



di Francesco Petroni

## Compendio di tecniche di grafica tridimensionale

*Più volte in questa rubrica è stato trattato l'argomento Computer Grafica Tridimensionale, sia per illustrarne le problematiche matematico-geometriche, sia, nell'ambito di argomenti più generali, come tecnica di rappresentazione di oggetti tridimensionali, e quindi come metodo di rappresentazione di superfici spaziali, di solidi geometrici, ecc.*

*Premesso quindi che la C.G. Tridimensionale non è che un metodo di rappresentazione, che è indispensabile conoscere per poterlo applicare correttamente alle varie problematiche, l'obiettivo che ci proponiamo con l'articolo è quello di raccogliere insieme tutte le tecniche, alcune delle quali già apparse in vari numeri della rivista (a vantaggio dei più interessati pubblichiamo un elenco degli articoli apparsi su MC di argomento Grafica Tridimensionale), e di inserirle tutte in uno stesso programma che lavora su uno stesso soggetto, permettendo così un confronto diretto tra le varie tecniche e in definitiva una migliore comprensione delle stesse.*

### Il sistema di riferimento e i gradi di libertà

Il primo concetto fondamentale che va capito è quello di sistema di riferimento, concetto che peraltro è indipendente dal successivo problema della visualizzazione.

Cercando di semplificare al massimo il discorso, a beneficio dei neofiti, occorre stabilire in quale punto dello spazio è situato l'oggetto e quindi stabilire prima quale è lo spazio di riferimento e poi come identificare la posizione dell'oggetto all'interno di questo.

Questo è un problema a sé stante presente in tutte le problematiche grafiche, che nelle applicazioni di Computer Grafica si tende giustamente a semplificare in quanto il metodo di rappresentazione deve tenere conto del fatto che lo strumento di output ha caratteristiche di uscita ben definite e allora si cerca di far coincidere lo spazio dove risiede l'oggetto con quello video.

L'altro problema è quello di identificare inequivocabilmente la posizione dell'oggetto nello spazio. In questo caso si dice che l'oggetto ha nello spazio 6 gradi di libertà, in quanto la sua collocazione è determinata dalla posizione di un suo punto (ad esempio il suo baricentro) e quindi occorrono le sue tre coordinate. Ma l'oggetto può assumere varie posizioni rispetto al suo baricentro tenuto fisso. Può ruotare in ciascuna delle tre direzioni fondamentali

(rispetto agli assi X,Y,Z) e quindi il totale dei movimenti possibili diventa sei. Non c'è modo di diminuire questo numero, che secondo la terminologia geometrica si chiamano gradi di libertà. Infatti se volessimo determinare la posizione di un oggetto nello spazio in un'altra maniera, ad esempio tramite due suoi punti, occorreranno sempre sei (tre più tre) coordinate.

num.	pag.	argomento
1	63	Grafica Tridimensionale - alcuni esempi
2	58	L'Italia vista dallo Space Lab
8	49	Teoria della Grafica Tridimensionale
9	50	Disegno di una Planivolumetria
10	56	Proiezioni Ortogonali - Assonometria
11	59	Introduzione alle Linee Nascoste
12	68	Linee Nascoste - due metodi
14	73	I Solidi Geometrici

Figura 1 - Elenco degli articoli inerenti alla Computer Grafica Tridimensionale apparsi su MC.

### Grafica Tridimensionale - Le tecniche

Dato un oggetto solido posizionato in un certo punto dello spazio esistono vari metodi per rappresentarlo su una superficie piana, quale un quadro o più semplicemente un foglio di carta o un monitor di un computer. Il problema ha una grande rilevanza anche in storia dell'arte dove l'utilizzazione delle tecniche di restituzione prospettiche, introdotte in epoca rinascimentale (Piero della Francesca), ha provocato addirittura il passaggio tra due epoche.

L'argomento è talmente interessante che oggi vengono condotti studi, guarda caso proprio con Computer Grafici, per ricostruire in forma reale, partendo dalle opere d'arte, gli ambienti rappresentati. Ne è un esempio la ricostruzione della sala in cui è ambientata l'Ultima Cena di Leonardo da Vinci, recentemente pubblicata su giornali specializzati, sia di arte sia di Computer Grafica.

Nel campo delle applicazioni di Computer Grafica prevalgono i sistemi di restituzione riferibili a precise regole geometriche

matematiche, quelle cioè più facilmente traducibili in un programma elaborativo, o per lo meno traducibili più facilmente di quanto non lo sia l'ispirazione artistica di un pittore.

