

## Jによるペンローズ・タイルのグラフィックス

西川 利男

J壁紙群（格子群）グラフィックスを、最近いろいろやっている途中で、偶然、次のような興味深い本に出合った。

イアン・スチュアート著、梶山あゆみ訳「自然界の秘められたデザイン」p. 121-124  
河出書房新社、(2009)

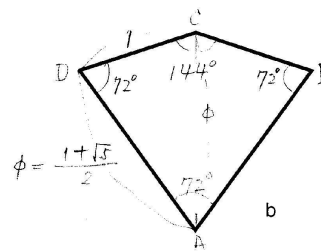
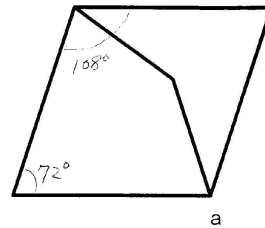
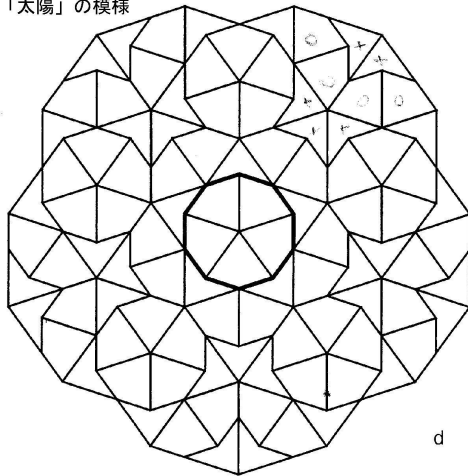
平面のタイリング(tiling)とは、平面を適当な多角形で、すきまなく敷き詰める問題である。正多角形であれば、正3角形、正方形、正6角形ならできるが、正5角形ではダメなことはすぐ分かる。

物理学者 R. Penrose によるペンローズ・タイルとは菱形を凧形(kite)と矢形(dart)の2つに分けて、それらを組み合わせて平面敷き詰め(tiling)を行うものである。

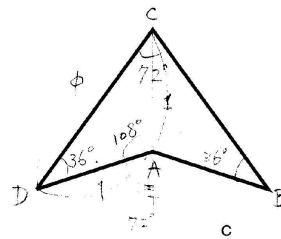
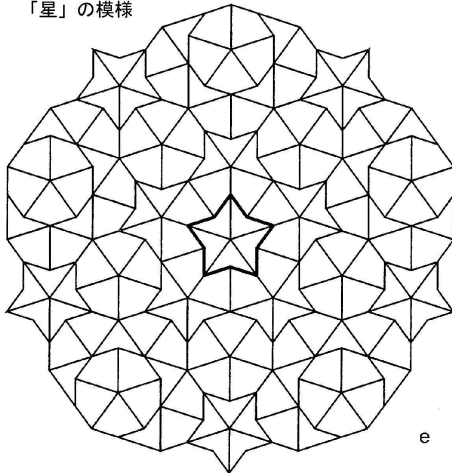
通常の格子より対称性は悪くなるが準格子として、金属材料の結晶の分野で、最近、興味を持たれている。

イアン・スチュアートの本より、ペンローズ・タイルの2つの例と作り方を示す。

「太陽」の模様



「星」の模様



## 1. 図形のタイリング（敷き詰め）とJグラフィックス

Jの哲学から、幾何学のシステムとはわたくしは次のように考えている。

幾何学図形…座標値の集合＝名詞

幾何学操作…図形の縮小拡大、回転、移動＝動詞

幾何学図形のコンピュータ画面上への表示＝動詞

したがって、タイリング（敷き詰め）の問題では、次のような一環した流れシステムとして、行いたいと思っている。

回転、移動などの操作（動詞）

図形の座標値（名詞） —————> 図形の座標値（名詞）

## 2. Jによる幾何学図形の座標値（＝名詞）と幾何学操作（＝動詞）

Jのgl2グラフィックスを使用し、以下の命令により作図することを前提にする。

gllines, glpolygon

・図形の座標値の集合はつぎのような単純リストとする。

X0, Y0, X1, Y1, X2, Y2, ... , Xn, Yn

・図形の回転、移動、画面表示など、その他関連の動詞をつぎのように定義した。

### 図形の回転 rotate

NB. eg. 30 rotate (x0, y0, x1, y1, .. , x0, y0)

rotate Figure(x0, y0, x1, y1, .. , x0, y0 ) around point(0, 0)

rot =: 3 : 0

:

'x y' =. y.

t =. x.

