-COS(Y1))-3:RETURN
350 X1=X/6:Y1=Y/6:Z1=.4*COS(X1*X1+Y1*Y1):RETURN
360 X1=X/4:Y1=Y/4:Z1=EXP(.02*(X1*X1+Y1*Y1))-3:RETURN
370 X1=X/4:Y1=Y/4:Z1=.02*((X1*Y1*Y1)-(Y1*X1*X1)):RETURN
380 X1=X/4:Y1=Y/4:Z1=.02*((X1*Y1*Y1)+(Y1*X1*X1)):RETURN
390 X1=X/4:Y1=Y/4:Z1=3*SIN(X1/2)*SIN(Y1/3):RETURN
400 X1=X/4:Y1=Y/4:K=(1-X1*X1/5-Y1*Y1/7):Z1=SGR(2*K*(SGN(K)+1)):RETURN
410 X1=X/6:Y1=Y/6:Z1=.02*SGR(X*X*Y*Y/2)-3:RETURN
420 X1=X/5:Y1=Y/5:Z1=(1-SIN(1.8*X1))*(1-COS(1.2*Y1))-2:RETURN
430 X1=X/5:Y1=Y/5:K=X1*X1+Y1*Y1:Z1=.3*(1+COS(K))*(SGN(9.87-K)+1):RETURN
440 REM switch to mono
450 DEF SEG=0:POKE &H410,(PEEK(&H410) OR &H30)
460 SCREEN 0:WIDTH 40:WIDTH 80:RETURN
470 REM switch to color
480 DEF SEG=0:POKE &H410,(PEEK(&H410) AND &HCF) OR &H10
490 SCREEN 0:WIDTH 40:SCREEN 1,0,0,0:RETURN

```

Figura 6 - Programma CAMPIONARIO - Listato. Il programma contiene le formule relative a dieci superfici spaziali espresse nella forma $Z = f(X, Y)$. Le superfici poi vengono visualizzate in sequenza.

seno e coseno. Questi valori sono poi memorizzati in due vettori S(I), C(I) richiamati direttamente tramite l'indice.

È noto che è molto più rapido l'accesso ad un vettore dato l'indice che non il calcolo, anche se avviene direttamente tramite l'istruzione Basic, di una funzione trigonometrica.

Il programma produce l'output di fig. 3. Nell'output pubblicato la tabella contiene 24 dati e cioè gli angoli differiscono di 15 gradi l'un l'altro. Quindi entrando ad esempio con l'angolo in gradi, con la funzione di riga 300, questo angolo si trasforma nel progressivo corrispondente della tabella dal quale si ricavano seno e coseno. La trasformazione che dato l'angolo trova l'indice della tabella si può eseguire tramite una funzione definita con una DEF FN.

La precisione del calcolo dipende dall'estensione della tabella. Con 24 dati l'errore massimo è di 7 gradi, con una tabella di 360 dati l'errore è di 0.5 gradi.

Il programma prosegue con due routine di trasformazione degli angoli, dati via input, nei valori corrispondenti nella tabella. Di tali programmi in figura 5 sono riportate le due hardcopy.

La prima accetta l'angolo in gradi, lo riduce all'interno dell'angolo giro (se l'input è maggiore di 360 gradi), ne ricava il valore in radianti e ne ricava il progressivo corrispondente nella tabella.

Notare come in una tabella di 24 dati l'angolo di 7 gradi viene assegnato al primo elemento della tabella e l'angolo di 8

gradi al secondo, e nei due casi si ha l'errore massimo di 7 gradi.

La seconda routine riceve in input l'angolo in gradi; poi, tramite la funzione di traduzione di riga 300, trova il puntatore della tabella e quindi il seno e il coseno.

Campionario

È un programma (listato in figura 6) con uscita su video a colori del PC IBM in modalità SCREEN 1. Produce 10 output differenti (da cui il nome Campionario) di superfici nello spazio.

Lo stato SCREEN 1 permette una definizione di 320 per 200 pixel singolarmente indirizzabili in quattro colori definibili in varie tavolozze.

Le modalità di visualizzazione, le cui specifiche sono definite in riga 120, non variano per cui è stato possibile definire ciascuna superficie in una sola riga di programma (righe dalla 330 alla 430).

Vi è quindi un loop generale da 1 a 10, aperto in riga 140 e chiuso in riga 240, all'interno del quale, per ogni valore di Q, viene visualizzata la funzione corrispondente.