Quindi il computer si presta facilmente ad applicazioni tecnico-scientifiche, quelle in cui il processo di visualizzazione, per quanto in maniera complessa, è sicuramente determinabile, senza lasciare margini alla casualità o all'estro.

### La scheda grafica Hercules per IBM

Il programma presentato, del quale parleremo tra poco, è stato sviluppato su un IBM, ma ovviamente gira su qualsiasi IBM compatibile dotato di capacità grafiche. Le foto a corredo sono state ottenute da un IBM dotato di scheda Hercules. Tale scheda usa come video grafico (in bianco e nero e con una definizione di 720 per 348 pixel) il monitor monocromatico.

Poiché tale scheda dispone anche di una uscita parallela per la stampante, sostituisce in tutto e per tutto la scheda monoprinter IBM, e quindi può rappresentare una alternativa "economica" per fare grafica (però in bianco e nero) ad alta definizione sull'IBM e inoltre la compatibilità software non rappresenta un problema in quanto tutti i package importanti la prevedono nei programmi di installazione.

Non vi sono problemi neanche per chi programma in Basic, anche se per lavorare con la Hercules occorre integrare il Basic con due o tre files da caricare dal dischetto soft fornito in dotazione.

Infatti le istruzioni Basic sono esattamente le stesse. In particolare SCREEN 2, se si lavora in Basic riconosce 640 per 200 pixel, se si lavora in HBasic (il Basic Hercules) riconosce 720 per 348 pixel. Se la funzione SCREEN 2 viene collegata ad una WINDOW (x1,y1) - (x2,y2) il programma diventa del tutto trasparente rispetto all'hardware in quanto l'uscita viene formattata in funzione dei parametri della WINDOW.

Così per utilizzare su tutto lo schermo Hercules un programma scritto per lo SCREEN 2 dell'IBM, ma che non utilizza la WINDOW, basta aggiungere, dopo SCREEN 2 un'istruzione WINDOW (0,0) - (720,348). In questa maniera è il Basic stesso che fa lo scaling tra i due formati. Il discorso vale anche per il viceversa.

Una misura del miglioramento delle prestazioni grafiche la dà il LOTUS 123, che sfrutta al massimo la definizione della Hercules, e questo non solo per la parte grafica

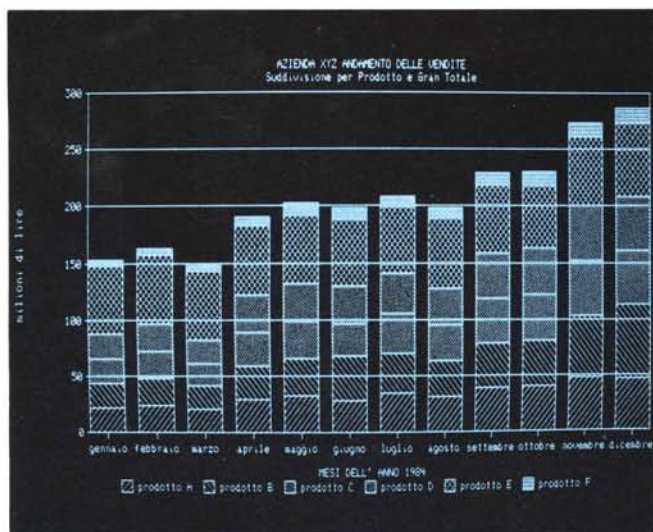


Figure 2/3 - Prestazioni grafiche della scheda Hercules in Lotus 123. La Hercules rappresenta una valida soluzione economica alle "carenze grafiche" del PC IBM base. Soprattutto perché è riconosciuta come standard da molti prodotti software di successo.

del disegno, ma anche e soprattutto per la parte "scritte" che non debbono più essere mostruosamente abbreviate per poter rientrare nella scarsa definizione dello SCREEN 1 dell'IBM. Nelle figure 2 e 3 mostriamo due grafici, realizzati con il binomio Lotus/Hercules, il primo di carattere trigonometrico e il secondo tipicamente business. Abbiamo dedicato all'argomento Grafica con il Lotus un articolo pubblicato su MC n. 41.

Il GWBASIC dell'Olivetti M24 invece accetta l'istruzione SCREEN 3, che permette una definizione di 640 per 400 pixel. Quindi per usare un programma scritto per lo Screen 2 IBM, migliorandone le prestazioni in termini di definizione, occorre modificare il 2 in 3 e, solo nel caso che non ci sia già, inserire un'istruzione WINDOW per una corretta formattazione dell'output.

