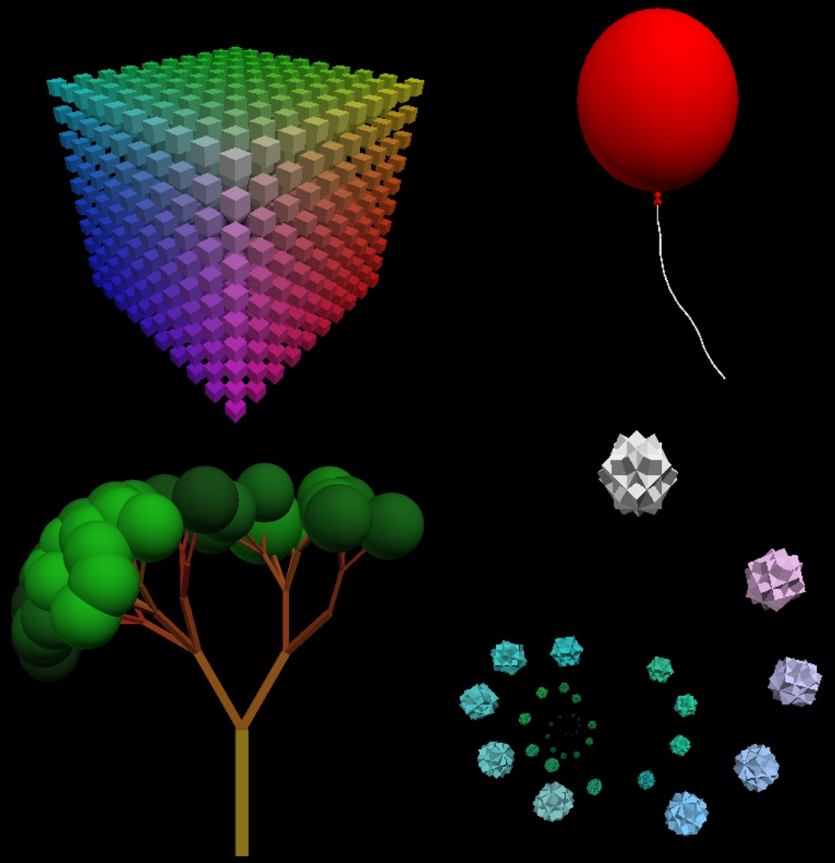
Teoría de Lenguajes

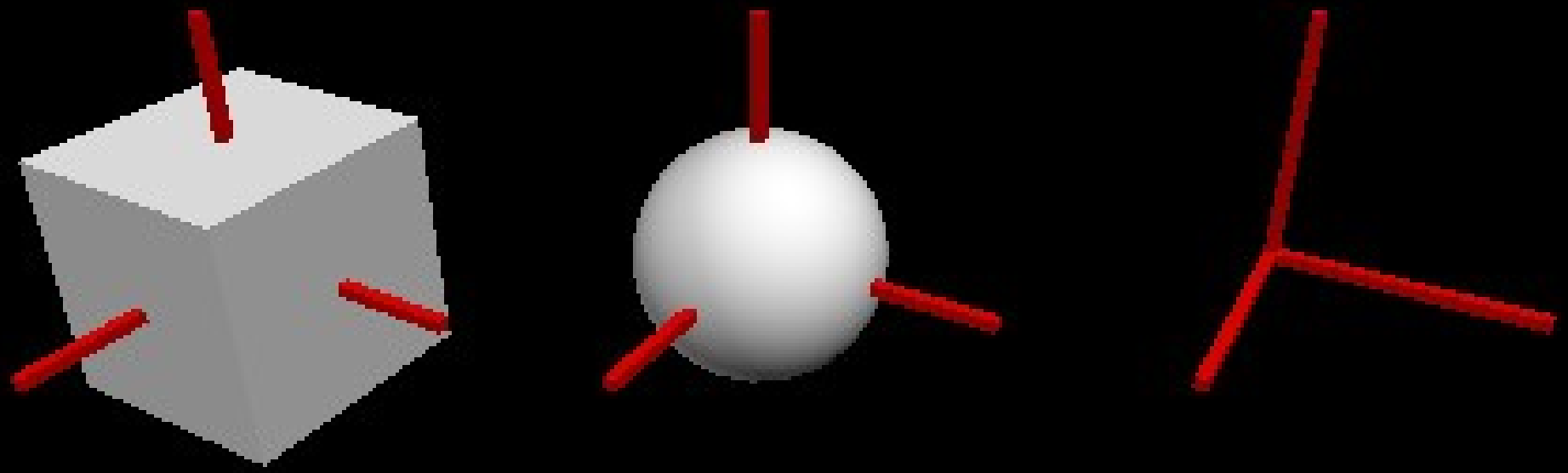
Procedural
Elements of
Grammatical
Synthesis

PEGS!



