

di Francesco Petroni

Esercizi di Computer Grafica Tridimensionale

Dopo alcuni articoli di contenuto informativo torniamo alle origini, per lo meno alle origini di questa rubrica, torniamo cioè a trattare argomenti di computer grafica pratica, destinata più agli hobbysti che ai semplici utilizzatori di computer.

Utilizziamo il termine esercizi in quanto come al solito intendiamo cercare di far capire qualche concetto, piuttosto che presentare programmi da copiare passivamente, e in quanto i programmi presentati, in quanto esercizi, non hanno nessuna utilità pratica.

Tratteremo grafica tridimensionale e realizzeremo una serie di programmi, abbastanza simili tra di loro, proprio con l'intento di comprendere il più possibile le varie funzionalità, l'importanza di ciascun elemento in gioco nei calcoli e quindi come il variare di un elemento comporti determinati effetti sul disegno.

Gli elementi che disegneremo sono ... uno solo: il cerchio e i suoi derivati. Anche questa semplificazione andrà a vantaggio della comprensione perché ciò che vedremo è sempre e comunque un cerchio o una figura che nasce dalla formula trigonometrica del cerchio.

Gli esercizi partono da un semplice cerchio sul piano, poi in questo cerchio varieremo alcuni parametri per scoprire quale sia la loro funzione. Trasferiremo il cerchio nello spazio e cominceremo a farne delle copie variando uno o più parametri.

Per quanto riguarda i listati appariranno molto simili tra di loro e certamente per economia sarebbe stato più opportuno assemblarli in un unico programma in cui le routine comuni fossero condivise. Non lo facciamo sia per motivi di comprensione sia per far rimanere la lunghezza del listato entro limiti accettabili, ovvero entro limiti entro i quali anche il lettore più pigro accetta di trascrivere il listato sul proprio computer.

I listati sono scritti in BASICA, ovvero nel Basic del PC IBM. La traduzione per altre macchine consiste nella modifica delle istruzioni di plottaggio del singolo punto PSET (X, Y) e del plottaggio

della singola linea LINE (X, Y) - (X1, Y1).

Più complicato è l'inserimento di una routine di scaling, necessaria per quei Basic che non hanno una istruzione specifica. Per la realizzazione di una routine rimandiamo i lettori più interessati alle puntate precedenti di questa rubrica.

Il cerchio sul piano

Per disegnare un cerchio sul video, quasi tutti i Basic hanno ormai l'istruzione CIRCLE, che necessita di una serie di parametri indispensabili per determinare la sua posizione sul video e la sua dimensione.

Esistono in genere anche parametri opzionali, per mezzo dei quali si possono tracciare con la stessa istruzione CIRCLE, ellissi, archi, settori di cerchi (archi completi di raggi).

Per quanto potente l'istruzione CIRCLE non è adatta ad essere manipolata e quindi non la utilizzeremo. Ricorremo alla formula trigonometrica della circonferenza, che è:

$$X = XC + R \cdot \cos(A)$$
$$Y = YC + R \cdot \sin(A)$$

con R Raggio, XC, YC coordinate del centro e con l'angolo A che varia da 0 a 360, ovvero a 2 PI greco. Per tracciare il cerchio occorre eseguire un loop su A, facendolo partire da 0 e incre-

mentandolo di uno step SA, e facendolo finire a 2*PI greco.

Tanto minore è lo step tanto maggiore sarà l'accuratezza del disegno, che però dipende anche dalla definizione permessa dall'hardware. In altre parole è inutile ridurre lo step, il che oltretutto rallenta la velocità di esecuzione, se l'hardware non ne apprezza i risultati in termini di precisione di disegno.