Quindi si può dire che l'utilizzazione di programmi Basic realizzati con il PC IBM o IBM compatibili non presenta seri problemi nel caso si passi da un hardware ad un altro o da una configurazione ad una altra.

Nel caso invece che si utilizzi un pacchetto software prodotto per l'IBM esistono due possibilità, la prima è che questo preveda in fase di configurazione l'uso di un particolare hardware e allora non esistono problemi. Oppure se tale configurazione specifica non è prevista occorre sceglierne una comune all'IBM e alla macchina che si sta usando. È appunto il caso dell'Olivetti che pur avendo caratteristiche grafiche migliori rispetto all'IBM, non le può "mettere in mostra" fin quando l'hard Olivetti non sarà previsto nei programmi di configurazione.

Dopo aver divagato un po' sul tema della compatibilità tra le macchine, che è però particolarmente scabroso negli aspetti grafici, che sono quelli che sfruttano di più l'hardware della macchina, torniamo al nostro programma.

## Il programma

Il programma pubblicato esegue in sequenza una serie di visualizzazioni, basate sui metodi di restituzione tridimensionale, di uno stesso oggetto. Si compone quindi di una sequenza di routine, a ciascuna delle quali corrisponde un metodo geometrico.

Comprende poi una serie di subroutine generali, non legate cioè al singolo metodo, utilizzati in fase di predisposizione e di test delle subroutine.

I REM riguardano due oggetti. Il primo oggetto utilizzato per i nostri esperimenti di visualizzazione è un parallelepipedo. Il parallelepipedo è un soggetto semplice da interpretare ed è quindi adatto ad essere utilizzato in fase di predisposizione e di test delle subroutine.

Una volta assicuratici che tutti gli algoritmi matematico-trigonometrici funzionano a dovere possiamo lavorare su soggetti più complessi. Nel nostro caso il soggetto complesso è costituito da due prismi solidi molto irregolari, che si incastrano l'uno nell'altro.

Per passare da un soggetto all'altro voi dovete eliminare dal programma le righe dalla 100 alla 150 e dovete trasformare le righe REM dalla 170 alla 370 in righe DATA. Questo perché essendoci altri DATA nel programma, non è possibile lavorare contemporaneamente con i due oggetti.

I primi 6 DATA corrispondono a tre segmenti che rappresentano i tre assi cartesiani. Tali coordinate, quando manipoleremo il soggetto rimarranno fisse, in quanto simuleremo il movimento proprio rispetto agli assi cartesiani.

La forma dei DATA è molto semplice, il primo DATA indica il numero dei vertici costituenti il nostro oggetto. Per ogni vertice vanno indicate le tre coordinate cartesiane e un quarto dato che indica se quel vertice è collegato o meno al precedente (è un flag che vale 0 per collegamento SI, 1 per collegamento NO).

Questo è il metodo più semplice di memorizzazione di dati tridimensionali. In pratica dell'oggetto viene memorizzata una serie di linee spezzate che uniscono i suoi vertici. L'inconveniente è che un vertice può apparire più volte, sia perché è attraversato da più spezzate sia perché la spezzata ad un certo punto si chiude (ad esempio un parallelepipedo ha 8 vertici, non 18 che siamo costretti a memorizzare con il nostro sistema).

L'altro rallentamento all'esecuzione del programma grafico sta nel fatto che per ogni vertice va testata la condizione di inizio/fine spezzata. Che in termini di programmazione significa dover, per ogni punto da visualizzare, eseguire un IF... THEN...

Il formato con cui vengono memorizzati tali valori è il formato singola precisione. In realtà si potrebbero realizzare economie di spazio e miglioramenti di prestazioni lavorando con numeri interi. In questo programma non lo possiamo fare in quanto sottoporremmo i nostri dati a manipolazioni di tipo trigonometrico e questo significa aver assoluta necessità dei decimali.

I DATA vengono letti nella routine di Inizializzazione (riga 1000). In tale routine viene eseguito anche un altro... lavoretto. Vengono caricate due tabelle S(I) e C(I), di dimensioni pari a 361 elementi. Gli elementi n-simi della tabella contengono i valori rispettivamente del seno e del coseno di un angolo pari a n gradi.

In questa maniera ogni volta che serve il valore del seno di un angolo n, basta richiamare il valore S(n), ottenendo due vantaggi. Il primo è che si lavora in gradi sessagesimali come vogliamo noi, e non in radianti, come vuole il computer. Il secondo è che è più veloce l'accesso ad un vettore, che non il calcolo di una funzione trigonometrica.

