

The Penrose package

Andrew Stacey

loopspace@mathforge.org

v1.4 from 2021/01/22

Contents

1	Introduction	1
2	Implementation	1
2.1	Initialisation	1
2.2	Creating the Tiles	2
2.3	Specifying the Tiles	6
2.4	User-Definable Tiles	11
2.5	Lindenmayer System	16
2.6	TikZ Pictures	28

1 Introduction

This is a TikZ library for drawing Penrose tiles (kite/dart, rhombus, and pentagon versions). It provides two methods of drawing: one in which an automatic pattern is built, and one where the tiles can be placed “by hand”. The tiles can be shaped and (hopefully!) still fit together. For full user documentation, see the `penrose.pdf` file.

2 Implementation

```
1 <*package>
2 <@@=penrose>
```

2.1 Initialisation

We use the `spath3` library for manipulating the paths that will make up the tiles.

```
3 \ProvidesExplFile {tikzlibrarypenrose.code.tex} {2021/01/22} {1.4} {TikZ pics for Penrose an
4 \RequirePackage{spath3}
5 \usetikzlibrary{spath3}
```

Now we move in to the realm of L^AT_EX3.

```
6 \ExplSyntaxOn
```

Start with some basic paths (lines) for the sides of the tiles so that we know that we have well-defined tiles at the outset. These are globally defined as we will frequently want to define them in one tikzpicture and use them in another.

```

7 \tl_new:N \g__penrose_side_a_tl
8 \tl_new:N \g__penrose_side_b_tl
9 \tl_new:N \g__penrose_side_c_tl
10 \tl_new:N \g__penrose_side_d_tl
11 \tl_new:N \g__penrose_side_e_tl
12 \tl_new:N \g__penrose_side_A_tl
13 \tl_new:N \g__penrose_side_B_tl
14 \tl_new:N \g__penrose_side_C_tl
15 \tl_new:N \g__penrose_side_D_tl
16 \tl_new:N \g__penrose_side_E_tl
17 \tl_gset:Nn \g__penrose_side_a_tl
18 {
19   \pgfsyssoftpath@movetotoken{0pt}{0pt}
20   \pgfsyssoftpath@linetotoken{1pt}{0pt}
21 }
22 \tl_gset_eq:NN \g__penrose_side_b_tl \g__penrose_side_a_tl
23 \tl_gset_eq:NN \g__penrose_side_c_tl \g__penrose_side_a_tl
24 \tl_gset_eq:NN \g__penrose_side_d_tl \g__penrose_side_a_tl
25 \tl_gset_eq:NN \g__penrose_side_e_tl \g__penrose_side_a_tl
26 \tl_gset_eq:NN \g__penrose_side_A_tl \g__penrose_side_a_tl
27 \tl_gset_eq:NN \g__penrose_side_B_tl \g__penrose_side_a_tl
28 \tl_gset_eq:NN \g__penrose_side_C_tl \g__penrose_side_a_tl
29 \tl_gset_eq:NN \g__penrose_side_D_tl \g__penrose_side_a_tl
30 \tl_gset_eq:NN \g__penrose_side_E_tl \g__penrose_side_a_tl

```

We need a few temporary variables to hold intermediate calculations.

```

\l__penrose_tmpa_fp
\l__penrose_tmpb_fp
\l__penrose_tmpe_fp
\l__penrose_tmpe_tl
\l__penrose_tmpe_tl
\l__penrose_tmpe_tl
\l__penrose_tmpe_tl
\l__penrose_tmpe_tl
\l__penrose_tmpe_int
\l__penrose_tmpe_int
\l__penrose_xa_dim
\l__penrose_ya_dim
\l__penrose_xb_dim
\l__penrose_yb_dim
\l__penrose_tmpe_prop
31 \fp_new:N \l__penrose_tmpe_fp
32 \fp_new:N \l__penrose_tmpe_fp
33 \fp_new:N \l__penrose_tmpe_fp
34 \str_new:N \l__penrose_tmpe_str
35 \str_new:N \l__penrose_tmpe_str
36 \tl_new:N \l__penrose_tmpe_tl
37 \tl_new:N \l__penrose_tmpe_tl
38 \tl_new:N \l__penrose_tmpe_tl
39 \tl_new:N \l__penrose_tmpe_tl
40 \tl_new:N \l__penrose_tmpe_tile_path_tl
41 \tl_new:N \g__penrose_smuggle_tl
42 \int_new:N \l__penrose_tmpe_int
43 \int_new:N \l__penrose_tmpe_int
44 \dim_new:N \l__penrose_xa_dim
45 \dim_new:N \l__penrose_ya_dim
46 \dim_new:N \l__penrose_xb_dim
47 \dim_new:N \l__penrose_yb_dim
48 \prop_new:N \l__penrose_tmpe_prop

```

(End definition for \l__penrose_tmpe_fp and others.)