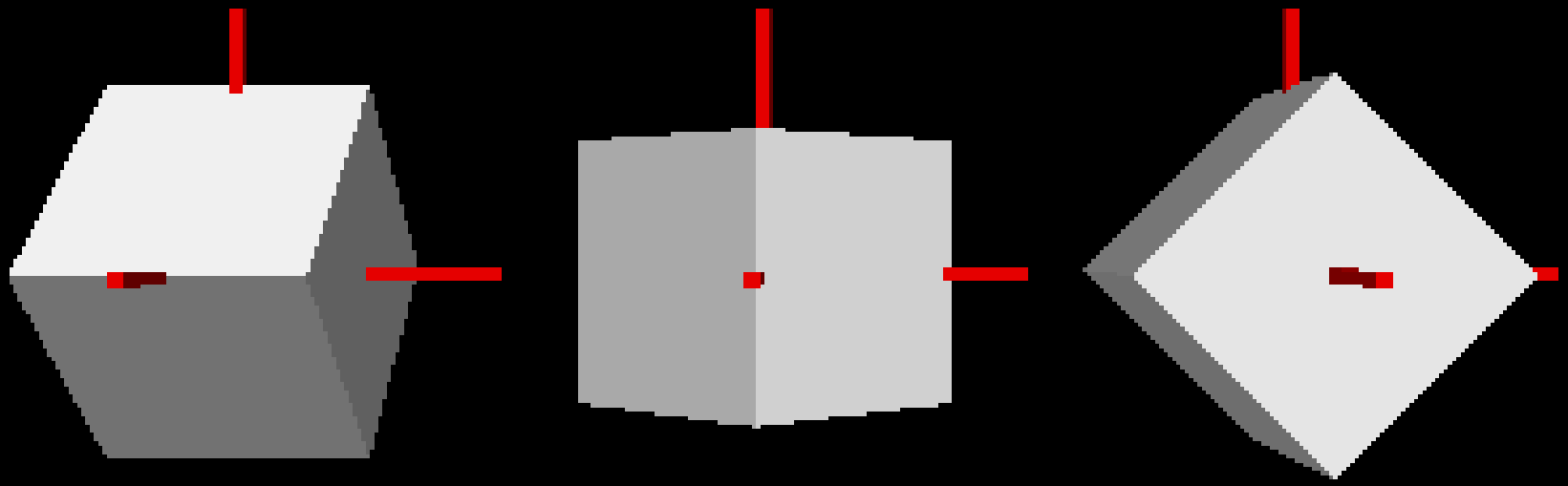
Primitivas

box ball _



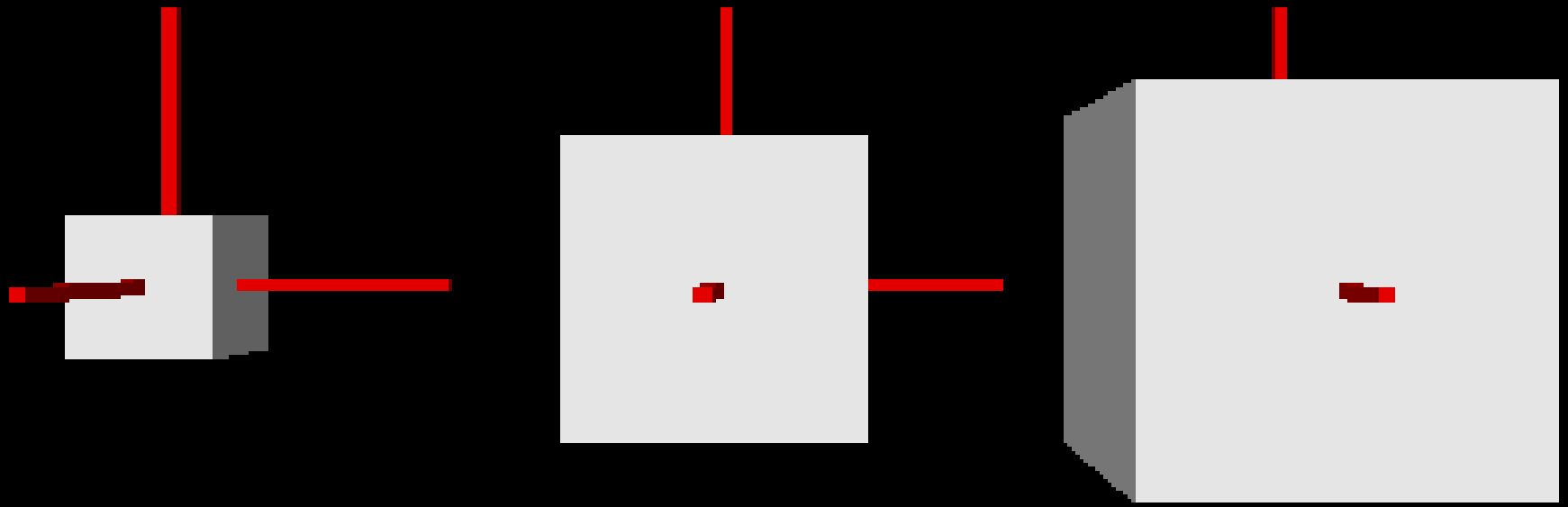
Transformaciones

box:rx 45 box:ry 45 box:rz 45



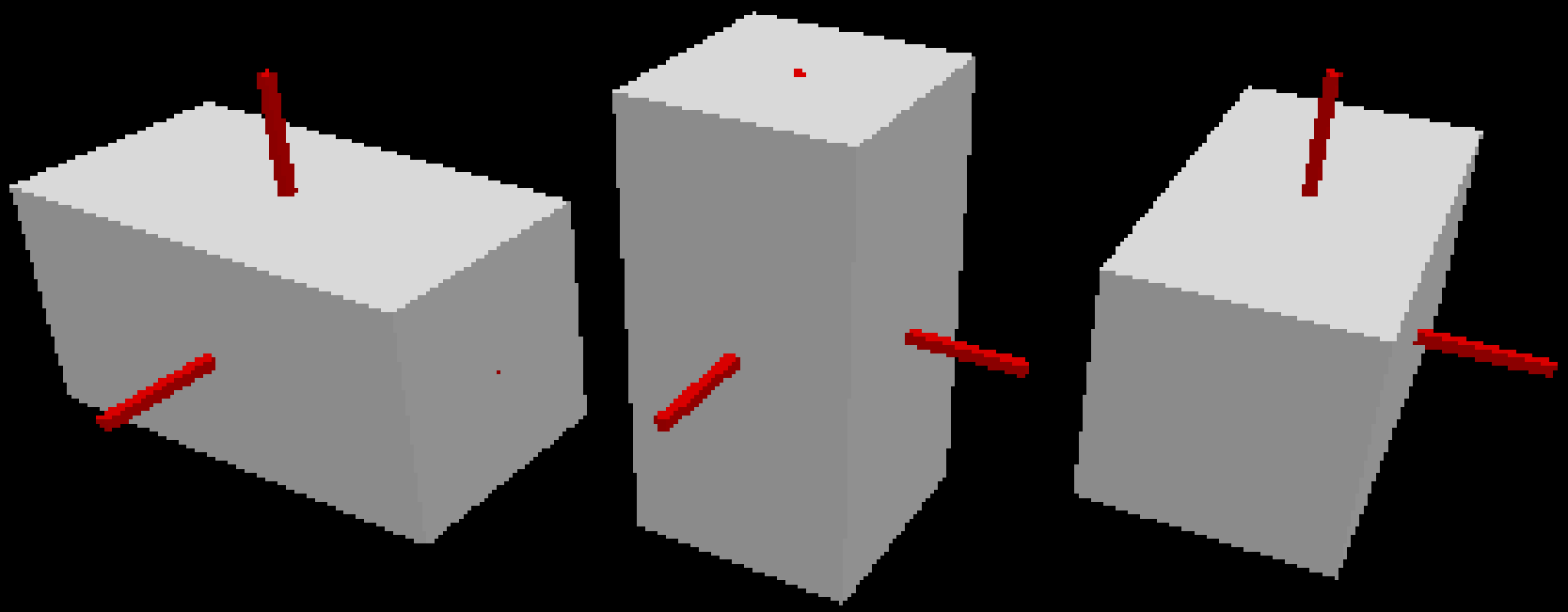
Transformaciones

box:s 1/2 box:s 1 box:s 4/3



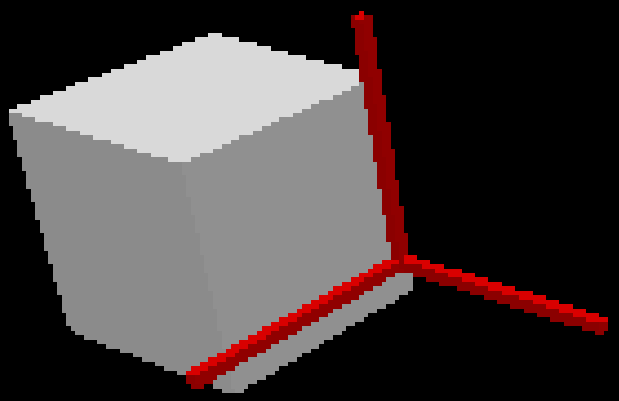
Transformaciones