Le altre routine generali sono poste in coda al programma e sono: la lettura della tastiera (riga 2000), che serve nella parte

interattiva del programma. La subroutine di visualizzazione (riga 3000) che è richiamata alla fine di ogni routine. La subroutine di passaggio di coordinate (riga 4000) che, partendo dalle coordinate iniziali (quelle lette via READ DATA), le traduce in altre coordinate tridimensionali, rappresentanti la posizione raggiunta dall'oggetto dopo la manipolazione.

La traduzione da coordinate tridimensionali (relative alla posizione finale) a coordinate bidimensionali (quelle da passare alla routine di visualizzazione), avviene a livello di routine specifica.

### Proiezioni sul piano

Sono le tre routine 1100, 1200, 1300, e costituiscono il sistema più semplice per visualizzare un oggetto tridimensionale. In pratica poiché l'oggetto tridimensionale è caratterizzato da tre coordinate, per passare alla vista bidimensionale, si elimina una

coordinata e si ottiene la proiezione sul piano costituito dalle altre due coordinate, ad esempio tramite l'eliminazione (o meglio la non considerazione) della coordinata Z, si ottiene la proiezione sul piano XY.

### Assonometrie

L'assonometria è un sistema di restituzione tridimensionale che si basa sulla conservazione delle dimensioni dell'oggetto. Quindi da una assonometria si può ricavare anche la misura dell'oggetto, ovviamente conoscendone la scala di rappresentazione.

In realtà il nostro modo di vedere è basato sulla prospettiva, infatti oggetti di pari dimensioni, ma che sono situati a distanze differenti dal nostro punto di osservazione, ci sembrano uno più piccolo dell'altro. Se vedessimo "assonometricamente" ci sembrerebbero comunque uguali (abbiamo trattato nel dettaglio la costruzione

delle assonometrie nel numero 10 di MC).

In pratica occorre definire inizialmente gli angoli formati, nel disegno risultante, da ciascun asse cartesiano rispetto agli altri due. Basta darne due e il terzo viene di conseguenza, ad esempio si può immaginare verticale sul video l'asse Z, orizzontale l'asse X, e l'asse Y obliquo rispetto ai primi due di 45 gradi. Un'altra assonometria standard è quella 120/120/120, in cui gli assi assumono l'aspetto di una raggiera di tre segmenti uscenti da uno stesso punto e formanti angoli uguali (ovviamente di  $360/3 = 120$  gradi).