((x\*cosd t) - (y\*sind t)), ((x\*sind t) + (y\*cosd t))  
)

rotate =: 3 : 0

:

ANG =. x.

RXY =. cut2 y.

RX =. {."(1) > ANG rot L:0 RXY

RY =. {:"(1) > ANG rot L:0 RXY

, |: RX ,: RY

)

### 図形の移動 shift

NB. eg. (1, 2) shift x, y

shift Figure(x0, y0, x1, y1, .. , x0, y0 ) to Position(x, y)

shift =: 3 : 0

:

'Xs Ys' =. x.

, (Xs, Ys) +"(1) > cut2 y.

)

### 図形座標値の単純リストをボックス表示に変換 cut2

```
NB. cut2 i.10
NB. +---+---+---+---+---+
NB. |0 1|2 3|4 5|6 7|8 9|
NB. +---+---+---+---+---+
cut2 =: 3 : 0
y =. ((-:#y.),2)$y.
<"(1) y
)
```

### 色を指定して、多角形の図形を画面表示 colorpolyline

```
colorpolylines =: 3 : 0
:
glrgb x.
glbrush ''
glpen 1 0
gllines , sz adj0 y.
)
```

### 画面表示の際の図形の位置、大きさの調整

```
adj0 =: 3 : 0
1 adj0 y.
:
NB. 500 + 100 * x. * y.
(x0, y0) + "(1) > cut2 100 * x. * y.
NB. (500, 400) + "(1) > cut2 100 * x. * y.
)
```

### 3. 段階的 Penrose 図形開発のための要素パターンと基本タイルの作成

本稿ページ1の右の図を参考にして、凧形と矢形の図形の座標値を計算する。

```
PHI =: (1 + %: 5) % 2
KiteA =: 0, 0
KiteB =: PHI * (cosd 90 - (-: 72)), (sind 90 - (-: 72))
KiteH1 =: PHI * (sind 90 - (-: 72))
KiteH2 =: 1 * (cosd (-: 144))
KiteC =: (0, (KiteH1 + KiteH2))
KiteD =: PHI * (- cosd 90 - (-: 72)), (sind 90 - (-: 72))
Kite =: KiteA, KiteB, KiteC, KiteD, KiteA
Kite1 =: , 180 rotate , > Kite

DartA =: 0, cosd 72
DartB =: (sind 72), 0
DartC =: 0, PHI*cosd (-: 72)
DartD =: (- sind 72), 0
Dart =: DartA, DartB, DartC, DartD, DartA
```

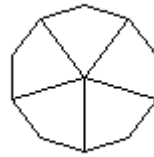
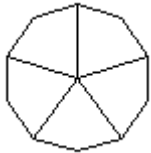
つぎに、凧形と矢形の図形の座標値を元に、前節の rotate 動詞と shift 動詞を使って要素パターンを作成する。(左から順に示す)

**Kite** 凧形図形を  $72^\circ$  ずつ回転させた図形を 5 つ組み合わせる。

**DartKite** 矢形図形の左右に凧形図形を貼り付ける。

**KiteY** Kite パターンを上下回転する。

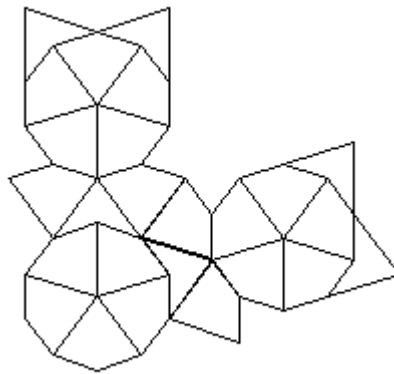
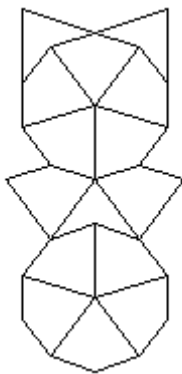
**Dart2** KiteY パターンに接触するよう、矢形図形を 2 つ回転して組み合わせる。