Ma mentre l'istruzione CIRCLE si ottiene direttamente un disegno sul video, utilizzando le formule trigonometriche si ottengono dei valori numerici che identificano punti sul video. Per visualizzarli si ricorre alle istruzioni standard di plottaggio di un punto (PSET, del solo primo punto) e di plottaggio da un punto ad un altro

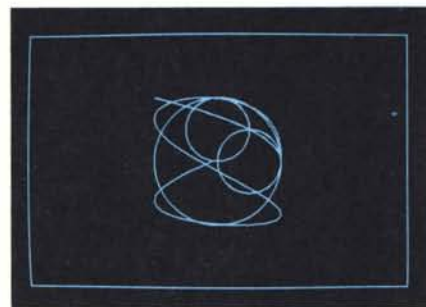


Figura 2 - Circonferenza nel Piano - Output. Le formule vengono utilizzate per cinque curve differenti, strettamente «impastate» tra di loro.

```
100 REM programma 1.0
110 PI=3.14:SP=PI/40:DA=SP*5:R=1:F=0:FOR C=0 TO 4
120 SCREEN 2:CLS:WINDOW (-3,-2)-(3,2):LINE (-3,-2)-(3,2),,B
130 FOR A=0 TO 2*PI STEP SP
140 IF C=0 THEN X=R*COS(A):Y=R*SIN(A)
150 IF C=1 THEN X=R*COS(A)*SIN(A):Y=R*SIN(A)*SIN(A)
160 IF C=2 THEN X=R*COS(A)*COS(A):Y=R*COS(A)*SIN(A)
170 IF C=3 THEN X=R*COS(A^1.3):Y=R*SIN(A)
180 IF C=4 THEN X=R*COS(A):Y=R*SIN(A/2)
190 IF A=0 THEN PSET (X,Y):ELSE LINE-(X,Y)
200 NEXT A:NEXT C:END
```

Figura 1 - Circonferenza nel Piano - Listato. Familiarizzazione con le formule trigonometriche che servono per esprimere una circonferenza nel piano. Formule che utilizzeremo per tutte le figure realizzate.

(-LINE (X,Y) per tutti i segmenti successivi).

Il primo programma produce un cerchio sul piano (=video), e produce inoltre altre 4 curve ottenute utilizzando lo stesso loop di tracciamento del cerchio, che utilizzano in pratica lo stesso angolo A, ma ne fanno, in un certo senso, un uso differente.

Il listato è in figura 1 e l'output in figura 2. L'angolo varia da 0 a 2π con uno step pari a $\pi/40$. Inoltre la window entro la quale lavoriamo va da -3, -2 a 3,2, e il raggio è 1. Questi valori saranno gli stessi per tutti gli altri programmi, proprio per limitare gli elementi di differenziazione tra un programma e l'altro.

Nel programma va notato quindi come a parità del loop, di raggio, di elementi di scaling, le differenti curve dipendono dalle differenti formule che contengono l'angolo A.

Va inoltre notato come la curva si chiude o meno a seconda che per 0 e 360 gradi le due funzioni abbiano lo stesso risultato, oppure (ed è il caso delle curve 2 e 3) nell'arco del 360 gradi le curve percorrano due volte lo stesso percorso.

Chi non sapesse nulla di formule trigonometriche potrebbe ricorrere, per tracciare una circonferenza, al teorema di Pitagora, la cui espressione è:

$$X^2 + Y^2 = R^2$$

Per disegnare la circonferenza occorre far «looppare» la X tra -R e R e, per ogni posizione calcolare, la Y con la stessa formula espressa nella forma:

$$Y = +/\sqrt{R^2 - X^2}$$

Poiché i valori della Y sono positivi e negativi occorre visualizzare sia la parte superiore della curva che la parte inferiore. In figura 3 vediamo il listato e in figura 4 l'output di un programma di tracciamento della circonferenza per mezzo del Teorema di Pitagora.

Vengono tracciati anche i raggi per far notare come il primo metodo, loopando sull'angolo A, produce segmentini di uguale lunghezza, al contrario del secondo, nel quale lo step della X può produrre segmenti corti o lunghi a seconda della sua posizione nei confronti del centro.