box:sx 2 box:sy 2 box:sz 2

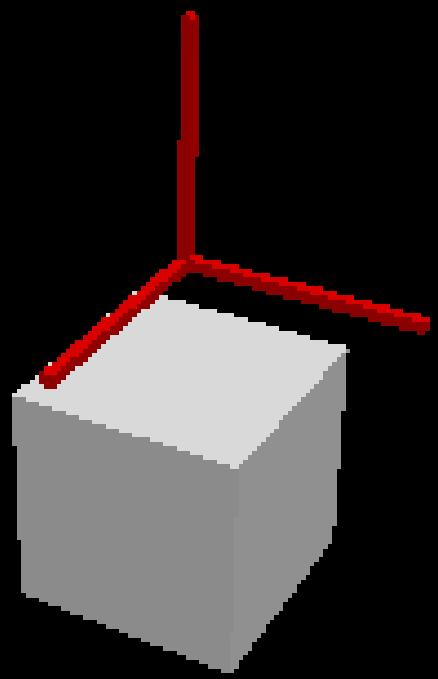


Transformaciones

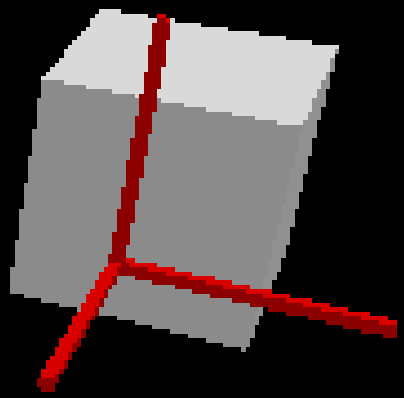
box:tx -1



box:ty -1



box:tz -1

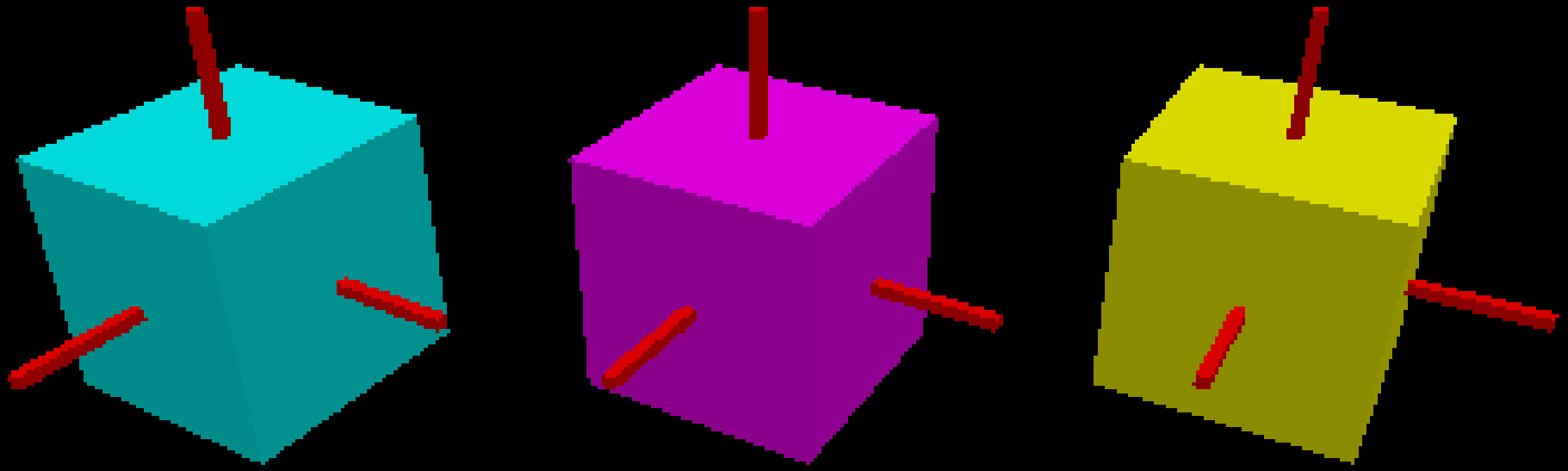


Transformaciones

box:cr 0

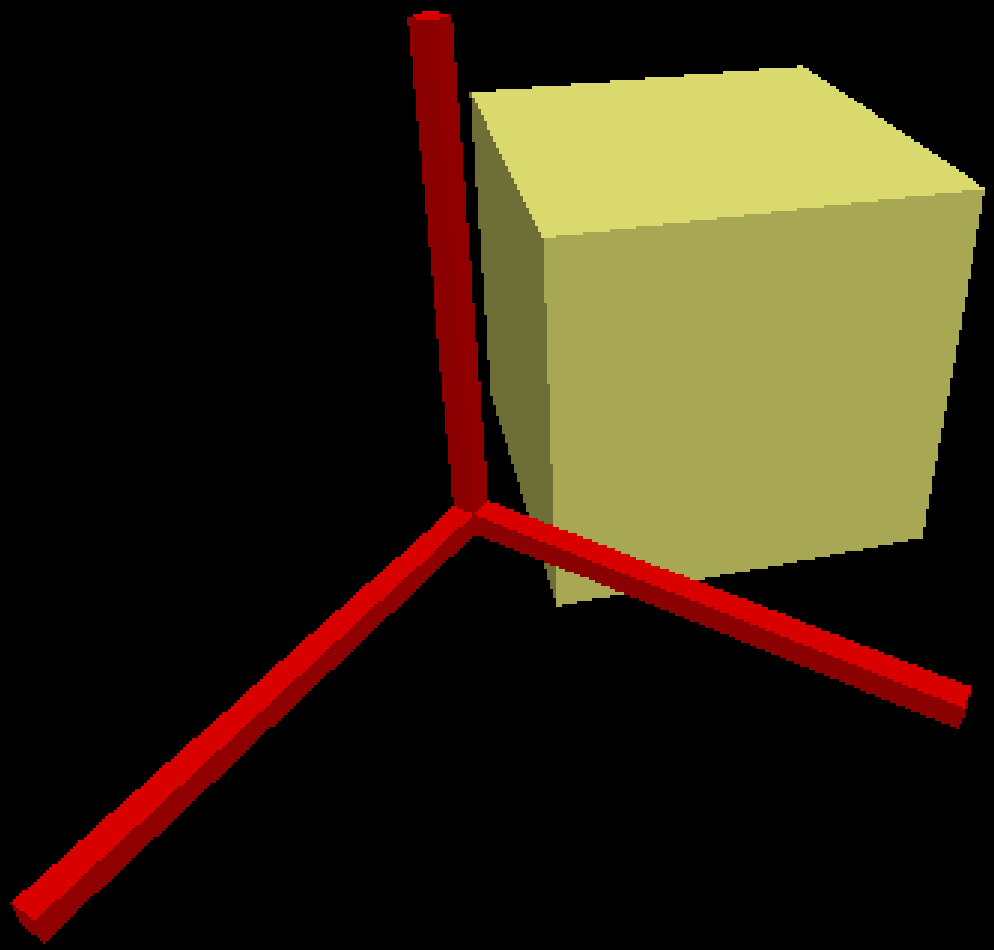
box:cg 0

box:cb 0