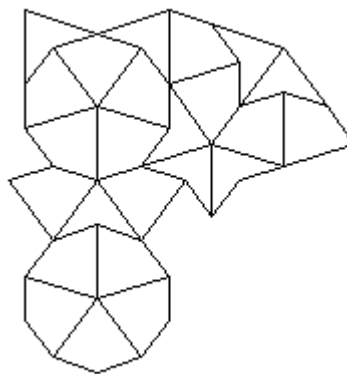
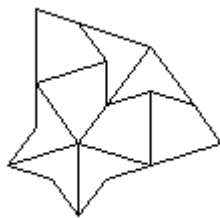
さらに、  
要素パターンを、組み  
**Penr** とを作る。

合わせて、2 つの基本タイル **Pen** と

基本タイル **Pen** はこれを 5 回対称回転して Penrose 図形をつくるものである。



また、基本タイル **Penr** は、上でまだ余白として残ってしまった部分を埋めるためのものである。



以上のようにして、Penrose タイル・システム作成の準備が整った。

#### 4. Penrose タイル・システムの実行の実際

本システムの特徴は Penrose タイル図形の完成図を示すのはもちろんだが、その途中経過をステップを追って、どう作られるのかを観察できるようにしたことにある。多数のボタンを設けたのはそのためである。下の Kite0 からのボタンは、凧形図形を順次組み合わせて、要素パターン Kite の作成の手順を示す。