Una circonferenza nello spazio

Una circonferenza, così come tutte le figure bidimensionali, appartiene ad un piano e quindi se questo piano lo ipotizziamo disposto nello spazio rientriamo nella «tridimensionalità» dove entra in gioco la nuova variabile Z. Ciò significa che la nostra circonferenza

può essere espressa con coordinate «bidi» all'interno del piano al quale appartiene, ma se la consideriamo nello spazio deve necessariamente essere espressa in coordinate «tridi».

In altre parole la differenza tra «bidi» e «tridi» è insita non nell'oggetto trattato, ma nell'ambito nel quale viene trattato.

Dal punto di vista formale vogliamo definire una routine di traduzione di coordinate da tridimensionali a bidimensionali che utilizzeremo in tutti i programmi, che non discuteremo sia perché l'abbiamo fatto più volte in passato, sia perché, una volta stabilita

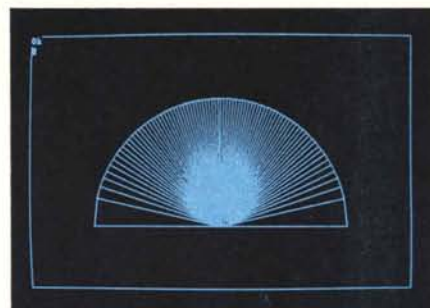


Figura 4 - Teorema di Pitagora - Output. Eseguendo il loop su una coordinata cartesiana e non sull'angolo, si ottiene una disomogeneità di lunghezza dei segmentini che costituiscono la circonferenza.

```
100 REM programma 1.1
110 SCREEN 2:CLS:WINDOW (-3,-1)-(3,3):LINE (-3,-1)-(3,3),,B
120 R=2:SR=R/40:RR=R^2
130 FOR X=-R TO R STEP SR:Y=SQR(RR-X^2)
140 IF X=-R THEN LINE (X,Y)-(0,0):X1=X:Y1=Y
150 LINE (X1,Y1)-(X,Y):LINE -(0,0):X1=X:Y1=Y
160 NEXT X:END
```

Figura 3 - Circonferenza realizzata con la Formula del Teorema di Pitagora. Listato. È possibile utilizzare metodi di tracciamento che non usano la trigonometria, ma le formule sono più complesse e i risultati sono peggiori.

```
100 REM inizializzazione
110 SCREEN 2:CLS:WINDOW (-3,-2)-(3,2):LINE (-3,-2)-(3,2),,B
120 PI=3.14:SP=PI/40:DA=SP*5:R=1:F=0
130 B=3:E=2:G=.3:S=SIN(G):C=COS(G):H=1.2:T=SIN(H):D=COS(H)
200 REM traduzione
210 XE=-X*S+Y*C:YE=-X*C*D-Y*S*D+Z*T:ZE=-X*T*C-Y*T*S-Z*D-B
220 M1=B*XE/ZE:N1=B*YE/ZE:RETURN
```

Figura 5 - Traduzione di Coordinate da Bidi a Tridimensionali. Useremo tali formule, che inseriremo, senza ridiscuterle, nei nostri listati.

e accettata una routine di traduzione va data per scontata.

La routine è isolata in figura 5 ed è composta da una parte di inizializzazione, nella quale vanno definiti i parametri che determinano la posizione dello «schermo» attraverso il quale si osserva l'oggetto (riga 100), e da una subroutine nella quale entrano X, Y, Z e escono M1, N1 (riga 200).

Il primo programma, listato in figura 6 e visualizzato in figura 7, è a semplice trasposizione della circonferenza nello spazio. Nella figura successiva, figura 8, vediamo i risultati che si ottengono facendo variare la Z da -1 a +1, con un passo .2.