Transformaciones

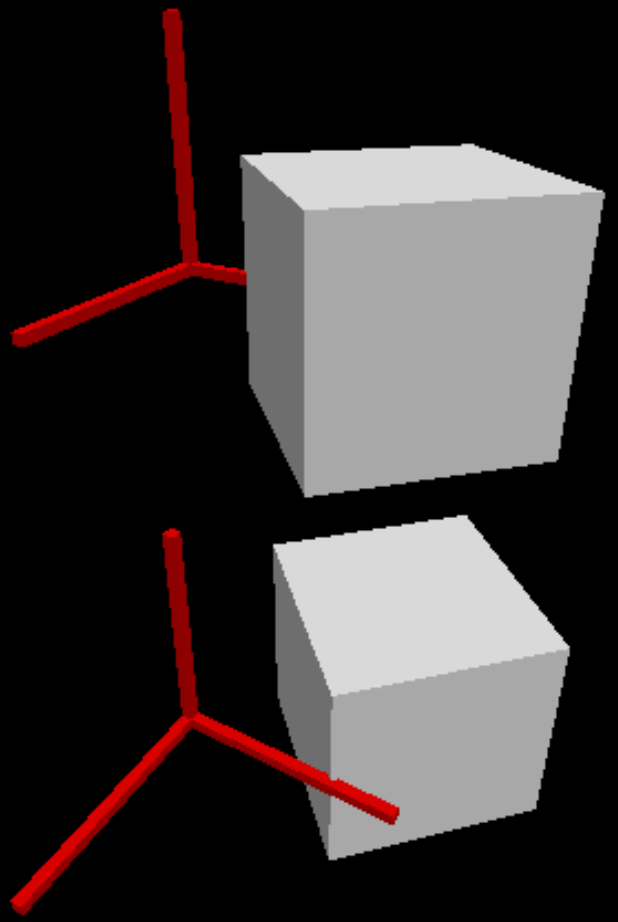
box : cb 0.5 : ry 360/8 : tz -1



Transformaciones

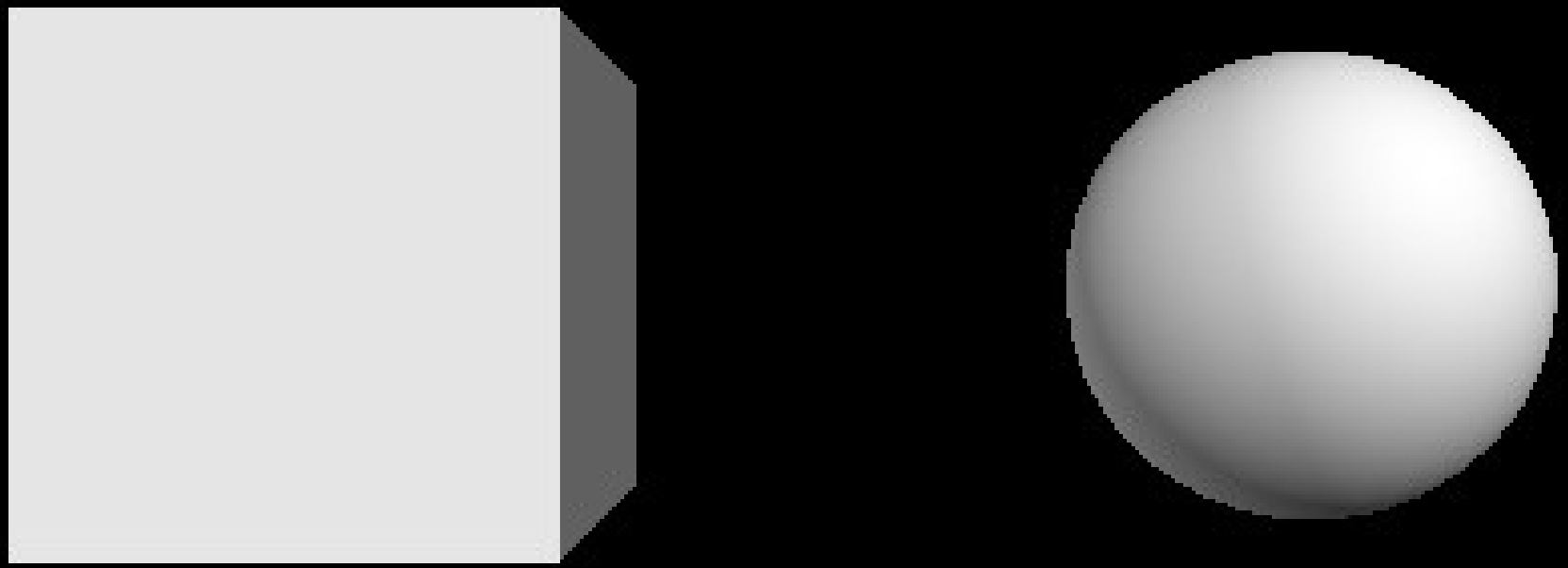
```
box:ry 45:tx  1
```

```
box:tx  1:ry 45
```



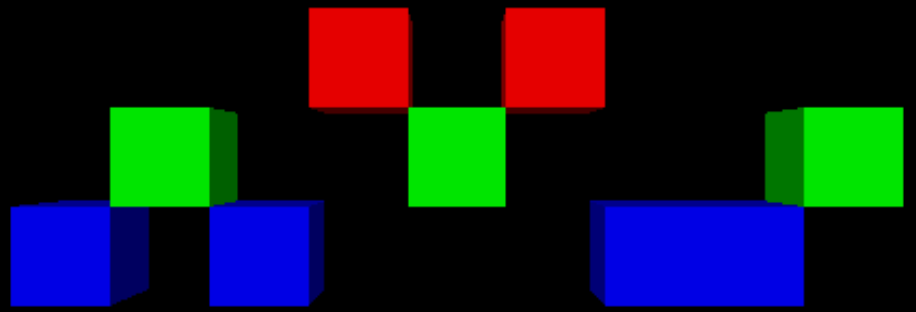
Operaciones

```
box:tx -1 & ball:tx 1
Box:tx -1 | ball:tx 1
```