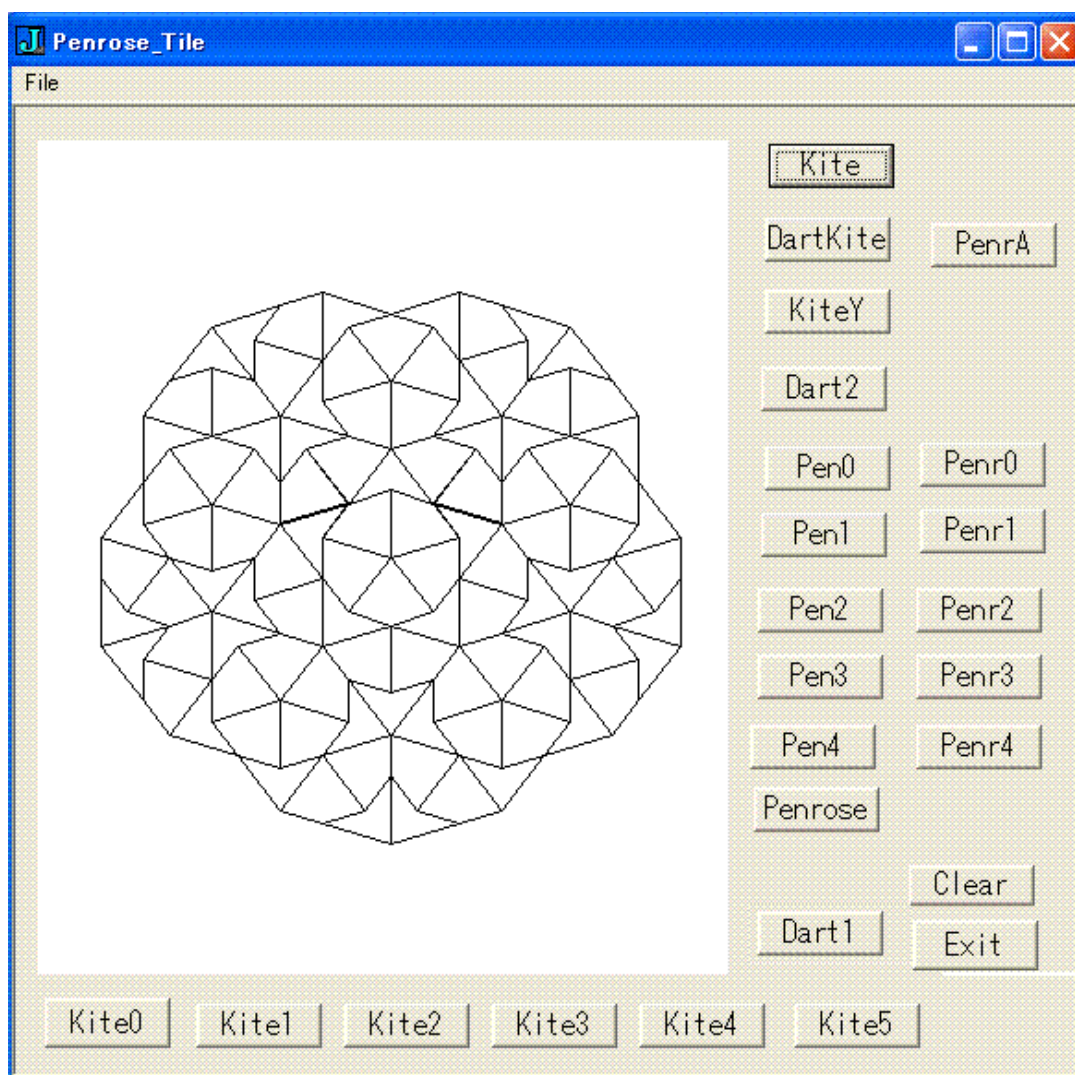
実行はつぎのように段階的に行う。

まず、**Kite** ボタンを押して中心部を作る。続いて **DartKite**, **KiteY**, **Dart2** ボタンを順次押すと、縦に最初のタイルが現れる。この操作は Pen0 ボタンでもできる。

**Pen1**, **Pen2**, **Pen3**, **Pen4** と次々に押して行こう。残念ながら、途中で余白部分が残る。次に、**Penr0** ボタンを押すと、うまくつながる。前と同様に

**Penr1**, **Penr2**, **Penr3**, **Penr4** と次々に押して行って、Penrose タイル図形が完成する。

図形表示がはみだしてしまったときには、マウスの左ボタンで、縮小して収められる。また、マウスの右ボタンでは、図形の中心をその位置に移動させ、好きな場所にきれいなパターンを表示させることができる。



NB. Penrose.ijs

NB. Penrose Tiling

NB. from wallpaper2.ijs

NB. by TN 2017/2/18  
NB. for JAPLA 2017/3/11

```
require 'trig'  
require 'gl2'  
  
PENROSE=: 0 : 0  
pc penrose;pn "Penrose_Tile";  
menupop "File";  
menu new "&New" "" "" "";  
menu open "&Open" "" "" "";  
menusep ;  
menu exit "&Exit" "" "" "";  
menupopz;  
xywh 8 215 34 11;cc Kite0 button;  
xywh 297 215 34 11;cc cancel button;cn "Exit";  
xywh 6 8 231 201;cc PenTile isigraph;  
xywh 45 216 34 11;cc Kite1 button;  
xywh 84 216 34 11;cc Kite2 button;  
xywh 123 216 34 11;cc Kite3 button;  
xywh 162 216 34 11;cc Kite4 button;  
xywh 203 216 34 11;cc Kite5 button;  
xywh 246 9 34 11;cc Kite button;  
xywh 247 44 34 11;cc KiteY button;  
xywh 253 197 34 11;cc Dart1 button;  
xywh 251 215 34 11;cc Clear button;  
xywh 246 27 34 11;cc DartKite button;  
xywh 249 82 34 11;cc Pen0 button;  
xywh 249 98 34 11;cc Pen1 button;  
xywh 249 116 34 11;cc Pen2 button;  
xywh 249 132 34 11;cc Pen3 button;  
xywh 249 148 34 11;cc Pen4 button;  
xywh 249 164 34 11;cc Penrose button;  
xywh 295 197 34 11;cc PenrA button;  
xywh 293 81 34 11;cc Penr0 button;  
xywh 293 97 34 11;cc Penr1 button;  
xywh 293 116 34 11;cc Penr2 button;  
xywh 293 132 34 11;cc Penr3 button;  
xywh 293 149 34 11;cc Penr4 button;  
xywh 248 63 34 11;cc Dart2 button;  
pas 6 6;pcenter;  
rem form end;  
)  
  
run =: penrose_run
```

```

penrose_run=: 3 : 0
wd PENROSE
NB. initialize form here
x0 =: 500
y0 =: 500
sz =: 0.5
wd 'pshow;'
)