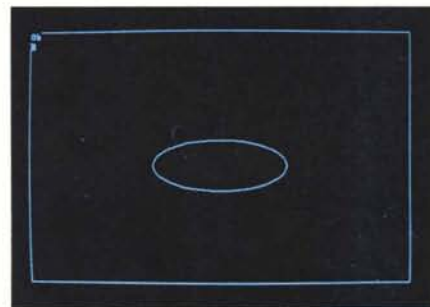


Figura 7 - Circonferenza nello Spazio - Output. In tale caso l'inclinazione non è data dalla posizione della circonferenza rispetto al piano XY, ma dai parametri che identificano la posizione e l'orientamento del punto di vista.

```
100 REM programma 2.0
110 SCREEN 2:CLS:WINDOW (-3,-2)-(3,2):LINE (-3,-2)-(3,2),,B
120 PI=3.14:SP=PI/40:DA=SP*5:R=1:F=0
130 B=3:E=2:G=.3:S=SIN(G):C=COS(G):H=1.2:T=SIN(H):D=COS(H)
140 FOR A=0 TO 2*PI STEP SP
150 X=R*COS(A):Y=R*SIN(A):Z=0:GOSUB 200
160 IF A=0 THEN PSET (M1,N1):ELSE LINE-(M1,N1)
170 NEXT A:END
200 XE=-X*S+Y*C:YE=-X*C*D-Y*S*D+Z*T:ZE=-X*T*C-Y*T*S-Z*D-B
210 M1=B*XE/ZE:N1=B*YE/ZE:RETURN
```

Figura 6 - Circonferenza nello Spazio - Listato. Va definita la nuova variabile Z, che nella grafica tridimensionale, indica l'altezza del punto rispetto al piano base, che è il piano XY.