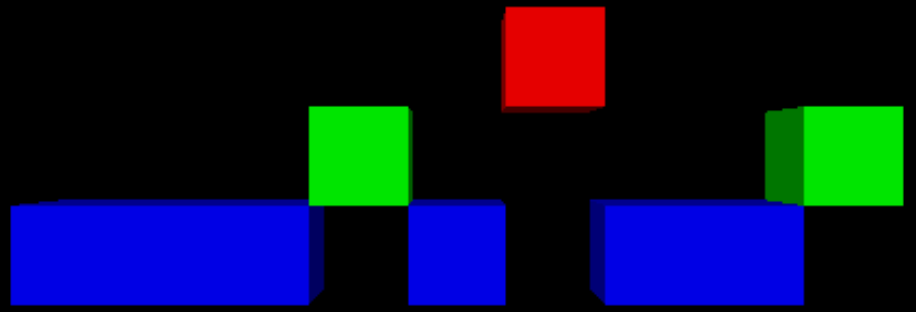


Operaciones

A | B | C

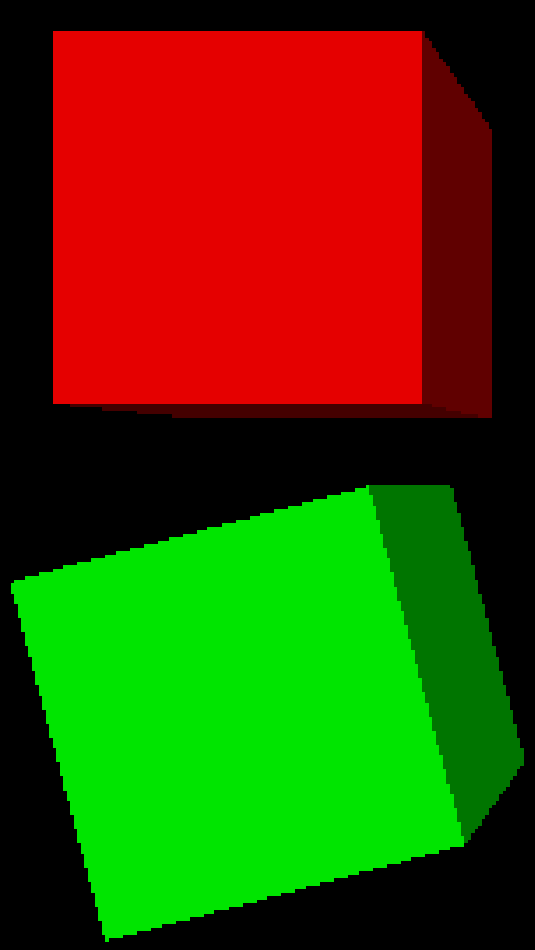


[[A | B]] | C

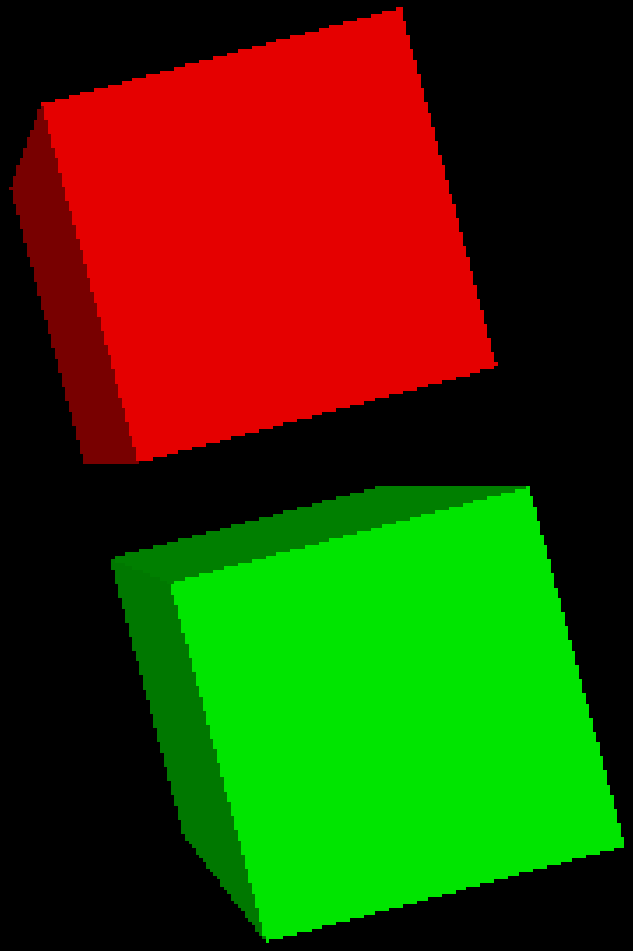


Operaciones

A & B:rz 15

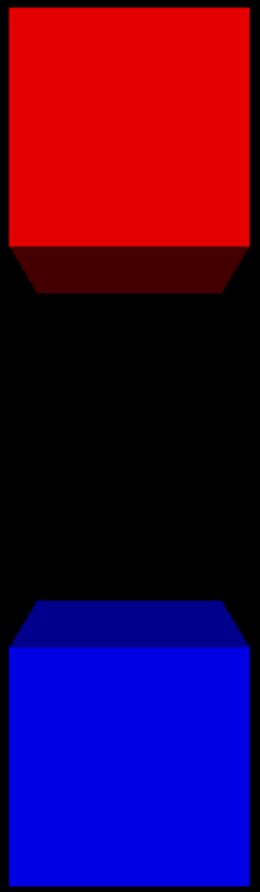


[[A & B]]:rz 15

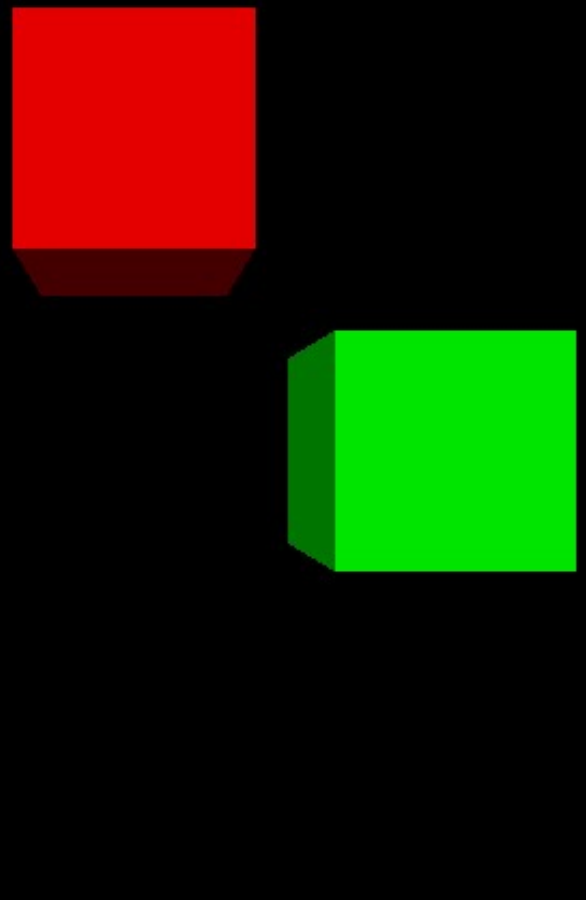


Operaciones

[A|B|C] ^3



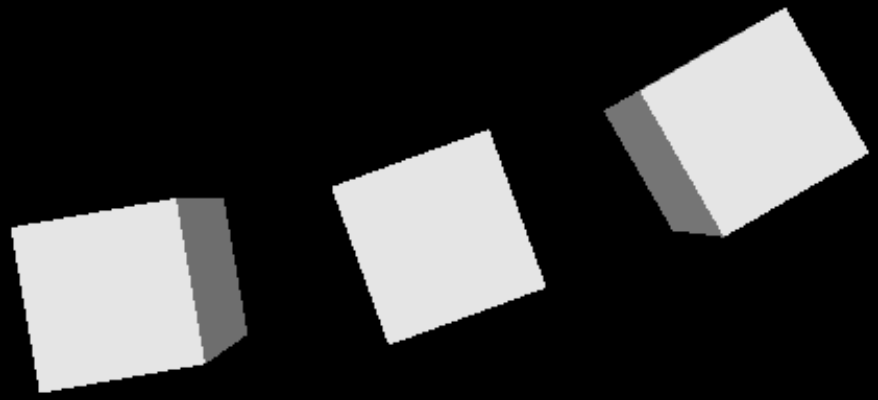
[A|B|C]:tx 1 ^3



Operaciones

```
[box:rz 10]
:tx 2 ^ 3
```

```
[box]:rz 10
:tx 2 ^ 3
```



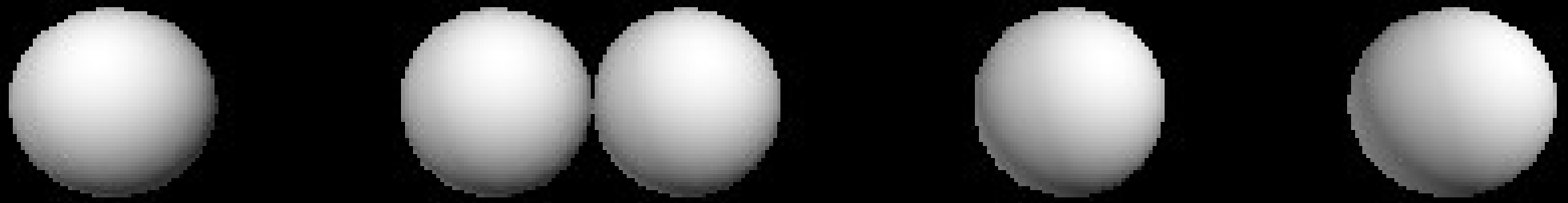
Operaciones

```
[box:rz 10]
:tx 2 ^ 3
```

```
box:rz 10
:tx 2 ^ 3
```

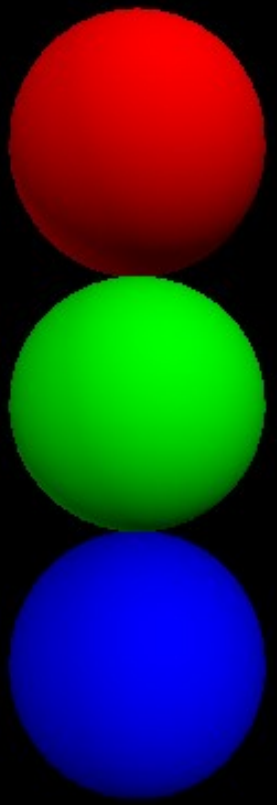
Operaciones

[[< ball >]]:tx 1 ^ 8



Reglas

```
A = ball:cg 0
:cb 0:ty 1

B = ball:cr 0
:cb 0

C = ball:cr 0
:cg 0:ty -1
```