2.2 Creating the Tiles

`__penrose_normalise_path:Nn` When defining the path for a side, we normalise so that it starts at the origin and ends at (1pt,0pt).

```

49 \cs_new_nopar:Npn \__penrose_normalise_path:Nn #1#2
50 {

```

Get the initial point of the path and convert to floating point.

```

51 \group_begin:
52 \spath_initialpoint:Nn \l__penrose_tmpa_tl {#2}
53 \fp_set:Nn \l__penrose_tmpa_fp {\tl_head:N \l__penrose_tmpa_tl}
54 \tl_set:Nx \l__penrose_tmpa_tl {\tl_tail:N \l__penrose_tmpa_tl}
55 \fp_set:Nn \l__penrose_tmpb_fp {\tl_head:N \l__penrose_tmpa_tl}

```

Get the final point of the path, and compute the difference of the final and initial points.

The resulting numbers, say a and b , will be put into a matrix to rotate and scale the path. The formula for the matrix is:

$$\frac{1}{a^2 + b^2} \begin{bmatrix} a & b \\ -b & a \end{bmatrix}$$

```

56 \spath_finalpoint:Nn \l__penrose_tmpa_tl {#2}
57 \fp_set:Nn \l__penrose_tmpa_fp
58 {\tl_head:N \l__penrose_tmpa_tl - \l__penrose_tmpa_fp}
59 \tl_set:Nx \l__penrose_tmpa_tl {\tl_tail:N \l__penrose_tmpa_tl}
60 \fp_set:Nn \l__penrose_tmpb_fp
61 {\tl_head:N \l__penrose_tmpa_tl - \l__penrose_tmpb_fp}

```

Now compute the square of the length of the path for scaling.

```

62 \fp_set:Nn \l__penrose_tmpc_fp
63 {\l__penrose_tmpa_fp^2 + \l__penrose_tmpb_fp^2}
64 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_tmpa_fp/\l__penrose_tmpc_fp}
65 \fp_set:Nn \l__penrose_tmpb_fp {\l__penrose_tmpb_fp/\l__penrose_tmpc_fp}
66 \fp_set:Nn \l__penrose_tmpe_fp {-\l__penrose_tmpb_fp}

```

Now construct the matrix.

```

67 \tl_set:Nx \l__penrose_tmpe_tl
68 {
69   {\fp_use:N \l__penrose_tmpa_fp}
70   {\fp_use:N \l__penrose_tmpe_fp} % swapped
71   {\fp_use:N \l__penrose_tmpe_fp} % swapped
72   {\fp_use:N \l__penrose_tmpe_fp}
73 }

```

Get the initial point back again for the translation part.

```

74 \spath_initialpoint:Nn \l__penrose_tmpa_tl {#2}

```

But we need to premultiply by the matrix because of how the transformations are applied.

```

75 \fp_set:Nn \l__penrose_tmpe_fp
76 {
77   (-1) * \l__penrose_tmpe_fp * \tl_head:N \l__penrose_tmpe_tl
78   + (-1) * \l__penrose_tmpe_fp * \tl_tail:N \l__penrose_tmpe_tl
79 }
80 \fp_set:Nn \l__penrose_tmpe_fp
81 {
82   (-1) * \l__penrose_tmpe_fp * \tl_tail:N \l__penrose_tmpe_tl
83   + \l__penrose_tmpe_fp * \tl_head:N \l__penrose_tmpe_tl
84 }

```

Finally, we apply the transformation to the path.

```

85 \tl_put_right:Nx \l__penrose_tmpb_tl {
86   {\fp_to_dim:N \l__penrose_tmpa_fp}
87   {\fp_to_dim:N \l__penrose_tmpb_fp}
88 }
89 \spath_transform:NnV \l__penrose_tmpa_tl {#2} \l__penrose_tmpb_tl
90 \tl_gset_eq:NN \g__penrose_smuggle_tl \l__penrose_tmpa_tl
91 \group_end:
92 \tl_set_eq:NN #1 \g__penrose_smuggle_tl
93 \tl_gclear:N \g__penrose_smuggle_tl
94 }
95 \cs_generate_variant:Nn \__penrose_normalise_path:Nn {NV, cn, cV}
96 \cs_new_protected_nopar:Npn \__penrose_normalise_path:N #1
97 {
98   \__penrose_normalise_path:NV #1#1
99 }
100 \cs_generate_variant:Nn \__penrose_normalise_path:N {c}

```

(End definition for __penrose_normalise_path:Nn.)