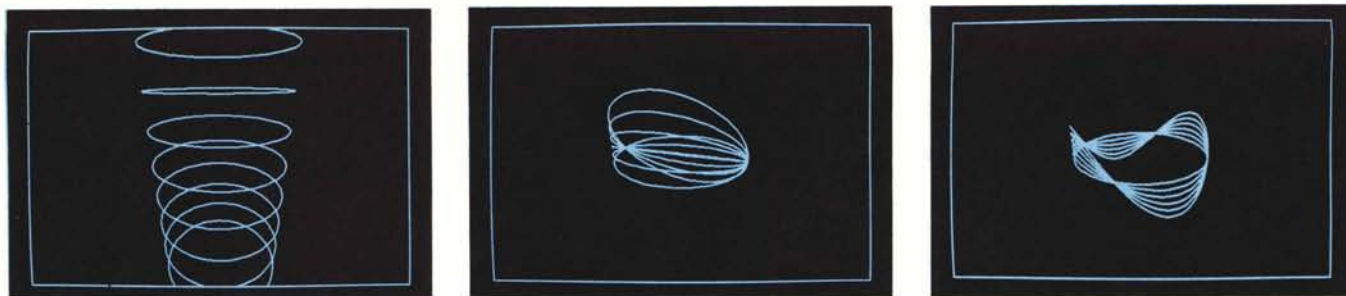


Figure 8, 9, 10 - Variazioni sul tema - Output. Le tre figure sono ottenute modificando solamente il valore Z, sul quale vengono eseguiti dei loop.

```

100 REM programma 2.1
110 SCREEN 2:PI=3.14:SP=PI/40:DA=SP*S:R=1:F=0
120 B=3:E=2:G=.3:S=SIN(G):C=COS(G):H=1.2:T=SIN(H):D=COS(H)
130 CLS:WINDOW (-3,-2)-(3,2):LINE (-3,-2)-(3,2),,B
140 FOR Z=-2 TO 2 STEP .5:FOR A=0 TO 2*PI STEP SP
150 X=R*COS(A):Y=R*SIN(A):GOSUB 400
160 IF A=0 THEN PSET (M1,N1):ELSE LINE-(M1,N1)
170 NEXT A:NEXT Z
230 CLS:WINDOW (-3,-2)-(3,2):LINE (-3,-2)-(3,2),,B
240 FOR K=0 TO 1 STEP .2:FOR A=0 TO 2*PI STEP SP
250 X=R*COS(A):Y=R*SIN(A):Z=X*K:GOSUB 400
260 IF A=0 THEN PSET (M1,N1):ELSE LINE-(M1,N1)
270 NEXT A:NEXT K
330 CLS:WINDOW (-3,-2)-(3,2):LINE (-3,-2)-(3,2),,B
340 FOR K=0 TO 1 STEP .2:FOR A=0 TO 2*PI STEP SP
350 X=R*COS(A):Y=R*SIN(A):Z=X*K:GOSUB 400
360 IF A=0 THEN PSET (M1,N1):ELSE LINE-(M1,N1)
370 NEXT A:NEXT K:END
400 XE=-X*S+Y*C:YE=-X*C*Y*S*D+Z*T:ZE=-X*T*C-Y*T*S-Z*D-B
410 M1=B*X*ZE/ZE:N1=B*Y*ZE/ZE:RETURN

```

Figura 11 - Variazioni sul tema - Listato. Vanno individuate routine di inizializzazione, subroutine di calcolo e di traduzione, e main routine, una per ogni disegno.

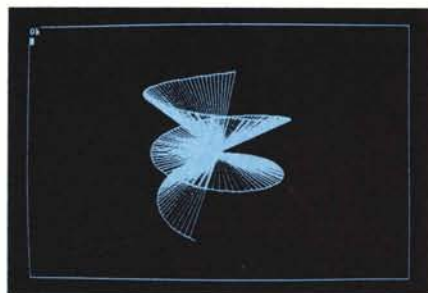


Figura 13 - Spirale nello Spazio - Output. La spirale è una delle curve più simile alla circonferenza. Nel nostro caso la trasportiamo nello spazio.

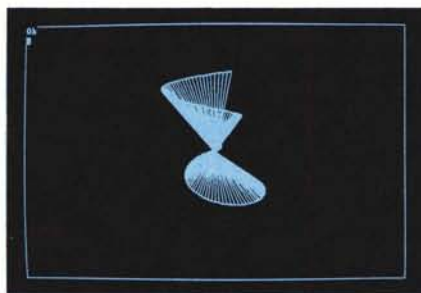


Figura 14 - Spirale nello Spazio - Output. In questo caso variano sia la coordinata Z che il raggio che ha un punto di minimo nel valore R pari a 0.

```

100 REM programma 2.2
110 SCREEN 2:CLS:WINDOW (-3,-2)-(3,2):LINE (-3,-2)-(3,2),,B
120 PI=3.14:SP=PI/40:DA=SP*S:R=1:F=0
130 B=3:E=2:G=.3:S=SIN(G):C=COS(G):H=1.2:T=SIN(H):D=COS(H)
140 X=0:Y=0:Z=0:GOSUB 200
150 Z=-1.2:FOR A=0 TO 5*PI STEP SP
160 X=R*COS(A):Y=R*SIN(A):Z=Z+.01:GOSUB 200
170 IF A=0 THEN PSET (M1,N1):M11=M1:N11=N1:GOTO 190
180 LINE (M11,N11)-(M1,N1):LINE (M11,N11)-(M0,N0):M11=M1:N11=N1
190 NEXT A:END
200 XE=-X*S+Y*C:YE=-X*C*Y*S*D+Z*T:ZE=-X*T*C-Y*T*S-Z*D-B
210 M1=B*X*ZE/ZE:N1=B*Y*ZE/ZE:RETURN

```

Figura 12 - Spirale nello Spazio - Listato. La Z viene incrementata in maniera autonoma rispetto alle X, Y, e all'angolo A.

Il passo successivo consiste nel legare la Z alle variabili X, Y. In questa maniera si ottiene una rotazione del piano della circonferenza, che appartiene sempre ad un piano (fig. 9). Se invece si lega la Z anch'essa alla A, si ottiene una figura tridimensionale che non «giace» su un piano (fig. 10).