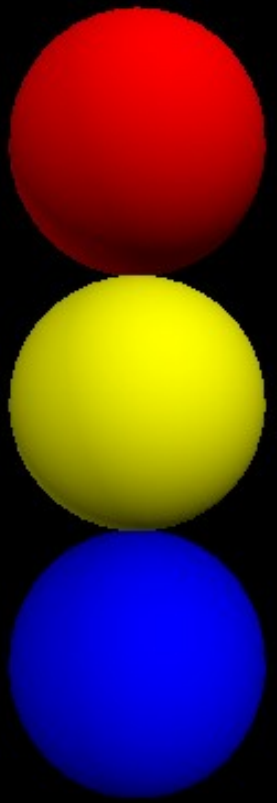


Reglas

```
A = ball:cg 0
:cb 0:ty 1

B = ball:cr 0
:cb 0

B = ball:cg 0

C = ball:cr 0
:cg 0:ty -1
```

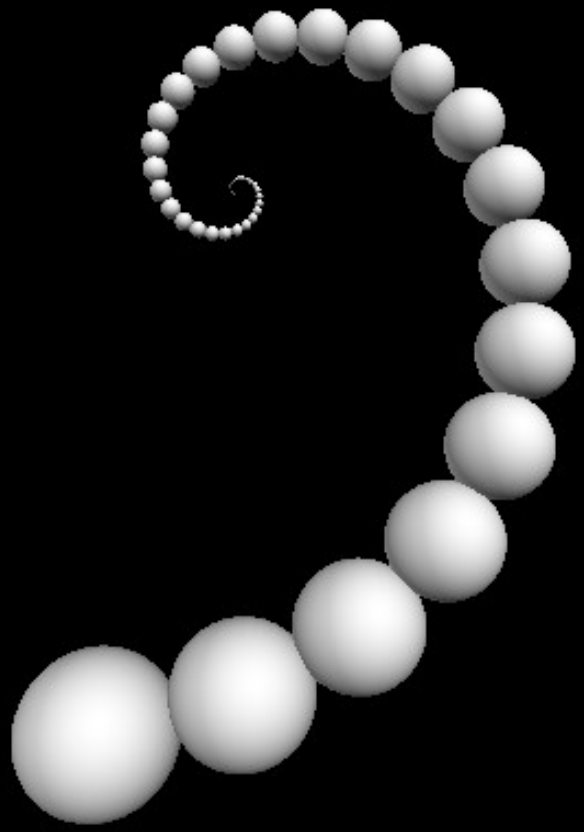
Reglas

```
A   = ball:cg 0
:cb 0:ty 1

B   = Bi | Bj

Bi  = ball:cb 0

Bj  = ball:cg 0

C   = ball:cr 0
:cg 0:ty -1
```



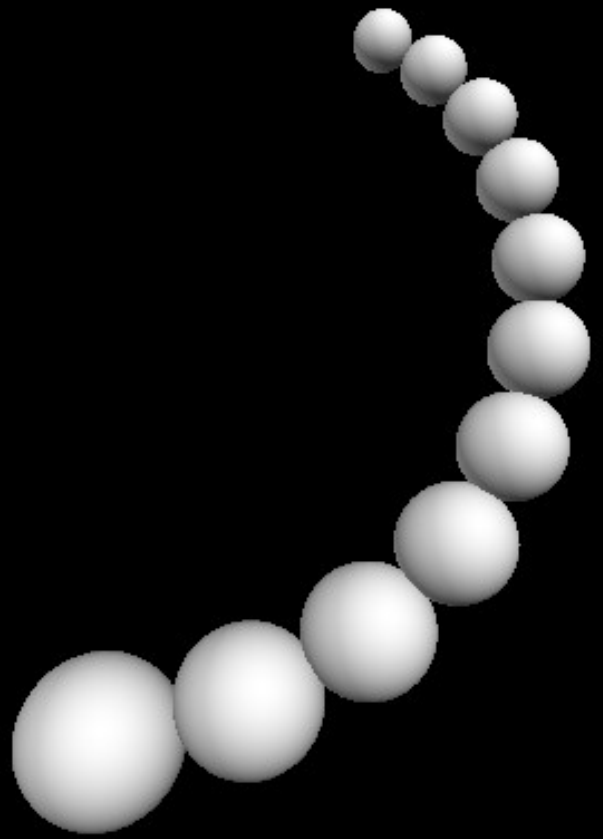
Reglas

```
spiral = ball
& spiral:tx 1
:rz 15:s 0.9
```



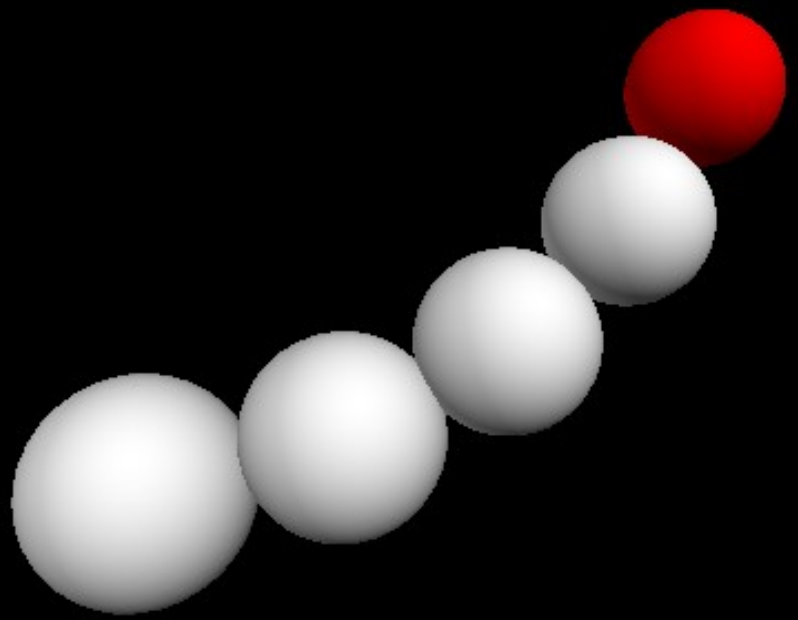
Reglas

```
scene  =
spiral:d 10


spiral = ball
& spiral:tx 1
:rz 15:s 0.9
```



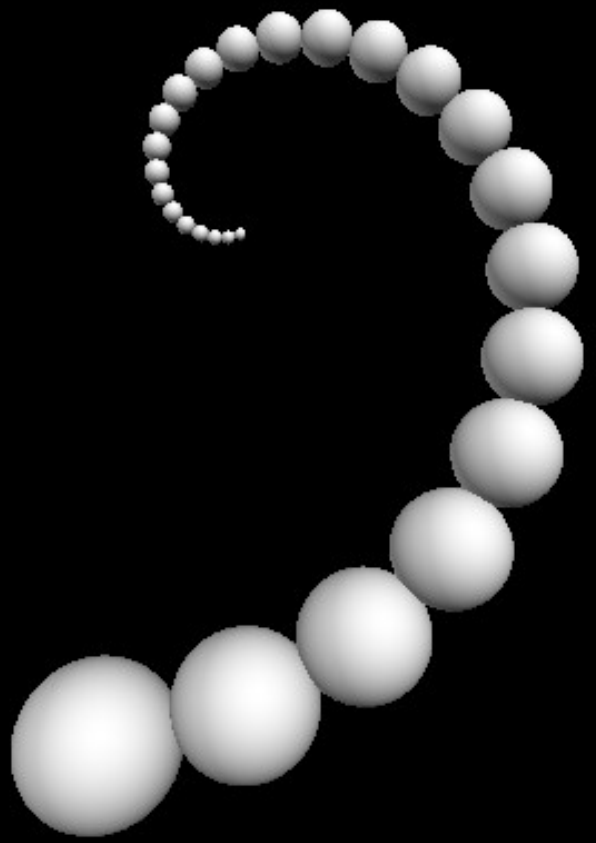
Reglas

```
scene  =
spiral:d 10

spiral = ball &
spiral:tx 1
:rz 15:s 0.9

spiral.=
ball:cg 0:cb 0
```



