Movimenti 3D

Nello spazio vengono introdotti comandi di movimento simili a quelli considerati per il piano.

Comandi che eseguono **traslazioni**:

ADDX dx

ADDY dy

ADDZ dz

ADD dx,dy,dz

In termini di coordinate equivalgono rispettivamente alle trasformazioni:

```
x2=x1 + dx,  y2=y1      ,  z2=z1
x2=x1      ,  y2=y1 + dy,  z2=z1
x2=x1      ,  y2=y1      ,  z2=z1 + dz
x2=x1 + dx,  y2=y1 + dy,  z2=z1 + dz
```

Comandi che eseguono **rotazioni** attorno ad un asse uscente dall'origine:

ROTX al   esegue una rotazione di al gradi attorno all'asse x

ROTY al   esegue una rotazione di al gradi attorno all'asse y

ROTZ al   esegue una rotazione di al gradi attorno all'asse z

ROTZ x,y,z,an   esegue una rotazione di al gradi attorno al vettore (x,y,z) con punto di applicazione in O

Tutte le rotazioni avvengono in senso antiorario, se osservare in direzione opposta al verso dell'asse.

I primi tre comandi possono essere espressi con trasformazioni analoghe a quella considerata per la rotazione del piano attorno all'origine, la quarta trasformazione risulta invece piuttosto complessa da esprimere in modo diretto. In termini di coordinate le prime tre espressioni equivalgono rispettivamente alle trasformazioni:

```
x2= x1,                        y2 = y1*cos(an) - z1*sin(an),   z2 = y1*sin(an)+ z1*cos(an)
x2 = z1*sin(an)+ x1*cos(an),   y2= y1,                         z2 = z1*cos(an) - x1*sin(an)
x2 = x1*cos(an)-y1*sin(an),    y2 = x1*sin(an)+ y1*cos(an),    z2=z1
```

Comandi che eseguono **allungamenti** nella direzione degli assi coordinati:

MULX mx   dilata di mx le dimensioni lungo l'asse x

MULY my   dilata di my le dimensioni lungo l'asse y

MULZ mz   dilata di mz le dimensioni lungo l'asse z

MUL mx,my,mz   allunga dei tre valori le dimensioni lungo gli assi x,y,z rispettivamente

Rotazioni 3D attorno ad assi verticali

Le formule dei movimenti sul piano possono essere adattate allo spazio, diventando rotazioni attorno ad assi paralleli a z. Il comando rotz be viene allora a corrispondere in 3D al comando rot2 be in 2D, mentre l'espressione add a,b,0 in 3D corrisponde a add2 a,b in 2D.

Alcune forme 3D predefinite, come il cilindro:

CYLIND h, r !cilindro con base sul piano xy, altezza h sull'origine e raggio r

la sfera :

SPHERE r !sfera di raggio r con centro sull'origine

e l'ellissoide :

ELLIPS h, r !semiellissoide di rotazione di raggio r, con base sul piano xy e altezza h

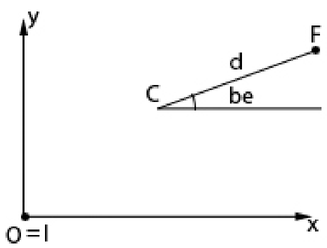
sono posizionate in modo da avere l'asse z come asse di simmetria.

Volendo fare in modo che una di tali forme venga a trovarsi ruotata attorno a un'asse verticale passante per il punto C (xc,yc,0) ad una distanza d dell'angolo be rispetto l'asse x, possiamo applicare la formula:

```
add  xc,yc,0
rotz be
add xc*cos(be)+  yc*sin(be) -xc, yc*cos(be)-xc*sin(be) -yc, 0
CYLIND h, r
del 3
```

Immaginando che il punto C si trovi in posizione (-d,0,0), il cilindro avrà rispetto ad esso la posizione (relativa) richiesta, poi dovrà ruotare di be, infine il tutto dovrà essere traslato di (d+ xc,yc,0), per porre il punto C nella posizione originaria, per cui la formula diventerà:

```
add  d+ xc,yc,0
rotz be
add d-d*cos(be), d*sin(be),0
CYLIND h, r
del 3
```



I = posizione iniziale dell'asse del cilindro

F = posizione finale

Applicando questa formula l'asse del cilindro viene a posizionarsi sul punto:
(xc+ d*cos(be),yc+ d*sin(be),0)

Per questo motivo, data la simmetria assiale del cilindro, sarebbe più semplice utilizzare il comando:
ADD xc+ d*cos(be),yc+ d*sin(be),0

Per forme che non hanno simmetria assiale, come i prismi, è invece necessario utilizzare la suddetta formula.

Il comando resol definisce il numero di facce delle forme apparentemente continue, come sfera e cilindro.

Con resol 4 o 5 il cilindro risulta un prisma a 4 o 5 facce: dopo la rotazione una faccia sarà rivolta all'asse per (xc,yc,0), cosa che non può avvenire utilizzando semplicemente il comando ADD.