```

```

penrose_close=: 3 : 0
wd'pclose'
)

```

```

penrose_cancel_button=: 3 : 0
penrose_close''
)

```

```

penrose_Clear_button=: 3 : 0
glclear ''
glshow ''
)

```

```

NB. =====
adj0 =: 3 : 0    NB. centered
1 adj0 y.
:
NB. 500 + 100 * x. * y.
(x0, y0) + "(1) > cut2 100 * x. * y.  NB. y-axis down
NB. (500, 400) + "(1) > cut2 100 * x. * y.  NB. y-axis down
)

```

```

adj =: 3 : 0    NB. start left down
1 adj y.
:
NB. (500&+)@(100&*)
200 + 40 * x. * y.
)

```

```

NB. cut2 i.10

```

```

NB. +---+---+---+---+---+
NB. |0 1|2 3|4 5|6 7|8 9|
NB. +---+---+---+---+---+
cut2 =: 3 : 0
y =. ((-:#y.),2)$y.
<"(1) y
)

```

```

NB. eg. 30 rotate figure(x0, y0, x1, y1, .. ) around (0, 0)
rot =: 3 : 0
:
'x y' =. y.
t =. x.
((x*cosd t) - (y*sind t)), ((x*sind t) + (y*cosd t))
)

```

```

rotate =: 3 : 0
:
ANG =. x.
RXY =. cut2 y.
RX =. {"(1) > ANG rot L:0 RXY
RY =. {"(1) > ANG rot L:0 RXY
, |: RX ,: RY
)

```

```

NB. eg. (1, 2) shift x, y
shift =: 3 : 0
:
'Xs Ys' =. x.
, (Xs, Ys) +"(1) > cut2 y.
)

```

```

colorpolygon =: 3 : 0
:
glrgb x.
glbrush ''
glpen 1 0
glpolygon , adj0 y.
)

```

```

colorpolylines =: 3 : 0
:
glrgb x.
glbrush ''

```



```

glpen 1 0
gllines , sz adj0 y.
)

```

```

draw_xy =: 3 : 0
glrgb 0 0 0
glpen 1 0
gllines 10 500 990 500
gllines 500 10 500 990
)

```

```

color_pen4 =: 3 : 0
if. y. = 0 do. glrgb 255 0 0 end.
if. y. = 1 do. glrgb 255 255 0 end.
if. y. = 2 do. glrgb 0 255 0 end.
if. y. = 3 do. glrgb 0 0 255 end.
glbrush ''
glpen 4 0
)

```

NB. Figure Kite =====

```

PHI =: (1 + %: 5) % 2
KiteA =: 0, 0
KiteB =: PHI * (cosd 90 - (-: 72)), (sind 90 - (-: 72))
KiteH1 =: PHI * (sind 90 - (-: 72))
KiteH2 =: 1 * (cosd (-: 144))
KiteC =: (0, (KiteH1 + KiteH2))
KiteD =: PHI * (- cosd 90 - (-: 72)), (sind 90 - (-: 72))
Kite =: KiteA, KiteB, KiteC, KiteD, KiteA
Kite1 =: , 180 rotate , > Kite

