

Nel precedente articolo, con il pretesto di utilizzare il comando Extrude per la costruzione di un improbabile pilastro, abbiamo preso confidenza con la sintassi usata dal manuale per la descrizione dei comandi. Abbiamo così visto che ogni comando è costituito da un' identificativo (o parola chiave) seguito da un certo numero di parametri.

A prescindere dal significato degli altri parametri, facilmente comprensibili e in molti casi analoghi e ricorrenti nella sintassi dei vari comandi, mi preme questa volta attirare l'attenzione sui parametri denominati "MASK" (maschera) e "S" (Status). Questi sono infatti una sorta di denominatore comune dei comandi citati nell'articolo precedente; una selezione tra quelli che utilizzano una o più polilinee per la generazione della forma 3D.

La causa di gran parte delle difficoltà sull'uso avanzato del GDL sta, a mio parere, proprio nel comprendere e riuscire a padroneggiare questi due tipi di parametri.

Se andiamo a rileggerci il manuale di riferimento del GDL (da pagina 44 a pagina 46) faremo luce su di un paio di cosette estremamente interessanti, e, direi fondamentali, per un proficuo uso di questo linguaggio.

Scopriamo infatti che le funzioni GDL che si servono di polilinee per generare una forma tridimensionale, ricorrono a due parametri chiamati MASK e STATUS. Questi parametri permettono di definire la presenza (e quindi visibilità o meno) di superfici e spigoli del solido 3D per quanto riguarda Mask, o la visibilità degli spigoli generati dai punti in movimenti della polilinea generatrice nel caso di Status.

MASK e Status...

due perfetti sconosciuti

Avelino De Sabbata

MASK

I valori di Mask sono definiti in un modo che potrebbe sembrare bizzarro a chi non ha dimestichezza con la numerazione binaria. Questi valori rappresentano infatti delle vere e proprie maschere binarie (da cui il nome) e la faccenda, a spanne, funziona così: Leggiamo a pagina 45 che il parametro Mask in genere può assumere i valori che derivano dalla seguente formulazione:

$$\text{mask} = J1 + 2 \cdot J2 + 4 \cdot J3 + 8 \cdot J4 + 16 \cdot J5 + 32 \cdot J6 + 64 \cdot J7$$

dove J1 ... J7 possono assumere il valore di uno o zero.

Potrebbe sembrare Aramaico antico, ma è invece molto semplice. Potrete facilmente constatare che il valore mask può assumere qualsiasi valore compreso tra zero (J1 ... J7 tutti zero) e 127 (J1 ... J7 tutti uno). Ora considerate i vari addendi della formula più

sopra riportata come fossero altrettanti interruttori che possono assumere la posizione di ON (acceso, ossia uno) oppure OFF (spento, ossia zero). Potremmo dire quindi che il parametro Mask non è altro che un contenitore di interruttori, ovvero un parametro che rispetto al valore assunto permette di controllare sette situazioni con l'unica restrizione che queste "situazioni" possono assumere uno tra due stati ben definiti.

Se rappresentiamo il numero risultante dalla formula citata in notazione binaria ci rendiamo immediatamente conto di come esso, con i singoli BIT non faccia altro che simulare questa serie di interruttori. L'assegnazione di zero o uno al corrispondente Jn ha in effetti la funzione di spegnere o accendere il relativo interruttore.

Per comprendere meglio osservate la seguente tabella, che riporta alcuni dei possibili valori di mask espressi anche in formato binario

Mask	J1	J2	J3	J4	J5	J6	J7
0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0
2	0	1	0	0	0	0	0
3	1	1	0	0	0	0	0
4	0	0	1	0	0	0	0
8	0	0	0	1	0	0	0
16	0	0	0	0	1	0	0
32	0	0	0	0	0	1	0
48	0	0	0	0	1	1	0
63	1	1	1	1	1	1	0
64	0	0	0	0	0	0	1
65	1	0	0	0	0	0	1
66	0	1	0	0	0	0	1
127	1	1	1	1	1	1	1

Chiariamo ulteriormente il concetto, riprendendo l'esempio del pilastro proposto nel precedente articolo. In quel caso la prima linea dell'istruzione Extrude era:

Extrude 53, 0, 0, 3, 16+32

In questo caso il valore di Mask è di 48, che, per maggiore chiarezza abbiamo scritto nella forma 16+32, ovvero interruttori J5 e J6 impostati a ON, e tutti gli altri impostati a zero ovvero OFF. Leggiamo infatti a pagina 47 che (per quanto riguarda Extrude) J5 e J7 definiscono rispettivamente la visibilità degli spigoli di base e superiori del solido generato. Se visualizzate il 3D di tale oggetto in modo opportuno potrete rendervi conto che effettivamente la superficie di base e quella superiore sono state omesse (interruttori J1 e J2 = 0). Volendo visualizzarle è sufficiente modificare il comando nel modo che segue:

Extrude 53, 0, 0, 3, 1+2+16+32

Nel caso dell'esempio, l'impostazione o meno dell'interruttore J3 è influente. Tale valore definisce infatti la presenza o meno della superficie laterale di chiusura, ma ovviamente solo in presenza di una polilinea generatrice aperta, dove cioè le coordinate del punto finale non sono uguali con quelle del punto iniziale.

STATUS

Da notare che quanto diremo in segui-

to riferendoci al parametro Status (riferito ai comandi EXTRUDE, PIRAMID, REVOLVE, SWEEP e TUBE), ad eccezione degli status aggiuntivi, vale anche per il parametro MASK in relazione ai comandi POLY_, PRISM_, cPRISM_, bPRISM_, POLY2_, e cROOF_. Quest'ultimo comando è stato introdotto con la versione 5 di Radar/Ch, ora ArchiCAD.

Per quanto riguarda il parametro Status, la faccenda, a prima vista sembrerebbe molto più semplice.

Questo parametro normalmente può assumere due soli valori: zero o uno ad eccezione dell'istruzione Extrude dove viene accettato anche il valore negativo -1.

Quando ad un punto della polilinea generatrice assegniamo il valore zero al parametro status significa che vogliamo che la traccia generata da quel punto (in relazione al movimento effettuato dalla polilinea generatrice) sia visibile, sia essa il punto di uno spigolo o di un arco, in relazione al tipo di comando (Extrude o Revolve ad esempio). Potremo invece rendere invisibile tale traccia assegnando il valore uno allo status di quel punto. A dire il vero lo status 1 delega il controllo della visibilità dello spigolo alle routine interne del software.

Il valore negativo nel caso dell'istruzione Extrude (corrispondente al valore negativo del parametro MASK nelle istruzioni più sopra elencate) ci permette di inserire dei fori nella polilinea generatrice. In questo caso, al punto finale della descrizione del perimetro della polilinea del contorno inseriamo il valore Status = -1 (le coordinate X, Y di tale punto debbono coincidere con il punto iniziale del contorno — polilinea chiusa) e di seguito iniziamo la descrizione del

perimetro del foro, che termineremo ancora con coordinate uguali a quelle di inizio del foro, ma con Status, invece, negativo.

STATUS ADDIZIONALI

Dicevo che la comprensione del parametro Status "sembrerebbe" più semplice; infatti quanto esposto era tutto fino all'introduzione dei cosiddetti STATUS ADDIZIONALI nati con la versione 4.5 di Radar/Ch.

Da questa revisione, Radar/Ch mette infatti a disposizione una maggiore varietà di modi per la descrizione delle polilinee piane nei testi GDL. La lettura del manuale da pagina 66 dovrebbe chiarire l'arcano. Ancora una volta il condizionale! In questo caso la difficoltà a mio parere è determinata dall'esagerata sinteticità dell'esposizione.

Come abbiamo visto nel precedente articolo, prima dell'introduzione degli status aggiuntivi, l'accuratezza nella creazione di una forma con linee curve era determinata dal numero di segmenti utilizzati per approssimare tali curve. Nello stesso articolo sono stati sottolineati gli svantaggi di tale modo di procedere.