### Prospettive

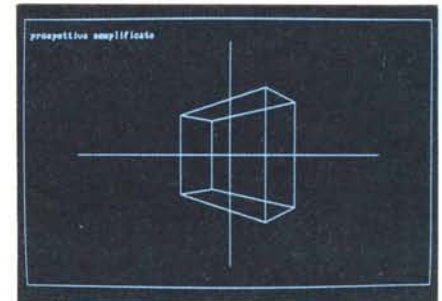
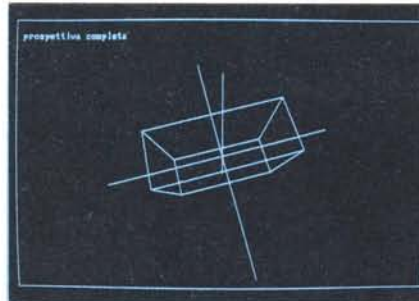
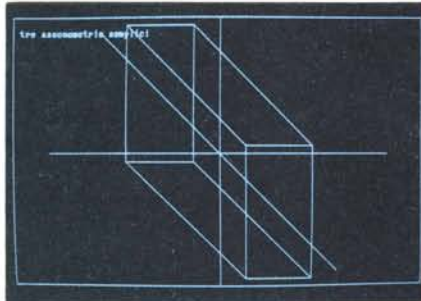
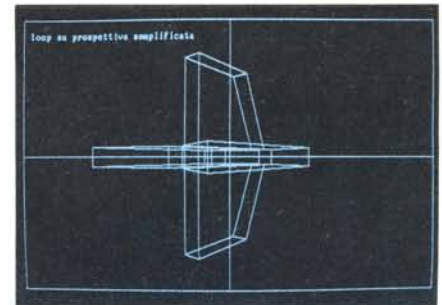
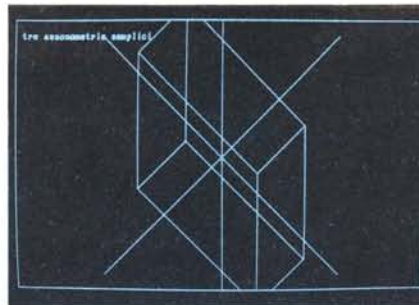
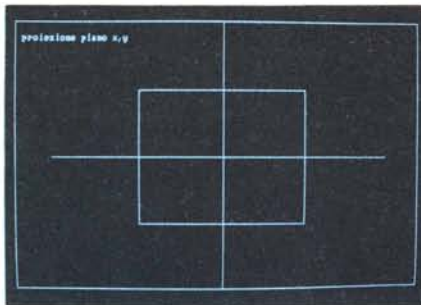
Il problema della prospettiva visto da un punto di vista geometrico matematico è abbastanza complesso in quanto entrato in gioco numerosi elementi. Poiché questi elementi sono interrelati è possibile che pur variando tutti quanti il risultato finale,

```

100 REM data parallelepipedo
110 DATA 24,-20,0,0,0,20,0,0,0,-20,0,0,0,0,20,0,0,0,0,-20,0,0,0,20,0
120 DATA 10,0,4,1,10,0,-4,0,10,-8,-4,0,10,-8,4,0,10,0,4,0
130 DATA -10,0,4,1,-10,0,-4,0,-10,-8,-4,0,-10,-8,4,0,-10,0,4,0
140 DATA 10,0,4,1,-10,0,4,0,10,0,-4,1,-10,0,-4,0
150 DATA 10,-0,-4,1,-10,-0,-4,0,10,-0,4,1,-10,-0,4,0
160 REM data incastro
170 REM 91,-20,0,0,0,20,0,0,0,-20,0,1,0,20,0,0,0,-20,1,0,0,20,0
180 REM -0,10,-1,1,2,10,-1,0,6,1,-1,0,-2,1,-1,0
190 REM -2,-1,-1,0,6,1,-1,0,2,-10,-1,0,-6,-10,-1,0,-6,10,-1,0
200 REM -6,10,1,0,2,10,1,0,6,1,1,0,-2,1,1,0
210 REM -2,-1,1,0,6,-1,1,0,2,-10,1,0,-6,-10,1,0,-6,10,1,0
220 REM 2,10,-1,1,2,10,1,0,6,1,-1,1,6,1,1,0
230 REM -2,1,-1,1,-2,1,1,0,-2,-1,-1,1,-2,-1,1,0
240 REM 6,-1,-1,1,0,-1,1,0,2,-10,-1,1,2,-10,1,0
250 REM -6,-10,-1,1,-6,-10,1,0,-6,10,-1,1,-6,10,1,0
260 REM -12,-1,0,1,4,-1,0,0,0,-1,1,0,-2,-1,1,0,-2,-1,-1,0,0,1,-1,0
270 REM 4,-1,-0,0,-12,-1,-0,0,-0,-1,-1,0,-2,-1,-1,0,-2,-1,1,0
280 REM -0,-1,1,0,-12,-1,0,0
290 REM -12,1,0,0,4,1,0,0,0,1,1,0,-2,1,1,0,-2,1,-1,0,0,1,-1,0
300 REM 4,1,-0,0,-12,1,-0,0,-0,1,-1,0,-2,1,-1,0,-2,1,1,0
310 REM -0,1,1,0,-12,1,0,0,-12,-1,0,0
320 REM 4,-1,0,1,4,1,0,0,0,-1,1,1,0,0,-2,-1,1,1,-2,1,1,0
330 REM -2,-1,-1,1,-2,-1,-1,0,0,-1,-1,1,0,1,-1,1
340 REM 4,-1,-0,1,4,1,-0,0,-12,-1,-1,0,1,-12,1,-0,0
350 REM -0,-1,-1,1,-0,1,-1,0,-2,-1,-1,1,-2,1,-1,0
360 REM -2,-1,1,1,-2,1,1,0,-0,-1,1,1,-0,1,1,0
370 REM 0,-1,-1,1,0,1,-1,0
390 REM
1000 REM inizializzazioni