Le tre figure citate sono ottenute in sequenza con un unico programma, listato in figura 11, nel quale vengono messe in subroutine le parti comuni.

Per esercizio potrete variare la posizione e l'orientamento del punto di vista, e non la posizione della circonferenza. In realtà l'effetto non è rilevabile dall'esame del disegno, in quanto uno stesso effetto può essere generato da cause differenti.

Ad esempio lo stesso effetto di ingrandimento di un oggetto lo otteniamo sia se noi ci avviciniamo all'oggetto sia se è l'oggetto che si avvicina. Questo principio vale nella grafica tridimensionale dove gli elementi in gioco nelle routine di calcolo sono decine, e quindi gli effetti possono essere ripetitivi, anche con parametri di calcolo differenti.

Le spirali

Una delle curve più strettamente imparentate con il cerchio è la spirale. Si ottiene infatti dalla stessa formula trigonometrica della circonferenza, ma si deve modificare un parametro.

Nel piano basta incrementare via via il Raggio in modo tale che la curva non si chiuda mai e quindi possa continuare all'infinito. Questa è la spirale di Archimede.

Nello spazio si può far variare, incrementandola con un passo fisso, la coordinata Z, in modo tale che il raggio della curva non varia, ma la stessa non si chiude perché quando si è completato un angolo di 360 gradi ci si è spostati su un altro piano.

In figura 12 vediamo il listato dell'esercizio relativo, e in figura 13 l'output.

Ulteriori varianti nel piano consistono nel far variare il Raggio con una legge differente (spirale logaritmica), oppure nello spazio far variare sia il raggio che la coordinata Z.

Per rendere più appariscente l'effetto abbiamo completato il disegno con una raggiera partente dal centro. (fig. 14).

Più curve nello spazio

Il successivo passo consiste nel tracciare più curve, nello spazio. Non ci riferiamo al caso in cui si disegnano in sequenza le varie curve ma al caso in cui vogliamo sfruttare un unico loop che serva per tutte le curve.

Il problema diventa quello di dover condividere le varie routine di plottaggio e quindi occorre utilizzare delle variabili in cui memorizzare coordinate dei punti delle varie curve.

Spieghiamo meglio. Per disegnare un segmentino di una curva si usa l'istruzione `LINE (X0, Y0)-(X1, Y1)` in cui $X1, Y1$ è l'ultimo punto calcolato e $X0, Y0$ il penultimo, ovvero quello dove è arrivata la curva. Per poter utilizzare una routine generica di loop occorre aver memorizzato in precedenti valori per poterli riutilizzare nel ciclo successivo quando servono di nuovo.

Nel listato in figura 15 si nota in riga 200 il lavoro del passaggio delle variabili. Ovvero per ogni curva si calcolano X, Y, Z che si passano alla routine di traduzione di coordinate (da riga 240 a 250) per ottenere le coordinate video M e N .

Queste vengono utilizzate nelle routine di plottaggio assieme ai precedenti valori memorizzati nelle variabili Mxx, Nxx . Il risultato è in figura 16.

Una ulteriore complicazione, ma sempre superabile, si presenta quando si vogliono tracciare anche i raggi di ciascuna circonferenza (vediamone solo il risultato in fig. 17).

Collegamento tra due curve

Per finire suggeriamo un altro esercizio, nel tridimensionale, nel quale vengono tracciate due curve nello spazio, calcolate con lo stesso loop. I pun-

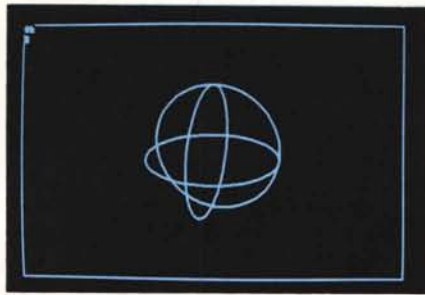


Figura 16 - Più curve nello Spazio - Output n. 1. Disegniamo tre circonferenze (tracciate contemporaneamente), ciascuna giacente su un piano nello spazio.