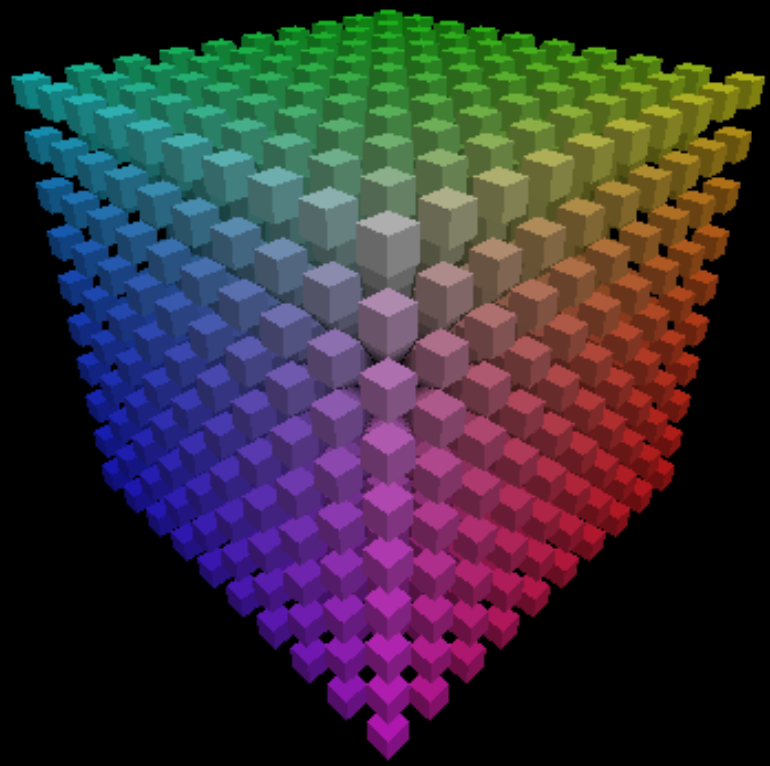
Reglas

```
spiral = ball
& spiral:tx 1
:rz 15:s 0.9

$ = spiral:d 25
```



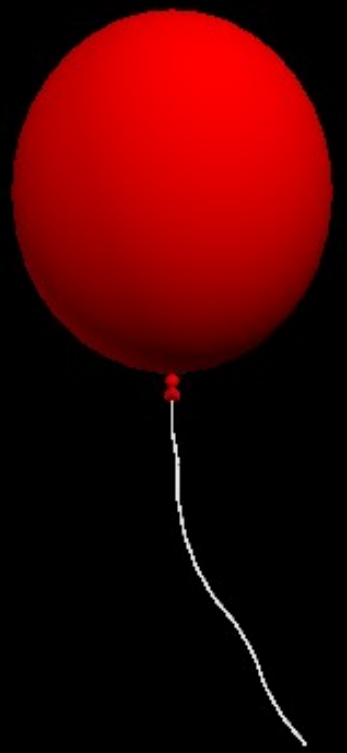
Ejemplos

```
cube =
[ [ [ box : s 0.1
    ] : tx -0.2 : cr 0.80 ^ 10
  ] : ty -0.2 : cg 0.80 ^ 10
] : tz -0.2 : cb 0.80 ^ 10


$ = cube : ry -45 : rx 30
```



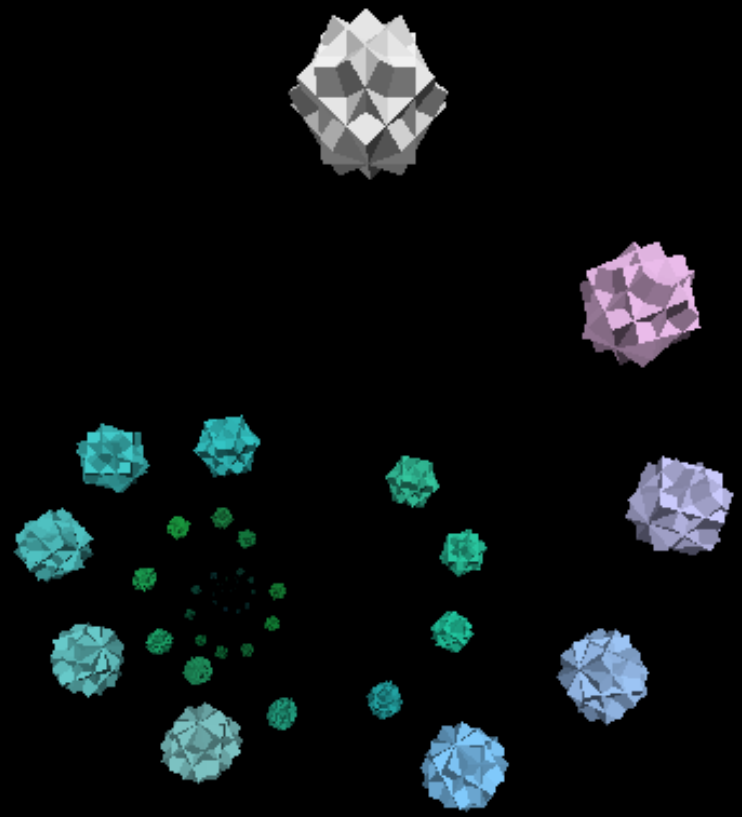
Ejemplos

```
balloon = ball:cg0:cb0:sx0.9:sz0.9
      & ball:cg0:cb0:s0.04:ty-0.52
      & box:cg0:cb0:s0.03
           :rz45:rx45:ty-0.56
      & line:ty-0.56:d10

line =
  box:sx0.01:sy0.1:sz0.01:ty-0.05
& [ line:rz10:ty-0.1
  | line:rz-10:ty-0.1 ]

$ = balloon
```