1010 CLS:PRINT "ATTENDERE":PP=3.14159/180
1020 DIM X0(199),Y0(199),Z0(199),FX(199)
1030 DIM X1(199),Y1(199),Z1(199),XS(199),YS(199)
1040 READ N:FOR I=1 TO N:READ X0(I),Y0(I),Z0(I),FX(I):NEXT I
1050 DIM S(360),C(360):FOR I=0 TO 360:S(I)=SIN(I*PP):C(I)=COS(I*PP):NEXT I
1099 REM
1100 DATA "proiezione piano x,y"
1110 READ T$:FOR I=1 TO N:XS(I)=X0(I):YS(I)=Y0(I):NEXT I:GOSUB 3000
1199 REM
1200 DATA "proiezione piano y,z"
1210 READ T$:FOR I=1 TO N:XS(I)=Y0(I):YS(I)=Z0(I):NEXT I:GOSUB 3000
1299 REM
1300 DATA "proiezione piano x,z"
1310 READ T$:FOR I=1 TO N:XS(I)=X0(I):YS(I)=Z0(I):NEXT I:GOSUB 3000
1399 REM
1400 DATA "tre assonometrie semplici"
1410 READ T$:A1=135:A2=45:S1=C(A1):C1=C(A1):S2=S(A2):C2=C(A2)
1420 FOR I=1 TO N:XS(I)=-X0(I)*C1+Z0(I)*C2:YS(I)=X0(I)*S1+Z0(I)*S2+Y0(I)
1430 NEXT I:GOSUB 3000
1440 A1=135:A2=0:S1=C(A1):C1=C(A1):S2=S(A2):C2=C(A2)
1450 FOR I=1 TO N:XS(I)=-X0(I)*C1+Z0(I)*C2:YS(I)=X0(I)*S1+Z0(I)*S2+Y0(I)
1460 NEXT I:GOSUB 3000
1470 A1=150:A2=30:S1=C(A1):C1=C(A1):S2=S(A2):C2=C(A2)
1480 FOR I=1 TO N:XS(I)=-X0(I)*C1+Z0(I)*C2:YS(I)=X0(I)*S1+Z0(I)*S2+Y0(I)
1490 NEXT I:GOSUB 3000
1499 REM
1500 DATA "prospettiva completa"
1510 D1=20:D2=20:A=75:B=15:C1=C(A):S1=S(A):C2=C(B):S2=S(B):READ T$
1520 FOR I=1 TO N:XE=X0(I)*S1+Z0(I)*C1
1530 YE=X0(I)*C1+C2+Y0(I)*S2-Z0(I)*S1*C2
1540 ZE=X0(I)*C1*S2-Y0(I)*C2-Z0(I)*S1*S2+D1
1550 XS(I)=D1*XE/ZE:YS(I)=D1*YE/ZE:NEXT I:GOSUB 3000
1599 REM
1600 DATA "prospettiva semplificata"
1610 D1=20:D2=30:A=135:C=C(A):S=S(A):READ T$
1620 FOR I=1 TO N:XE=X0(I)*C+Z0(I)*S:YE=Y0(I):ZE=-X0(I)*S+Z0(I)*C
1630 XS(I)=D1*XE/(D2-ZE):YS(I)=-D1*YE/(D2-ZE):NEXT I:GOSUB 3000
1699 REM
1700 DATA "loop su prospettiva completa"
1710 READ T$:FOR K=0 TO 10:D1=25*K:D2=20-K:A=90+K*3:B=100-K*4
1720 C1=C(A):S1=S(A):C2=C(B):S2=S(B):FOR I=1 TO N
1730 XE=X0(I)*S1+Z0(I)*C1
1740 YE=X0(I)*C1+C2+Y0(I)*S2-Z0(I)*S1*C2
1750 ZE=X0(I)*C1*S2-Y0(I)*C2-Z0(I)*S1*S2+D1
1760 XS(I)=D1*XE/ZE:YS(I)=D1*YE/ZE:NEXT I:GOSUB 3000:NEXT K
1799 REM