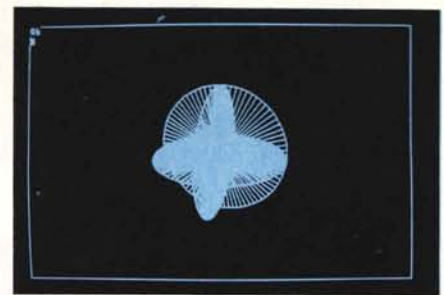


Figura 17 - Più curve nello Spazio - Output n. 2. Stesso discorso di prima, ma in questo caso con più il tracciamento dei raggi.

```
100 REM programma 3.0
110 SCREEN 2:CLS:WINDOW (-3,-2)-(3,2):LINE (-3,-2)-(3,2),,D
120 PI=3.14:SP=PI/40:DA=SP*S:R=1:F=0
130 B=3:E=2:G=.3:S=SIN(G):C=COS(G):H=1.2:T=SIN(H):D=COS(H)
140 FOR A=0 TO 2*PI STEP SP
150 X=R*COS(A):Y=R*SIN(A):Z=0:GOSUB 240:M1=M:N1=N
160 X=R*COS(A):Z=R*SIN(A):Y=0:GOSUB 240:M2=M:N2=N
170 Y=R*COS(A):Z=R*SIN(A):X=0:GOSUB 240:M3=M:N3=N
180 IF A=0 THEN PSET (M1,N1):PSET (M2,N2):PSET (M3,N3)
190 IF A=0 THEN M11=M1:N11=N1:M21=M2:N21=N2:M31=M3:N31=N3:GOTO 230
200 LINE (M11,N11)-(M1,N1):M11=M1:N11=N1
210 LINE (M21,N21)-(M2,N2):M21=M2:N21=N2
220 LINE (M31,N31)-(M3,N3):M31=M3:N31=N3
230 NEXT A:END
240 XE=-X*S+Y*C:YE=-X*C*D-Y*S*D+Z*T:ZE=-X*T*C-Y*T*S-Z*D-B
250 M=B*XE/ZE:N=B*YE/ZE:RETURN
```

Figura 15 - Più curve nello Spazio - Listato. Quando bisogna tracciare più curve che però dipendono da uno stesso loop il problema più grosso sta non nella formula della singola curva quanto nel passaggio di variabili necessario per plottare i vari pezzetti di curva senza «mischiare» gli elementi.

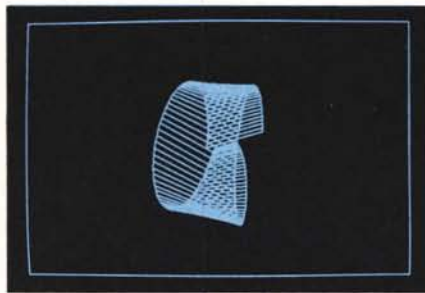


Figura 19 - Collegamento tra due Curve - Output. Una delle due curve viene «mandata avanti» di un angolo prestabilito.

ti corrispondenti, cioè quelli realizzati nello stesso ciclo di calcolo, vengono uniti tra di loro.

Nel caso più semplice si ottiene un «cilindro», ma variando qualche parametro si ottengono effetti ancora più gradevoli.

In figura 18 vediamo il listato dell'ultimo programma presentato e in figura 19 il suo output. L'effetto di più semplice realizzazione è quello raggiunto «sfalsando» le due curve. In pratica pur utilizzando lo stesso loop di calcolo sull'angolo A , si manda avanti una delle due curve di un certo angolo.

Suggerimento conclusivo

Esercizi di questo genere hanno l'unico scopo di far familiarizzare sia con la materia trattata, in questo caso la geometria nello spazio, che con il computer.