```

```

penrose_Kite0_button=: 3 : 0
(0 0 0) colorpolylines Kite
draw_xy ''
glshow ''
)

```

```

penrose_Kite1_button=: 3 : 0
Kite1 =: , 180 rotate , > Kite
(255 0 0) colorpolylines Kite1
glshow ''
)

```

```

penrose_Kite2_button=: 3 : 0
Kite2 =: , (1*72) rotate , > Kite1
(0 255 0) colorpolylines Kite2
glshow ''
)

```

```

penrose_Kite3_button=: 3 : 0
Kite3 =: , (2*72) rotate , > Kite1
(0 0 255) colorpolylines Kite3
glshow ''
)

```

```

penrose_Kite4_button=: 3 : 0
Kite4 =: , (3*72) rotate , > Kite1
(0 255 255) colorpolylines Kite4
glshow ''
)

```

```

penrose_Kite5_button=: 3 : 0
Kite5 =: , (4*72) rotate , > Kite1
(0 0 0) colorpolylines Kite5
glshow ''
)

```

```

penrose_Kite_button=: 3 : 0
Kites =: ((4*72) rotate Kite1), ((3*72) rotate Kite1), ((2*72) rotate Kite1),
(72 rotate Kite1), (0 rotate Kite1)
(0 0 0) colorpolylines Kites
glshow ''
)

```

```

penrose_KiteY_button=: 3 : 0
Kitess =: 180 rotate Kites
NB. KiteY =: , (0, (1 + 2* PHI)) +"(1 1) > cut2 Kitess
KiteY =: (0, (1 + 2* PHI)) shift Kitess
(0 0 0) colorpolylines KiteY
glshow ''
)

```

```

NB. Figure Dart =====
DartA =: 0, cosd 72
DartB =: (sind 72), 0
DartC =: 0, PHI*cosd (-: 72)
DartD =: (- sind 72), 0

```

```
Dart =: DartA, DartB, DartC, DartD, DartA
```

```
penrose_Dart1_button=: 3 : 0
Dart1 =: , (0, PHI*(cosd -: 72)) shift Dart
(0 0 0) colorpolylines Dart1
draw_xy ''
glshow ''
)
```

```
penrose_DartKite_button=: 3 : 0
KiteR =. , ((0.15 + cosd 36), 0) shift Kite
NB. (0 0 0) colorpolylines KiteR
KiteL =. , ((- 0.15 + cosd 36), 0) shift Kite
NB. (0 0 0) colorpolylines KiteL
DartRL =: KiteR, Dart, KiteL
NB. (0 0 0) colorpolylines DartRL
DartY =: (0, PHI*(cosd -: 72)) shift DartRL
(0 0 0) colorpolylines DartY
glshow ''
)
```

```
penrose_Dart2_button=: 3 : 0
Dart30 =. , (_36) rotate "(1) > cut2 Dart
Dart30x =. (_2 {. _2 }. Dart30), (_2 }. Dart30) NB. start point from
A to D
Dart30y =. 2 {. (_2 {. _2 }. Dart30), (_2 }. Dart30)
Dart31 =. , (- Dart30y) shift Dart30x
Dart32 =. , > (_1, 1) * L:0 cut2 Dart31
Dart33 =. Dart31, Dart32
Dart2 =: Dart34 =. , (0, (1 + 3*PHI)) shift Dart33 NB. global
(0 0 0) colorpolylines Dart2
glshow ''
)
```

```
penrose_Pen0_button=: 3 : 0
Kites =. ((4*72) rotate KiteL), ((3*72) rotate KiteL), ((2*72) rotate KiteL),
(72 rotate KiteL), (0 rotate KiteL)
Kitess =. 180 rotate Kites
PenK0 =: (0, (1 + 2* PHI)) shift Kitess NB. global
(0 0 0) colorpolylines PenK0
KiteR =. , ((0.15 + cosd 36), 0) shift Kite
KiteL =. , ((- 0.15 + cosd 36), 0) shift Kite
```

```

DartRL =. KiteR, Dart, KiteL
PenD0 =: (0, PHI*(cosd -: 72)) shift DartRL      NB. global
NB. Pen0 =: KiteY, DartY
(0 0 0) colorpolylines PenD0
(0 0 0) colorpolylines Dart2