1800 DATA "loop su prospettiva semplificata"
1810 READ T$:FOR K=0 TO 10:D1=20*K:D2=40-K:A=1+K*5:C=C(A):S=S(A)
1820 FOR I=1 TO N:XE=X0(I)*C+Z0(I)*S:YE=Y0(I):ZE=-X0(I)*S+Z0(I)*C
1830 XS(I)=D1*XE/(D2-ZE):YS(I)=-D1*YE/(D2-ZE):NEXT I:GOSUB 3000:NEXT K
1899 REM
1900 DATA "manipolazione figura"
1910 D1=20:D2=30:A=120:C=C(A):S=S(A):READ T$
1920 X0=0:Y0=0:Z0=0:A0=100:B0=100:C0=100:SC=5
1930 GOSUB 4000
1940 FOR I=1 TO N:XE=X0(I)*C+Z1(I)*S:YE=Y1(I):ZE=-X1(I)*S+Z1(I)*C
1950 XS(I)=D1*XE/(D2-ZE):YS(I)=-D1*YE/(D2-ZE):NEXT I:GOSUB 3000
1960 GOSUB 2000:GOTO 1930
1999 REM
2000 REM lettura tastiera
2010 JS=INKEY$:IF JS="" THEN 2010 ELSE IF VAL(JS)>0 THEN SC=VAL(JS):GOTO 2010
2020 IF LEN(JS)<2 THEN 2010 ELSE JX=ASC(MID$(JS,2,1))
2030 REM traslazioni
2040 IF JX=71 THEN X0=X0+SC:RETURN:REM pgdn
2050 IF JX=81 THEN X0=X0-SC:RETURN:REM home
2060 IF JX=80 THEN Y0=Y0+SC:RETURN:REM up
2070 IF JX=72 THEN Y0=Y0-SC:RETURN:REM down
2080 IF JX=77 THEN Z0=Z0+SC:RETURN:REM right
2090 IF JX=75 THEN Z0=Z0-SC:RETURN:REM left
2100 REM rotazioni
2110 IF JX=119 THEN A0=A0+SC:RETURN:REM cntr.pgdn
2120 IF JX=118 THEN A0=A0-SC:RETURN:REM cntr.home
2130 IF JX=132 THEN B0=B0+SC:RETURN:REM cntr.pgup
2140 IF JX=137 THEN B0=B0-SC:RETURN:REM cntr.end
2150 IF JX=116 THEN C0=C0+SC:RETURN:REM cntr.right
2160 IF JX=115 THEN C0=C0-SC:RETURN:REM cntr.left
2170 IF JX=82 THEN END:REM end
2180 RETURN
2199 REM
2200 REM visualizzazione
2210 SCREEN 2:CLS:WINDOW (-24,-16)-(24,16):LINE (-24,-16)-(24,16),B
2220 LOCATE 2,2:PRINT T$:PRINT CHR$(7)
2230 FOR J=1 TO N:IF FX(J)=1 THEN 3050
2240 LINE (XS(J),YS(J))- (XS(J+1),YS(J+1))
2250 NEXT J:RETURN
2299 REM
2300 REM passaggio di coordinate
2310 FOR I=1 TO 6:X1(I)=X0(I):Y1(I)=Y0(I):Z1(I)=Z0(I):NEXT I
2320 FOR I=7 TO N
2330 X1(I)=X0(I)*C(B0)+X0+Y0(I)*S(B0)+Z0(I)*S(C0)
2340 Y1(I)=Y0(I)*C(A0)+Y0+X0(I)*S(A0)+Z0(I)*S(C0)
2350 Z1(I)=Z0(I)*C(A0)+Z0+X0(I)*S(A0)+Y0(I)*S(B0)
2360 NEXT I:RETURN

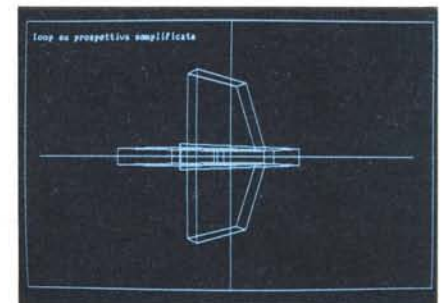
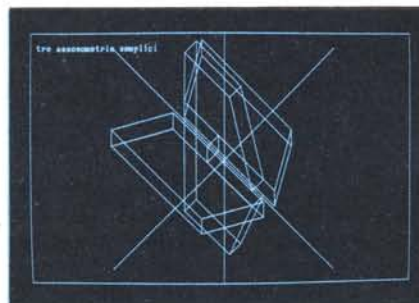
```