La modalità di calcolo per tradurre le coordinate tridimensionali della superficie in coordinate video è quella detta meridiani/paralleli con la quale vengono modificati due volte due loop, la prima volta su X e Y e la seconda su Y e X (rispettivamente righe 160 e 210) ed in corrispondenza di ogni coppia di valori X e Y, tramite il

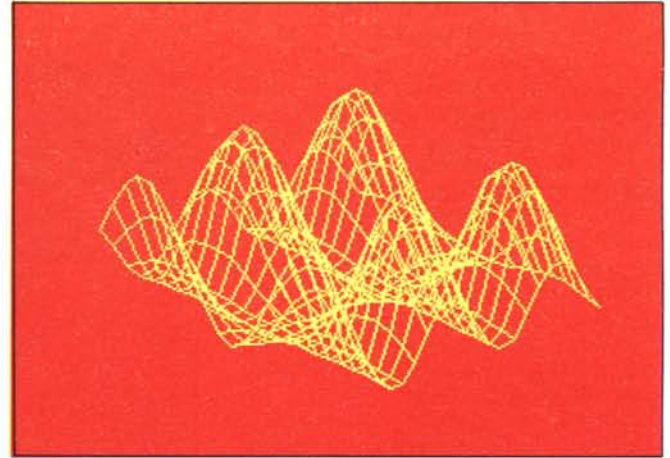
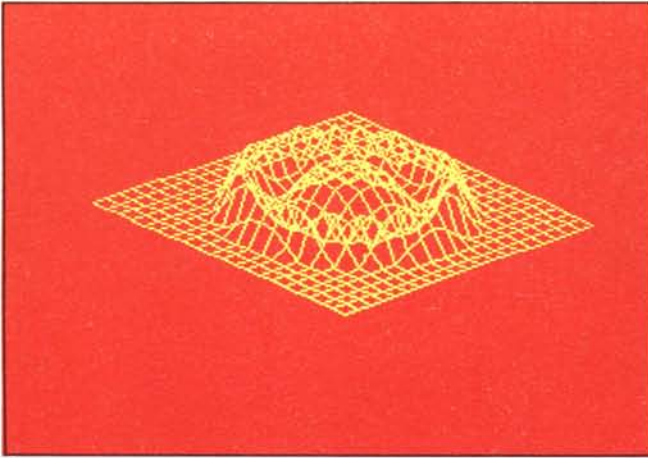


Figure 7/8 - Programma CAMPIONARIO - Due output su video a colori. La formattazione dell'output è talmente semplificata che in pratica la formula di ciascuna curva occupa una sola riga di programma.

richiamo alla subroutine di funzione, viene calcolato il valore Z relativo. I tre valori vengono tradotti in coordinate schermo dalla routine di riga 280.

Le routine generali sono inoltre quelle di pulizia e formattazione schermo, che produce una cornice e visualizza il numero progressivo della funzione (riga 290 e successive).

Ci sono poi le due routine (righe 440 e 470) per il passaggio da uscita su video a colori a uscita su bianconero e viceversa. Queste vanno inserite solo se oltre al video a colori comunque indispensabile per l'uscita grafica si lavori anche con un video monocromatico.

Dei dieci output ne pubblichiamo solo due nelle figure 7 e 8.

Programma EPI & IPOICCLOIDE

C'è una famiglia di curve sul piano che si chiama cicloide. La curva cicloide è quella descritta da un punto rigidamente collegata ad un cerchio che rotola. Se tale cerchio rotola su una linea a sua volta circolare la curva risultante prima o poi si richiude su se stessa, in relazione al valore del rapporto tra i due raggi.

Esistono due sottofamiglie EPI e IPO a

seconda che il cerchio mobile rotoli all'esterno o all'interno del cerchio fisso. Il risultato grafico che si presta a numerose varianti con il variare dei due raggi è sempre gradevole.

Il programma EPI & IPO (listato in figura 9) sempre in output sul monitor a colori

dell'IBM, produce le due curve (fig. 10 e 11), queste vengono definite in riga 130 tramite i valori A e B che sono i due raggi e tramite i valori C e D che sono due valori derivati dai primi e che appaiono nelle formule delle cicloidi.