```

```

glshow ''
)

```

```

penrose_Pen1_button=: 3 : 0
PenK1 =. , _72 rotate "(1) > cut2 PenK0
(0 0 0) colorpolylines PenK1
PenD1 =. , _72 rotate "(1) > cut2 PenD0
(0 0 0) colorpolylines PenD1
PenDD1 =. , _72 rotate "(1) > cut2 Dart2
(0 0 0) colorpolylines PenDD1
glshow ''
)

```

```

penrose_Pen2_button=: 3 : 0
PenK2 =. , (_2*72) rotate "(1) > cut2 PenK0
(0 0 0) colorpolylines PenK2
PenD2 =. , (_2*72) rotate "(1) > cut2 PenD0
(0 0 0) colorpolylines PenD2
PenDD2 =. , (_2*72) rotate "(1) > cut2 Dart2
(0 0 0) colorpolylines PenDD2
glshow ''
)

```

```

penrose_Pen3_button=: 3 : 0
PenK3 =. , (_3*72) rotate "(1) > cut2 PenK0
(0 0 0) colorpolylines PenK3
PenD3 =. , (_3*72) rotate "(1) > cut2 PenD0
(0 0 0) colorpolylines PenD3
PenDD3 =. , (_3*72) rotate "(1) > cut2 Dart2
(0 0 0) colorpolylines PenDD3
glshow ''
)

```

```

penrose_Pen4_button=: 3 : 0
PenK4 =. , (_4*72) rotate "(1) > cut2 PenK0
(0 0 0) colorpolylines PenK4
PenD4 =. , (_4*72) rotate "(1) > cut2 PenD0
(0 0 0) colorpolylines PenD4
PenDD4 =. , (_4*72) rotate "(1) > cut2 Dart2

```

```
(0 0 0) colorpolylines PenDD4
glshow ''
)
```

```
NB. Penrose whole figure =====
penrose_Penrose_button=: 3 : 0
glclear ''
penrose_Kite_button ''
penrose_Pen0_button ''
penrose_Pen1_button ''
penrose_Pen2_button ''
penrose_Pen3_button ''
penrose_Pen4_button ''
penrose_Penr0_button ''
penrose_Penr1_button ''
penrose_Penr2_button ''
penrose_Penr3_button ''
penrose_Penr4_button ''
glshow ''
)
```

```
NB. mouse left button => figure move to click position
penrose_PenFile_mbltdown=: 3 : 0
d=. ". sysdata
x=. (0{d) * 1000 % (2{d)
y=. (1{d) * 1000 % (3{d)
x0 =: x
y0 =: y
penrose_Penrose_button ''
penrose_Penr0_button ''
penrose_Penr1_button ''
penrose_Penr2_button ''
penrose_Penr3_button ''
penrose_Penr4_button ''
penrose_Penr0_button ''
penrose_Penr1_button ''
penrose_Penr2_button ''
penrose_Penr3_button ''
penrose_Penr4_button ''
)
```

```
NB. mouse right buton => figure size will be smaller
penrose_PenFile_mbrdown=: 3 : 0
sz =: 0.8 * sz
penrose_Penrose_button ''
```

```

penrose_Penr0_button ''
penrose_Penr1_button ''
penrose_Penr2_button ''
penrose_Penr3_button ''
penrose_Penr4_button ''
penrose_Penr0_button ''
penrose_Penr1_button ''
penrose_Penr2_button ''
penrose_Penr3_button ''
penrose_Penr4_button ''
)