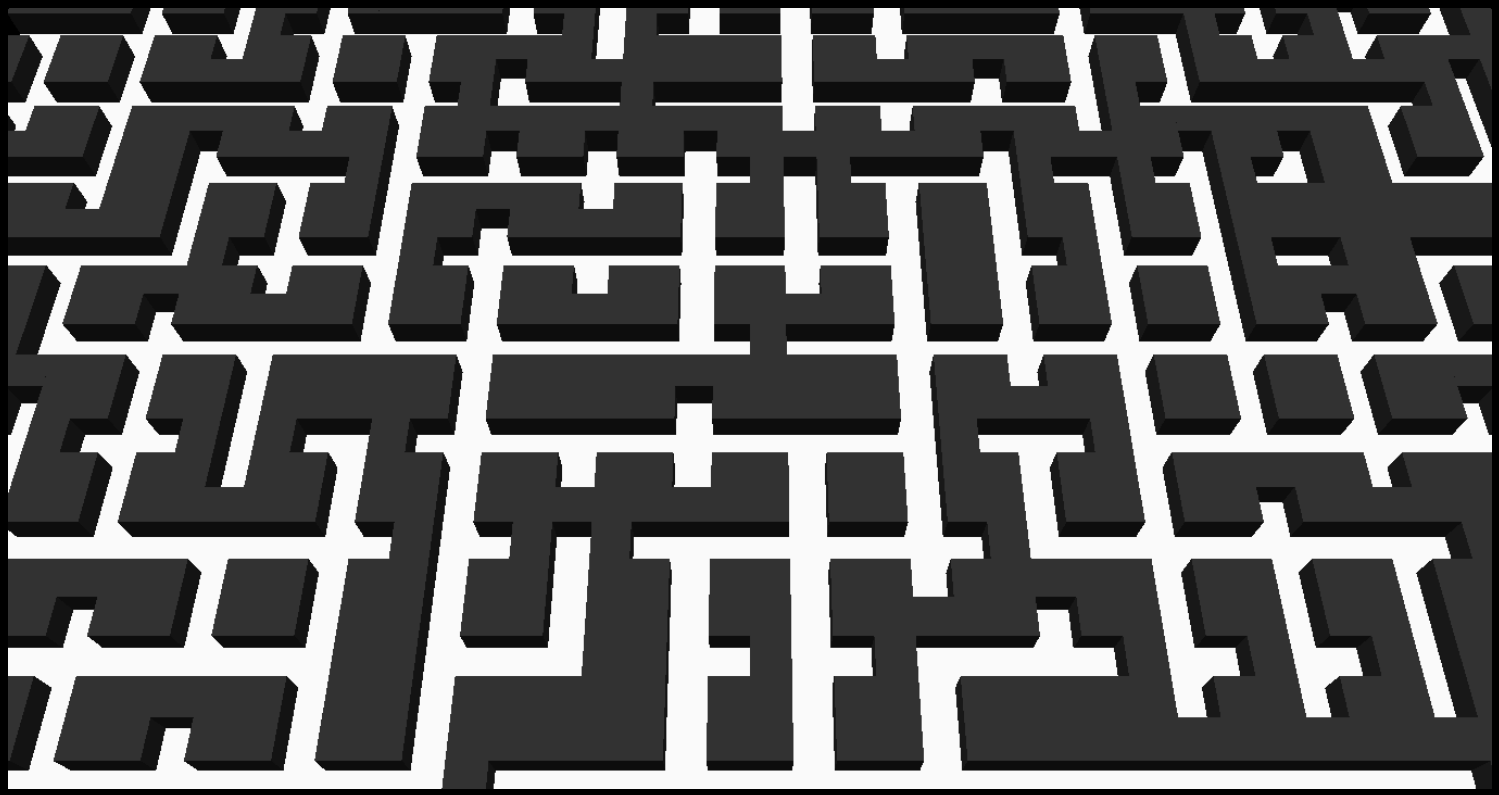
Ejemplos

```
spark = ⟦ box:rz45 ⟧:ry45 ^ 8


head = spark & tail

tail = head :s0.9
  :tx3 :rz-40 :ry10 :cr 0.8

tail = head :s0.9
  :tx4 :rz-30 :ry20 :cg 0.8

tail = head :s0.9
  :tx5 :rz-20 :ry30 :cb 0.8


$ = head
```



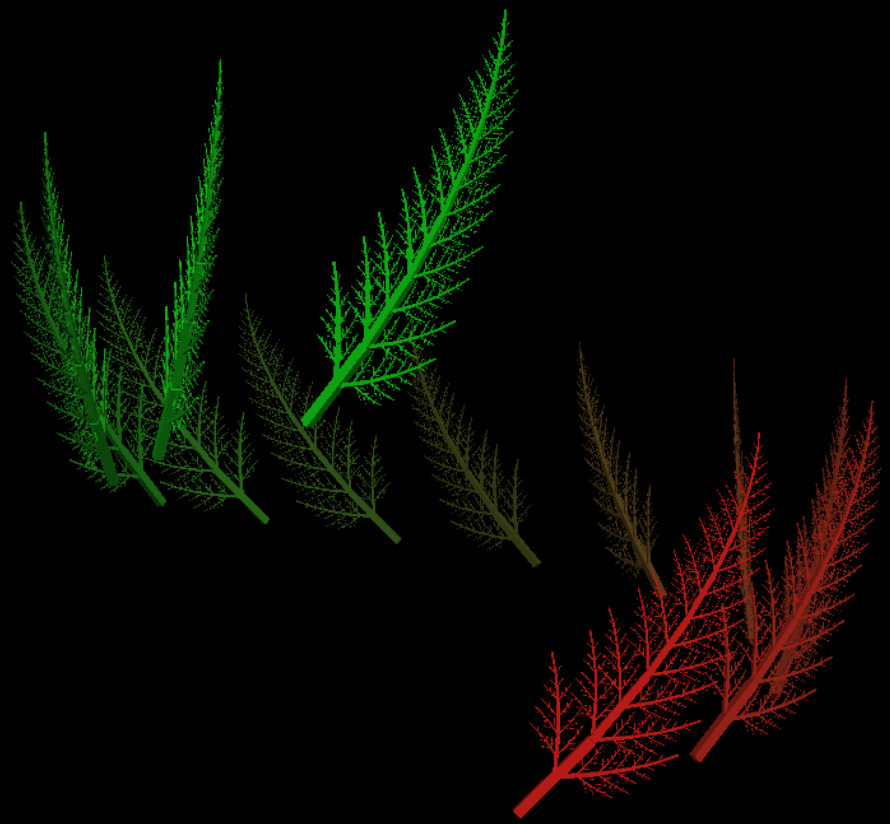
Ejemplos

```
maze = [ [ path :tx -8 :ty 8 ] :tx 1 ^ 16 ] :ty -1 ^ 16
path = box :sz 0.2 :tz 0.1 & walk:tz 1/3 & wall:tz 1/3
walk = [ box :s 1/3 :cr 0.2 :cg 0.2 :cb 0.2 :tx 2/6 :ty 2/6 ]:rz 90 ^ 4
wall = [ << box :s 1/3 :cr 0.2 :cg 0.2 :cb 0.2 :tx 2/6 >> ]:rz 90 ^ 4
$ = maze :rx -30
```



Ejemplos

```
leaf  = box:sx 0.1:sz 0.1:ty 0.5
      & leaf:s 0.40:rz -40:ty 0.5:d 3
      & leaf:s 0.40:rz +40:ty 0.5:d 3
      & leaf:s 0.35:rz -40:ty 1.0:d 3
      & leaf:s 0.35:rz +40:ty 1.0:d 3
      & leaf:s 0.8:rz 5:ty 1

autumn =
 [ leaf:rz -45:d 8:cg 0.1:cb 0.1 ]
 :tx 2:ty 1/3:ry 30:cr 0.8:cg 1.2
 ^ 12

$ = autumn:ry -35
```



Ejemplos

```
$ = arbol :d9

arbol   = tronco & copa :ty 1

tronco  = box :cr 0.6 :cg 0.5 :cb 0.1
              :sx 0.1 :sz 0.1 :ty 0.5

copa =
  [[ < arbol :s 0.7 :rz 30 :cg 0.9> ]]
  :ry 60 ^ 6

copa =
  [[ < arbol :s 0.7 :rz 30 :cg 0.7> ]]
  :ry 60 ^ 6

copa =
  [[ < arbol :s 0.7 :rz 30 :cg 1.2> ]]
  :ry 60 ^ 6

copa. = ball :cr 0.1 :cb 0.1 :s 2 :ty 1
```