Quando, avendo un po' di tempo a disposizione e volete trascorrerlo con il vostro microcomputer provate a fare esercizi con spirito «sperimentale». Secondo me è molto più costruttivo del «game» ed è altrettanto appassionante se avete bisogno di appassionarvi e rilassante se avete bisogno di rilassarvi.

```
100 REM programma 5.0
110 SCREEN 2:CLS:WINDOW (-3,-2)-(3,2):LINE (-3,-2)-(3,2),,D
120 PI=3.14:SP=PI/40:DA=SP*S:R=1:F=0
130 B=3:E=2:G=.3:S=SIN(G):C=COS(G):H=1.2:T=SIN(H):D=COS(H)
140 FOR A=0 TO 2*PI STEP SP:B=A+DA
150 X=R*COS(A):Z=R*SIN(A):Y=-.5*R:GOSUB 200:M1=M:N1=N
160 X=R*COS(B):Z=R*SIN(B):Y=.5*R:GOSUB 200:M2=M:N2=N
170 IF F=1 THEN LINE (M1,N1)-(M11,N11):LINE (M2,N2)-(M21,N21)
180 F=1:LINE (M1,N1)-(M2,N2):M11=M1:N11=N1:M21=M2:N21=N2:NEXT A
190 END
200 XE=-X*S+Y*C:YE=-X*C*D-Y*S*D+Z*T:ZE=-X*T*C-Y*T*S-Z*D-B
210 M=B*XE/ZE:N=B*YE/ZE:RETURN
```

Figura 18 - Collegamento tra due Curve - Listato. Vengono collegati i punti delle due curve calcolati nello stesso ciclo del loop.

SCARTATE IL N°1.



Come si chiama il Personal Computer professionale portatile più venduto del mondo?

Non tirate ad indovinare, molto probabilmente dovrete scartare il primo nome che vi viene in mente.

Non stiamo parlando del primo in ordine di tempo o del più famoso in Italia. Non è lui il N° 1 che ci interessa. Avete un secondo nome? Ci dispiace, ma dovrete scartare anche questo. Con ogni probabilità si tratta soltanto e davvero di un N° 2.

Noi parliamo del N° 1.

Non viene dalla Silicon Valley e non è figlio del Sol Levante.

Il PC professionale portatile più piccolo e più leggero - il vero N° 1 - viene dal Texas.

Se questo vi fa venire in mente un nome, scartatelo.

Il N° 1 è quasi sconosciuto in Italia.

Ma lo sarà ancora per poco, perché se girate pagina saprete tutto.

Anche come si diventa N° 1.

AFFERRATE IL N°1.

IBM è un marchio registrato dalla International Business Machine Corporation.



Compaq. Questo è il nome del PC professionale portatile più venduto nel mondo. Il N° 1. Un grande successo che ha il motivo più semplice del mondo: i PC Compaq sono giudicati i migliori. Per due anni consecutivi e con due prodotti diversi infatti Compaq ha vinto il titolo di "Personal Portatile dell'Anno". E chi vince in questi casi è sempre il migliore. Il successo Compaq non ha precedenti nella storia economica degli USA. Un vero record che però non fa dormire il N° 1 sugli allori. La dimostrazione è il nuovissimo Portable II che vedete qui sopra.

Lo abbiamo confrontato con il portatile più venduto del mondo: il nostro Compaq Portable. Risultato: il Portable II è il 30% più leggero, il 17% più piccolo e, soprattutto, è da 3 a 5 volte più veloce. Persino di altri PC non portatili. Utilizza tutti i migliori Software e può dialogare con tutti i PC IBM e IBM compatibili. Anche con quelli che forse avete già. Così si diventa N° 1, lavorando sempre al meglio. E così siamo diventati anche il N° 2 del mondo per i Business PC. Ma vogliamo migliorare.

COMPAQ

Lavorare meglio è il nostro business.