Chi ha avuto la pazienza di copiare il breve listato GDL riportato alla fine dell'articolo si sarà reso conto dei notevoli vantaggi che derivano dall'uso degli status aggiuntivi. Questi ci permettono infatti di descrivere polilinee complesse con la definizione di pochi nodi (un nodo viene definito dalle coordinate X, Y e dallo Status).

Il vantaggio maggiore deriva dal fatto di poter descrivere archi di cerchio o intere circonferenze con uno o due soli nodi. Da notare inoltre che sebbe-

ne gli archi e le circonferenze creati con gli status addizionali siano veri archi e vere circonferenze, le superfici create con tali polilinee generatrici sono comunque influenzate dal settaggio RESOL. In condizioni normali produrranno comunque un ottimo rendering.

Per rendervi conto di questo provate a introdurre la direttiva RESOL 3, RESOL 4 o valori superiori come prima istruzione nel listato citato. Sostituite inoltre lo status 1001 nella terz'ultima linea e verificate i risultati della visualizzazione 3D.

RESOL 4

```
FOR J=0 TO 3
  ROTZ 90*J
  Extrude 4, 0, 0, 3, 16+32
  -0.20, 0.05, 600,
  0.1, 270, 200,
  -1, 0, 800,
  -0.05, -0.20, 1000
  DEL 1
NEXT J
```

figura 1

Il principio generale che governa gli status addizionali è il seguente: Ai valori di Mask/Status utilizzati normalmente vengono addizionati i valori dello status addizionale.

Quasi tutti gli status addizionali partono dal presupposto che sia già definita la posizione di partenza (nodo corrente, ovvero ultimo nodo definito) ed una direzione corrente (tangente, ovvero la prosecuzione ideale della polilinea corrente).

Nel caso in cui il nodo da definire sia il primo nodo della polilinea è necessario definire la posizione di partenza

e la direzione corrente.

Questo è possibile grazie gli status 600 e 800. In pratica le coordinate X, Y abbinate allo status 600 definiscono il punto di origine e le coordinate seguenti, con status 800, definiscono la direzione (tangente) tramite il vettore la cui origine è data dalle coordinate precedenti (status 600).

Detto questo, dovremmo essere in grado di analizzare il breve listato dell'oramai celebre pilastro, riportato in figura 1:

Come prima cosa possiamo osservare che l'istruzione Extrude vera e propria la troviamo all'interno di un ciclo For...Next. Questo è semplicemente un espediente per limitare la lunghezza del programma. Infatti, visto che è possibile scomporre il perimetro in quattro parti perfettamente uguali noi faremo altrettanto.

Ricordate la regoletta che segue, sempre valida ma essenziale nella programmazione: se possibile cerca di scomporre il problema principale in tanti sottoproblemi. Per la risoluzione di questi applicate ancora lo stesso approccio.

Alla fine, la soluzione di tanti piccoli problemi risulterà senza dubbio più agevole che l'affrontare di petto tutto il lavoro in un colpo solo.

La funzione di ripetizione è determinata dalle istruzioni For j=0 to 3 e alla fine Next J. La traduzione di queste istruzioni in linguaggio corrente potrebbe essere quella che segue:

- Assegna il valore zero alla variabile J
- Istruzioni
- Incrementa di uno il valore di J e se non è maggiore di 3 ripeti le istruzioni precedenti

Ma ripetere per quattro volte le stes-

se istruzioni nel nostro caso non basta, è anche necessario eseguire ogni volta una rotazione di 90° sull'asse zeta. Noi otteniamo lo stesso risultato ruotando l'istruzione Extrude ogni volta di 90° moltiplicati per il valore della variabile J e cancellando l'effetto della rotazione alla fine del ciclo. Così la prima volta non effettuiamo alcuna rotazione (infatti RotZ 90*0 non cambia alcunchè), quindi ruotiamo di 90°, poi di 180° ed infine di 270°.