Poiché a parità di raggi la Epicicloide

```

100 REM Curve sul Piano Epicicloidi e Ipocicloidi
110 GOSUB 300:SCREEN 1:COLOR 1,0:P=3.1415926#
120 CLS:WINDOW(-20,-12)-(20,12):LINE (-20,-12)-(20,12),2,BF
130 A=4:B=1.8:C=(A+B)/B:D=(A-B)/B
140 Q=0:GOSUB 230:PSET (X,Y)
150 I#=INKEY#:IF I#<>" " THEN 170
160 Q=Q+.1:GOSUB 230:LINE -(X,Y):GOTO 150
170 GOSUB 270:LOCATE 20,1:INPUT "Premi RETURN per Continuare ",S#
180 GOSUB 300:SCREEN 1:COLOR 1,0:P=3.1415926#
190 CLS:WINDOW(-8,-6)-(8,6):LINE (-8,-6)-(8,6),2,BF
200 Q=0:GOSUB 260:PSET (X,Y)
210 I#=INKEY#:IF I#<>" " THEN GOSUB 270:END
220 Q=Q+.1:GOSUB 260:LINE -(X,Y):GOTO 210
230 REM prima funzione
240 X=1.5*(A+B)*COS(Q)-B*COS(C*Q):Y=(A+B)*SIN(Q)-B*SIN(C*Q):RETURN
250 REM seconda funzione
260 X=1.5*(A-B)*COS(Q)+B*COS(D*Q):Y=(A-B)*SIN(Q)-B*SIN(D*Q):RETURN
270 REM switch to mono
280 DEF SEG=0:POKE &H410,(PEEK(&H410) OR &H30)
290 SCREEN 0:WIDTH 40:WIDTH 80:RETURN
300 REM switch to color
310 DEF SEG=0:POKE &H410,(PEEK(&H410) AND &HCF) OR &H10
320 SCREEN 0:WIDTH 40:SCREEN 1,0,0,0:RETURN

```

Figure 9 - Programma EPI & IPO (... CICLOIDE) - Listato. Sono due famiglie di curve sul piano esprimibili in coordinate polari.

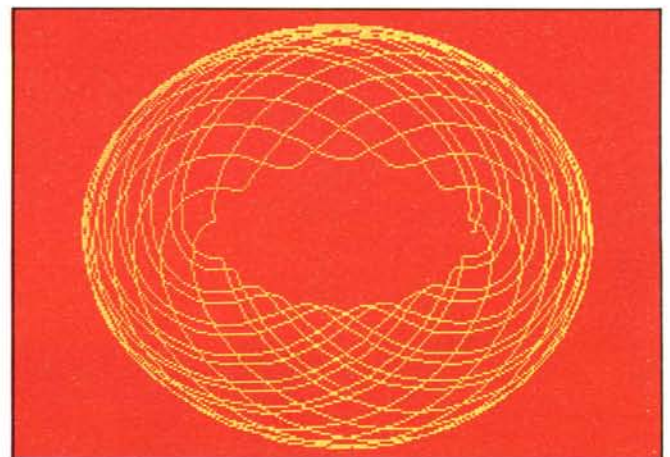
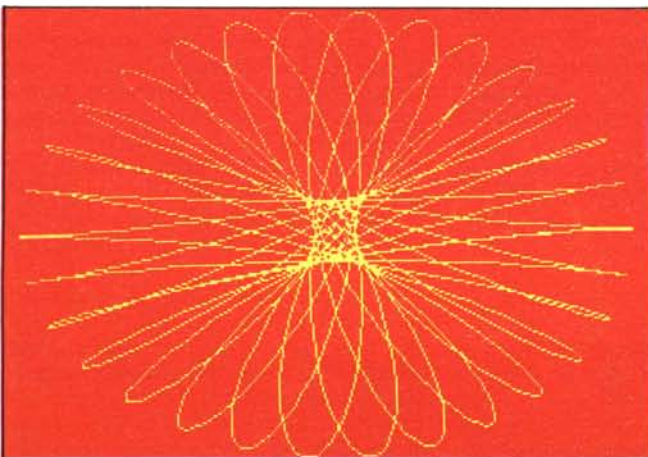


Figure 10/11 - Programma EPI & IPO - Due output su video a colori. Tra le curve bidimensionali quelle di rotazione sono le più facili e le più ... fotografiche.