\SetPenrosePath

This sets the path corresponding to a particular side to the current path, and normalises it.

```

101 \NewDocumentCommand \SetPenrosePath { m }
102 {
103   \pgfsyssoftpath@getcurrentpath\l__penrose_tmpa_tl
104   \__penrose_normalise_path:N \l__penrose_tmpa_tl
105   \tl_gset_eq:cN {g__penrose_side_#1_tl} \l__penrose_tmpa_tl
106 }

```

`\tikz_scan_point:n` This is a wrapper around `\tikz@scan@one@point` to make it easier to use with L^AT_EX3 variables.

```

107 \cs_new_nopar:Npn \tikz_scan_point:n #1
108 {
109   \tikz@scan@one@point\pgfutil@firstofone#1\relax
110 }
111 \cs_generate_variant:Nn \tikz_scan_point:n {V}

```

(End definition for \tikz_scan_point:n.)

`__penrose_make_tile:nnn` This builds the tile path from its pieces. The arguments are the name of the tile, the descriptions of the sides, and a token list of the coordinates.

```

112 \cs_new_nopar:Npn \__penrose_make_tile:nnn #1#2#3
113 {

```

Get the first coordinate and initialise the path with a move to this point.

```

114 \group_begin:
115 \tl_set:Nn \l__penrose_tmpa_tl {#3}
116 \tl_set:Nx \l__penrose_tmpb_tl {\tl_head:N \l__penrose_tmpa_tl}
117 \tl_set:Nn \l__penrose_tmpa_tl {\pgfsyssoftpath@movetotoken}
118 \tikz_scan_point:V \l__penrose_tmpb_tl
119 \tl_put_right:Nx \l__penrose_tmpa_tl
120 {

```

```

121     {\dim_use:N \pgf@x}{\dim_use:N \pgf@y}
122   }
123   \tl_set_eq:NN \l__penrose_tmp_tile_path_tl \l__penrose_tmpa_tl

```

Now we have our path initialised, we can start appending the side paths according to the specification in the second argument.

We append the initial coordinate to the end of the list to make a closed cycle.

```

124   \tl_set:Nn \l__penrose_tmpa_tl {#3}
125   \tl_put_right:Nx \l__penrose_tmpa_tl {{\tl_head:N \l__penrose_tmpa_tl}}

```

Now we walk through the description of the sides, adding the specified paths to our tile path.

```

126   \tl_map_inline:nn {#2} {

```

Clone the path for this side.

```

127     \tl_set:Nx \l__penrose_tmpc_tl {\tl_head:n {##1}}
128     \tl_set_eq:Nc \l__penrose_tmpd_tl {g_penrose_side_ \tl_use:N \l__penrose_tmpc_tl _tl}

```

Strip off the next coordinate, and convert it to a point.

```

129     \tl_set:Nx \l__penrose_tmpb_tl {\tl_head:N \l__penrose_tmpa_tl}
130     \tl_set:Nx \l__penrose_tmpa_tl {\tl_tail:N \l__penrose_tmpa_tl}
131     \tikz_scan_point:V \l__penrose_tmpb_tl

```

Store the resulting coordinate.

```

132     \fp_set:Nn \l__penrose_tmpa_fp { \pgf@x }
133     \fp_set:Nn \l__penrose_tmpb_fp { \pgf@y }

```

Now get the next coordinate.

```

134     \tl_set:Nx \l__penrose_tmpb_tl {\tl_head:N \l__penrose_tmpa_tl}
135     \tikz_scan_point:V \l__penrose_tmpb_tl

```

We want the difference between the two coordinates.

```

136     \fp_set:Nn \l__penrose_tmpa_fp {\pgf@x - \l__penrose_tmpa_fp}
137     \fp_set:Nn \l__penrose_tmpb_fp {\pgf@y - \l__penrose_tmpb_fp}

```

This is converted into a transformation matrix.

```

138     \fp_set:Nn \l__penrose_tmpc_fp {-\l__penrose_tmpb_fp}
139     \tl_set:Nx \l__penrose_tmpb_tl
140     {
141       {\fp_use:N \l__penrose_tmpa_fp}
142       {\fp_use:N \l__penrose_tmpb_fp} % not swapped
143       {\fp_use:N \l__penrose_tmpc_fp} % not swapped
144       {\fp_use:N \l__penrose_tmpa_fp}
145       {0}
146       {0}
147     }