Figura 4 - Listato del programma. Il primo gruppo di dati riguarda un parallelepipedo, utile per la sua facilità di lettura in fase di test. Il secondo gruppo riguarda un disegno più complesso, nel nostro caso un incastro tra solidi prismatici.



### Le fotografie

*Pubblichiamo una serie di otto foto, raffiguranti i nostri due oggetti in varie posizioni. In realtà trattandosi di un programma che produce immagini in sequenza, dapprima secondo una logica predeterminata, poi interattivamente tramite tastiera, le figure sono puramente indicative.*



cioè il disegno dell'oggetto, sia lo stesso. Per fare un esempio cretino noi possiamo vedere la parte posteriore del nostro computer sia rivoltandolo, sia lasciandolo dove è ma spostandoci noi di dietro, sia infine utilizzando specchi dove riformare l'immagine.

Quindi il problema geometrico matematico ammette semplificazioni, che avvantaggiano la programmazione, in termini di semplicità e di velocità di esecuzione. Quindi possiamo prevedere ad esempio, ed è quello che faremo, di muovere l'oggetto dalla sua posizione iniziale, e di vederne lo spostamento senza muovere né l'osservatore né lo schermo attraverso il quale vede.

Utilizziamo due livelli di semplificazione, il primo prevede il punto di osservazione posto in un posto qualsiasi dello spazio, la seconda, più spinta, prevede il punto di osservazione, sempre posto sull'asse Z. Questo ha come conseguenza che, mentre con il primo metodo vedremo in generale tutti e tre gli assi, con il secondo vedremo sempre e solo quelli X e Y, mai Z.

Anche con la semplificazione più spinta potremo comunque vedere tutti gli aspetti dell'oggetto... rigirandolo un po' di più.

Le routine di prospettiva sono 5. Le prime due producono ciascuna una immagine statica di un oggetto, con i due metodi

completo (o meglio quello meno semplificato) e semplificato. Le seconde due eseguono dei LOOP in cui variano alcune grandezze che entrano nel calcolo della posizione finale. In tal modo si simula un movimento che, se il LOOP è fatto con criterio, si può cogliere nella successione delle immagini.

La quinta ed ultima routine è interattiva in quanto il movimento dell'oggetto rispetto alla sua posizione iniziale viene modificato intervenendo, via tastiera. Vediamo nel dettaglio solo quest'ultima in quanto le altre ne sono una semplificazione.

### Prospettiva interattiva

L'oggetto ha una sua posizione iniziale, corrispondente ai valori dei DATA. Si fanno variare, via tastiera, 6 parametri, che potendo assumere valori positivi e negativi diventano in un certo senso 12. Tali parametri corrispondono a spostamenti rispetto ai tre assi cartesiani (in avanti e all'indietro), e in rotazioni rispetto agli stessi tre assi (in un senso e in quello opposto).

Ad ognuno di questi 12 movimenti è collegato, dalla routine di riconoscimento tastiera, un tasto. Quindi premendo uno di questi tasti si provoca la modifica di un parametro numerico che indica di quanto

ci si è spostati rispetto alla posizione iniziale. Modificato un parametro viene eseguita la routine 4000 che ricalcola la nuova posizione finale, e poi la 2000 che visualizza l'oggetto nella nuova posizione.

Abbiamo utilizzato i vari tasti freccia, pgup, pgdn, ecc. nel tentativo di simulare il più possibile il senso dei movimenti. In realtà con la tastiera, questa sensazione è un po' difficile da raggiungere, specie per i movimenti di rotazione.

Per quanto riguarda questi ultimi hanno un limite nel senso che, essendo legati alla tabella dei SENI e COSENI, possono variare tra 0 e 360 gradi, altrimenti si esce fuori dalla tabella. Si poteva comunque inserire una formuletta per tradurre un angolo qualsiasi al corrispondente angolo compreso tra 0 e 360, ma non l'abbiamo fatto per non complicare troppo il programma. L'unica precauzione che abbiamo preso è stata quella di partire con angoli posti a 180 gradi (vedi riga 1920), per cui prima di uscire fuori dalla tabella... ce ne vuole.

La routine Tastiera riconosce anche i valori numerici per settare la variabile SC che stabilisce l'incremento di ciascuno spostamento. Valori bassi per piccoli spostamenti, valori alti per spostamenti più marcati.

MC

**OGGI C'E'**

# ISFO

## **AL COMPLETO SERVIZIO DEI RIVENDITORI**

**agente per Lazio, Umbria e Abruzzo:**



**Philips  
Computers** ◦ P3100 Personal Computer

**agente esclusivo per il Lazio:**

**telcom**

- stampanti ad aghi **MITSUI**
- floppy **MAXELL**
- stampanti low cost **CP/JP - 80**
- stampanti a margherita **JUKI**
- accoppiatori acustici **NOVATION CAT,**  
**ANDERSON - JACOBSON** ◦ plotter **YEW, ENTER C**
- digiter **GTCO** ◦ mouse **MOUSE SYSTEM**

disponibili LOTUS e SYMPHONY  
per IBM e compatibili  
e JAZZ per Macintosh

**agente esclusivo per Lazio e Umbria:**

**J.soft**

◦ software **J.soft** per Apple, IBM, Olivetti M24 e compatibili IBM



**GRUPPO  
EDITORIALE  
JACKSON**

◦ **tutti i libri della casa editrice**

**COVER**

◦ accessori per home e personal computer