```

100 REM INIZIALIZZAZIONE
110 D$ = CHR$(4): PL = 0: REM SWITCH TO PLOT
120 IF PL = 0 THEN HGR2: HCOLOR= 3
130 A = 300: B = 280: C = (A + B) / 2: D = 6 * A
140 R = 3 * A: XC = D / 2: YC = D / 2
150 J = 1: GOSUB 410: REM CORNICE QUADRATA
160 GOSUB 380
170 J = 2: GOSUB 410: REM CORNICE CIRCOLARE
180 GOSUB 440
190 J = 3: GOSUB 410: REM MAIN LOOP
200 GOSUB 280
210 GOSUB 320
220 Q = Q + 1
230 IF Q > 88 THEN END
240 GOSUB 310
250 GOSUB 280
260 GOSUB 350
270 GOTO 220: REM RITORNO
280 REM CALCOLO COORDINATE
290 X = INT ((A + B) * COS (Q) - B * COS (C * Q) + XC + 5)
300 Y = INT ((A + B) * SIN (Q) - B * SIN (C * Q) + YC + 5)
310 RETURN
320 REM MOVE
330 IF PL = 0 THEN HMOVE: RETURN
340 PRINT D$"PR#4": PRINT "M", X, ", ", Y: PRINT D$"PR#0": RETURN
350 REM DRAW
360 IF PL = 0 THEN HMOVE: RETURN
370 PRINT D$"PR#4": PRINT "D", X, ", ", Y: PRINT D$"PR#0": RETURN
380 REM BOX
390 IF PL = 0 THEN HMOVE: RETURN
400 PRINT D$"PR#4": PRINT "%1", " ", D: PRINT D$"PR#0": RETURN
410 REM CAMBIO PENNA
420 IF PL = 0 THEN RETURN
430 PRINT D$"PR#4": PRINT "J", J: PRINT D$"PR#0": RETURN
440 REM ARCO/CERCHIO/SPIRALE/POLIGONO
450 IF PL = 0 THEN RETURN
460 PRINT D$"PR#4": PRINT "W", XC, ", ", YC, ", ", R: PRINT D$"PR#0": RETURN
470 PRINT D$"PR#0": RETURN

```

Figura 12 - Programma APPLE/GRAPHTEC - Listato. È una versione per Apple e Plotter Graphtec del programma EPI & IPO. Lo switch PL invia su Plotter o su Video.

“viene più grande” della Ipocicloide, utilizziamo due Window differenti, la prima in riga 120 e la seconda in riga 190.

Il calcolo delle funzioni e la loro visualizzazione non avviene tramite un loop FOR/NEXT in quanto non è possibile conoscere a priori, a meno di calcoli parecchio complessi, per quale valore di Q, dipendente dai valori A e B, la curva si richiude. Cioè predefinendo il valore finale del loop si rischia o di non chiudere la curva o di ripercorrerla più volte.

Per cui all'interno della routine di incremento dell'angolo Q, c'è la riga 150 (e la

successiva 210) che riconosce il momento in cui viene premuto un tasto qualsiasi. In questo momento l'esecuzione viene interrotta.

Le due funzioni sono rispettivamente nelle routine 240 (è la Epicicloide) e 260 (Ipocicloide) che vengono richiamate dapprima all'inizio della curva, per il valore iniziale dell'angolo Q e poi via via che tale valore viene incrementato.

Il passaggio dalla prima alla seconda curva e la fine del programma avvengono premendo un tasto qualsiasi.

Per disegnare altri “parenti” della fami-

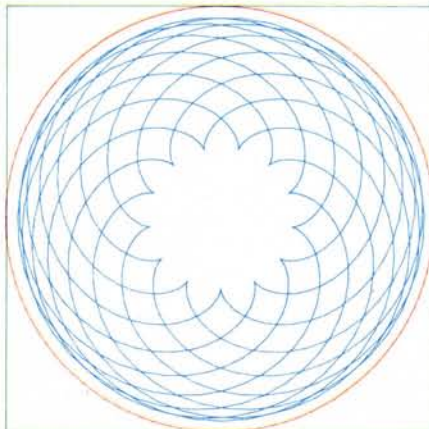
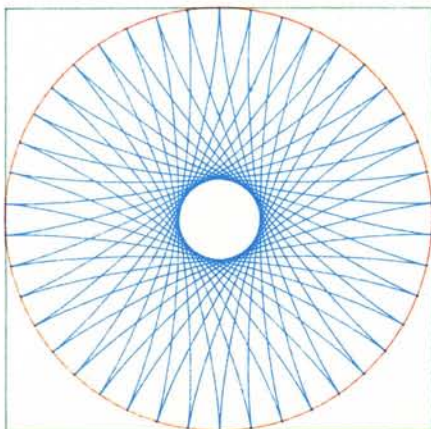


Figura 13/14 - Programma APPLE/GRAPHTEC - Due output su carta. Le uscite su carta, essendo più definite, permettono dei loop più lunghi ed ... estenuanti.

glia cicloidi, occorre cambiare i valori di A e/o di B in riga 130.