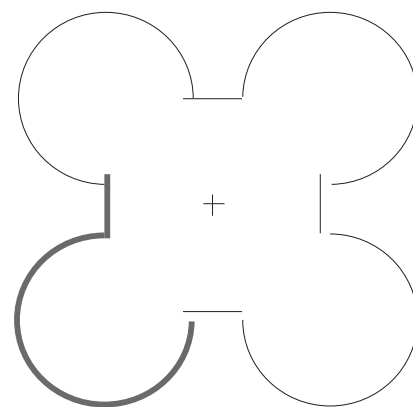


figura 2

E veniamo quindi ai fatidici status addizionali dell'istruzione Extrude. Con il primo nodo (status 600) impostiamo il punto iniziale della nostra polilinea nelle coordinate X= -0.20 e Y= 0.05. Tracciamo quindi un segmento che grazie allo status 200 viene determinato da lunghezza (0.1) e angolo (270°). È bene ricordare che la misurazione degli angoli nei sistemi CAD avviene sempre in senso antiorario con lo zero sull'asse X positivo.) A questo punto, grazie al nodo con status 800 impostiamo la direzione (tangente) per la prossima istruzione.

La direzione è determinata da un vettore che ha inizio nel punto di arrivo del segmento testè definito ed il punto finale è dato dalle coordinate relative $X=-1$ e $Y=0$. La direzione definita è cioè verso sinistra, ossia a 180° .

L'istruzione che segue (status 1000+1) traccia il settore di arco e conclude la polilinea. Questo modo di definire l'arco è forse il più bizzarro, e non è stato scelto a caso per il nostro studio. Normalmente, dovendo disegnare un cerchio o un arco pensiamo subito al centro ed al raggio. In questo caso invece usiamo le coordinate dei punti iniziale e finale, unitamente alla tangente, ovvero la direzione dell'arco nel punto di partenza che avevamo definito con il nodo precedente.

Ecco che la semplificazione estrema del listato ci permette di parametrizzare facilmente il nostro oggetto. Potremo cioè ottenere con comodità oggetti con la stessa geometria, ma

con dimensioni differenti. A questo punto non ci soffermeremo sulle singole istruzioni. Ci basti sapere che è sufficiente inserire nell'elenco dei parametri dell'oggetto le due righe come si vede nella figura 3, e sostituire quindi nel listato 3D i valori delle coordinate dei vari nodi con altrettante variabili opportunamente definite. Nella figura 4 è riportato il listato così modificato.

Ai più volenterosi il compito di aggiungere un parametro e rendere indipendenti misure X e Y del prisma centrale del pilastro.

```
IF D<=E*2 THEN 1000
X=D/2 : L=(D-(E*2))/2

FOR J=0 TO 3
  ROTZ 90*J
  Extrude 4, 0, 0, 3, 16+32,
    -X,    L,    600,
    L*2,   270,   200,
    -1,    0,    800,
    -L,    -X,   1001
  DEL 1
NEXT J

1000: END
```

Figura 4

Nella figura che segue (figura 5) troviamo infine il codice per il simbolo 2D parametrico che come vedete è assolutamente simile al precedente, ad eccezione degli status (normali, non addizionali) che nei comandi Extrude e Poly2 sono differenti.

```
IF D<=E*2 THEN 1000

X=D/2 : L=(D-(E*2))/2

FOR J=0 TO 3
  ROT2 90*J
  Poly2_ 4, 1,
    -X,    L,    601,
    L*2,   270,   201,
    -1,    0,    800,
    -L,    -X,   1001
  DEL 1
NEXT J

1000: END
```

Figura 5

Alla fine dell'articolo precedente suggerivo di modificare il valore negativo nella riga $-1, 0, 800$ con un valore positivo.

Chissà se c'è qualcuno che si è preso la briga di verificare quello che succede. Per i più pigri ecco qui di seguito il risultato.

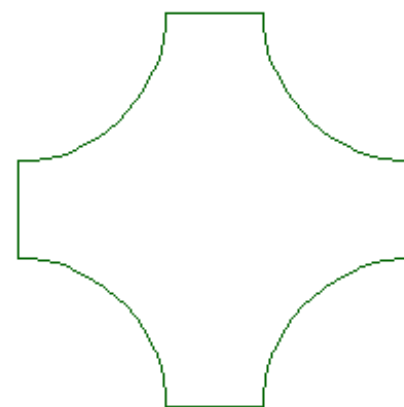


Figura 6