```

```

penrose_PenrA_button=: 3 : 0
Dart1 =: , (0, PHI*(cosd -: 72)) shift Dart NB. global used in Penr0,...,
Penr4
Kite3 =: , (2*72) rotate , > Kite1 NB. global used in Penr0,...,
Penr4
Kite4 =: , (3*72) rotate , > Kite1 NB. global used in Penr0,...,
Penr4
PenrA0 =. , (0, ((1 + PHI)) ) shift Dart
PenrA1 =. , (0, _0.61) shift Kite3
PenrA2 =. , (0, _0.61) shift Kite4
PenrA4 =. Dart1, Kite3, Kite4
NB. (0 0 0) colorpolylines PenrA4
PenrA5 =. , (0, (- 1+PHI)) shift PenrA4
NB. (0 0 0) colorpolylines PenrA5
PenrA6 =. , (_36) rotate "(1) > cut2 PenrA5
NB. (0 0 0) colorpolylines PenrA6
PenrA7 =. , (36) rotate "(1) > cut2 PenrA5
NB. (0 0 0) colorpolylines PenrA7
PenrA8 =: , (0, ((1+PHI)) ) shift PenrA6 NB. global used in Penr0,...,
Penr4
NB. (0 0 0) colorpolylines PenrA8
PenrA9 =: , (0, ((1+PHI)) ) shift PenrA7 NB. global used in Penr0,...,
Penr4
NB. (0 0 0) colorpolylines PenrA9
PenrA10 =: , (0, (-(1 + cosd 72) )) shift Dart NB. global used in Penr0,...,
Penr4
NB. (0 0 0) colorpolylines PenrA10
PenrA11 =: , (72) rotate "(1) > cut2 PenrA10 NB. global used in Penr0,...,
Penr4
NB. (0 0 0) colorpolylines PenrA11
PenrA12 =: , (_72) rotate "(1) > cut2 PenrA10 NB. global used in Penr0,...,
Penr4

```

```
NB. (0 0 0) colorpolylines PenrA12
glshow ''
)
```

```
penrose_Penr0_button=: 3 : 0
penrose_PenrA_button ''
Penr08 =. , (_36) rotate "(1) > cut2 , (0, (1 + 2*PHI)) shift PenrA8
(0 0 0) colorpolylines Penr08
Penr09 =. , (_36) rotate "(1) > cut2 , (0, (1 + 2*PHI)) shift PenrA9
(0 0 0) colorpolylines Penr09
Penr010 =. , (_36) rotate "(1) > cut2 , (0, (1 + 2*PHI)) shift PenrA10
(0 0 0) colorpolylines Penr010
Penr011 =. , (_36) rotate "(1) > cut2 , (0, (1 + 2*PHI)) shift PenrA11
(0 0 0) colorpolylines Penr011
Penr012 =. , (_36) rotate "(1) > cut2 , (0, (1 + 2*PHI)) shift PenrA12
(0 0 0) colorpolylines Penr012
glshow ''
)
```

```
penrn =: 3 : 0
PRN =. y.
penrose_PenrA_button ''
Penr08 =. , (PRN * _36) rotate "(1) > cut2 (0, (1 + 2*PHI)) shift PenrA8
(0 0 0) colorpolylines Penr08
Penr09 =. , (PRN * _36) rotate "(1) > cut2 (0, (1 + 2*PHI)) shift PenrA9
(0 0 0) colorpolylines Penr09
Penr010 =. , (PRN * _36) rotate "(1) > cut2 (0, (1 + 2*PHI)) shift PenrA10
(0 0 0) colorpolylines Penr010
Penr011 =. , (PRN * _36) rotate "(1) > cut2 (0, (1 + 2*PHI)) shift PenrA11
(0 0 0) colorpolylines Penr011
Penr012 =. , (PRN * _36) rotate "(1) > cut2 (0, (1 + 2*PHI)) shift PenrA12
(0 0 0) colorpolylines Penr012
glshow ''
)
```

```
penrose_Penr1_button=: 3 : 0
penrn 3
)
```

```
penrose_Penr2_button=: 3 : 0
penrn 5
)
```

```
penrose_Penr3_button=: 3 : 0
```

```
penrn 7  
)
```

```
penrose_Penr4_button=: 3 : 0  
penrn 9  
)
```