Programma EPI & IPO (su Apple II e uscita su Plotter Graphtec)

L'ultimo programma è stato realizzato sull'Apple II, con due uscite differenti: il video grafico, in modalità HGR2, proprio dell'Apple e un Plotter MP 1000 della Graphtec (già Watanabe) con interfaccia parallela Centronics, posta nello slot 4 del computer.

Le due uscite sono indirizzabili tramite lo switch PL, che se è posto a zero, attiva il video, altrimenti attiva l'uscita su plotter.

Altro provvedimento da prendere per differenziare i due output è quello di dimensionare opportunamente i valori A, B di riga 130, in funzione della diversa definizione (280 per 192 pixel, contro 3600 per 2600 decimi di millimetro) delle due modalità di uscita.

Nei nostri due output, l'uno riferito ad una Epi (figura 13) e l'altro riferito ad una Ipo (figura 14), quest'ultima ottenuta variando le formule di righe 290 e 300, viene utilizzata solo la quarta parte della superficie indirizzabile in uscita sul Plotter.

Nel programma sono totalmente isolate le parti relative al calcolo della curva (sub-routine di riga 290) da quelle relative al disegno via Plotter.

Le routine riconosciute dal Plotter, che come noto è visto dal computer come se fosse una stampante, sono varie. Ognuna di queste contiene una uscita attivata nel caso lo switch sia posto a zero.

Le routine sono:

Riga 410 - routine di cambio penna.

Riga 380 - routine di box, utilizzata per il disegno della cornice.

Riga 440 - routine arco/cerchio/spirale/poligono, utilizzata per il disegno della cornice rotonda.

Riga 320 - routine di move, per posizionare la penna all'inizio della curva.

Riga 350 - routine di draw, per tracciare tratti rettilinei.

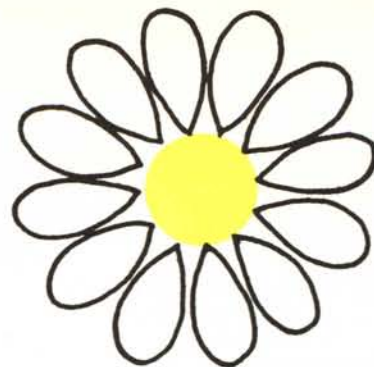
Il programma si divide in tre parti: la prima, al solito, di inizializzazione variabili, pulizia schermo, ecc. (righe 100/140). Vi possiamo comprendere anche il disegno delle due cornici, che avviene solo se utilizziamo il Plotter, e che consiste nel rimando alle due routine 380 e 440.

La seconda parte è il MAIN loop che gira tra le righe 220 e 270, incrementando via via il valore dell'angolo Q (fino a che non superi il valore di chiusura) e rimandando via via alle routine di calcolo dei valori X e Y, e di tracciamento.

La terza ed ultima parte è costituita dall'insieme delle routine di Plot (o su Video o su Plotter).

La seconda delle due figure pubblicate non è ricavata direttamente dal programma listato. Infatti questo prevede in uscita una Epi e per adattarlo ad una Ipo occorre modificare le istruzioni di calcolo X,Y magari copiandole dal programma Epi & Ipo di figura 9.

SHARP



MZ-700

Il Personal Computer più completo e più compatto per la famiglia e per la scuola

La serie MZ 700 impiega una CPU ad alta velocità ed una ampia memoria a 64 KB.

In questo compatto Personal Computer (MZ 731) sono integrati anche un registratore cassette e una stampante-plotter a colori.

Strutturata in maniera precisa e compatta, la serie MZ 700 offre elevate prestazioni per soddisfare le necessità più varie dal campo hobbistico al didattico ed al gestionale.

- Prestazioni elevate ed alta velocità con la CPU Z80A
- Area di memoria programmabile di 64 Kbytes
- Tastiera-Unità centrale sottile e compatta.
- Una varietà di sistemi per ogni necessità:
 - MZ-721... Tastiera-CPU con unità a cassette
 - MZ-731... Tastiera-CPU con unità a cassette e stampante.
 - Plotter a 4 colori

Distribuito



**MELCHIONI
COMPUTERTIME®**

MELCHIONI COMPUTERTIME

20093 COLOGNO MONZESE (MI) - Viale Europa, 49 - Tel. 02/25.35.035 - 25.40.607 - Tlx 310352 METIME

RIVENDITORI E SERVIZI DI ASSISTENZA SU TUTTO IL TERRITORIO NAZIONALE