Per lo stesso risultato esiste la formula equivalente, più semplice:

```
add  xc,yc,0
rotz be
addx d
CYLIND h, r
del 3
```

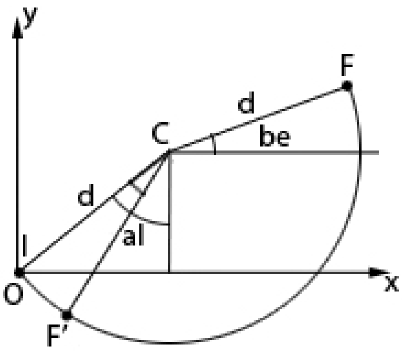
Infatti dopo la rotazione di be il vettore (-d*cos(be), d*sin(be),0) corrisponde al vettore iniziale (-d,0,0), per cui gli spostamenti corrispondenti si annullano a vicenda.

Consideriamo ora la rotazione dell'oggetto attorno all'asse per C, a partire dalla sua posizione iniziale I coincidente con l'origine.
Muovendosi da I l'oggetto ruota di un arco pari a be e va a finire in F'.

In questo caso la distanza d non è arbitraria, ma corrisponde al segmento OC. Inoltre l'angolo non è più riferito ad x, ma è riferito al vettore OC, che rispetto l'asse x è ruotato di -90-al, dove al rappresenta l'angolo rispetto y del vettore OC.
Risulta:
d*cos(al)=yc

Pertanto con questo tipo di rotazione la formula diventa:

```
d=sqr(xc^2+yc^2)
al=acs(yc/d)*sgn(xc)
add  xc,yc,0
rotz be-90-al
addx d
CYLIND h, r
del 3
```



ovvero:

```
add  xc,yc,0
rotz be-90- acs(yc/d)*sgn(xc)
addx d
CYLIND h, r
del 3
```

Questa formula può essere utilizzata per posizionare un oggetto dall'origine O ad una nuova posizione ruotata rispetto un asse obliquo AB. La nuova posizione dovrà immaginarsi appartenente ad un determinato piano ortogonale ad AB, che lo taglia in:

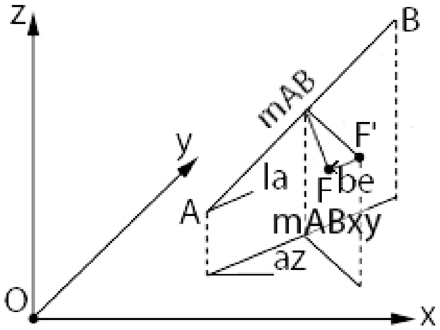
T=(xt,yt,zt)= (xa+(xb-xa)*t, ya+(yb-ya)*t, za+(zb-za)*t)

L'oggetto inizialmente si posiziona in F' lungo tale piano normale a distanza d da AB sopra l'asse per T parallelo a xy:

```
mAB=sqr((xb-xa)^2+(yb-ya)^2+(zb-za)^2)  !modulo di AB
mABxy=sqr((xb-xa)^2+(yb-ya)^2)            !proiezione su xy del modulo di AB
la=atn((zb-za)/mabxy)                      !latitudine di AB, vale 0 su xy: zb-
za=mAB*sin(la)=mABxy*tan(la)
az=acs((xb-xa)/mabxy)*sgn(yb-ya)          !azimut di AB:  mAB*cos(la) *cos(az)=mABxy*cos(az)=xb-xa
add xt+d*cos(az-90),yt+d*sin(az-90),zt
rot yb-ya,xa-xb,0, la-90
```

A questo punto l'oggetto, mantenendosi sul piano normale, ruota di be attorno ad AB, lungo un arco di circonferenza di raggio d fino a trovarsi in F:

```
xc=d*cos(90+ az) !posizione relativa dell'asse di rotazione
yc=d*sin(90+ az)
add  xc,yc,0     !rotazione di be attorno all'asse //z per xc,yc: nel nuovo sistema z=AB
rotz be-90- acs(yc/d)*sgn(xc)
addx d
CYLIND h, r
```



Formula:

```
xt=
yt=
zt=  !perno della rotazione
vx=
vy=
vz=  !direzione vettore asse AB
be= !angolo rotazione su orizzontale
v=sqr(vx^2+  vy^2+  vz^2)     !modulo di AB
vxy=sqr(vx^2+  vy^2)              !proiezione su xy del modulo di AB
az=acs(vx/vxy)*sgn(vy)        !azimut di AB
la=atn(vz/vxy)                        !latitudine di AB, vale 0 su xy
add xt+ d*cos(az-90),yt+ d*sin(az-90),zt
rot vy,-vx,0, la-90
xc=d*cos(90+ az) !posizione relativa dell'asse di rotazione
yc=d*sin(90+ az)
add  xc,yc,0          !rotazione di be attorno all'asse //z per xc,yc
rotz d-90- acs(yc/d)*sgn(xc)
addx d
CYLIND h, r
del  5
```