```

The transformation is applied to the cloned path.

```

148     \spath_transform:NV \l__penrose_tmpd_tl \l__penrose_tmpb_tl

```

And this is welded to the tile path.

```

149     \spath_weld:NV \l__penrose_tmp_tile_path_tl \l__penrose_tmpd_tl
150   }

```

At the end we close the path.

```

151 \spath_close:N \l__penrose_tmp_tile_path_tl
152 \tl_gset_eq:NN \g__penrose_smuggle_tl \l__penrose_tmp_tile_path_tl
153 \group_end:
154 \tl_gclear_new:c {g__penrose_tile_#1_tl}
155 \tl_gset_eq:cN {g__penrose_tile_#1_tl} \g__penrose_smuggle_tl
156 \tl_gclear:N \g__penrose_smuggle_tl
157 }

```

(End definition for `__penrose_make_tile:nnn`.)

`__penrose_make_tile:nn` A wrapper around the above which allows us to specify the second two arguments as two items in a token list.

```

158 \cs_new_nopar:Npn \__penrose_make_tile:nn #1#2
159 {
160   \__penrose_make_tile:nnn {#1} #2
161 }
162 \cs_generate_variant:Nn \__penrose_make_tile:nn {nV}

```

(End definition for `__penrose_make_tile:nn`.)

2.3 Specifying the Tiles

The tile specifications are contained in a prop.

```

163 \prop_new:N \g__penrose_tiles_prop

```

`__penrose_add_coordinate:Nnn`

Process a coordinate through `fp` and adds it to a token list.

`__penrose_add_coordinate:w`

```

164 \cs_new_nopar:Npn \__penrose_add_coordinate:Nnn #1#2#3 {
165   \fp_set:Nn \l__penrose_tmpa_fp{#2}
166   \fp_set:Nn \l__penrose_tmpb_fp{#3}
167   \tl_put_right:Nx #1
168   {
169     {(\fp_use:N \l__penrose_tmpa_fp, \fp_use:N \l__penrose_tmpb_fp)}
170   }
171 }

```

Wrapper around the add coordinate command to split at a comma.

```

172 \cs_new_nopar:Npn \__penrose_add_coordinate:w #1#2,#3 \q_stop
173 {
174   \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {#2}{#3}
175 }

```

(End definition for `__penrose_add_coordinate:Nnn` and `__penrose_add_coordinate:w`.)

Now we specify the tiles. The specification is a clockwise list of the vertices together with the labels of the corresponding sides. There are three basic paths, a, b, c, and their complements (which are capitalised).

- Thin Rhombus.

```

176 \tl_clear:N \l__penrose_tmpa_tl
177 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
178 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(18)}{sind(18)}
179 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {2*cosd(18)}{0}
180 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(18)}{-sind(18)}

```

```

181
182 \prop_gput:Nnx \g__penrose_tiles_prop {thin~ rhombus}
183 {{a A B b} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Thick Rhombus.

```

184 \tl_clear:N \l__penrose_tmpa_tl
185 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
186 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(36)}{sind(36)}
187 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {2*cosd(36)}{0}
188 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(36)}{-sind(36)}
189
190 \prop_gput:Nnx \g__penrose_tiles_prop {thick~ rhombus}
191 {{B a A b} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Dart.

```

192 \tl_clear:N \l__penrose_tmpa_tl
193 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
194 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl
195 {2*sind(18)*cosd(108)}{2*sind(18)*sind(108)}
196 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {2*sind(18)}{0}
197 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl
198 {2*sind(18)*cosd(108)}{-2*sind(18)*sind(108)}
199
200 \prop_gput:Nnx \g__penrose_tiles_prop {dart}
201 {{c a A C} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Kite.

```

202 \tl_clear:N \l__penrose_tmpa_tl
203 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
204 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(36)}{sind(36)}
205 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {1}{0}
206 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(36)}{-sind(36)}
207
208 \prop_gput:Nnx \g__penrose_tiles_prop {kite}
209 {{a c C A} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Golden Triangle.

```

210 \tl_clear:N \l__penrose_tmpa_tl
211 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
212 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(18)}{sind(18)}
213 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(18)}{-sind(18)}
214
215 \prop_gput:Nnx \g__penrose_tiles_prop {golden~ triangle}
216 {{a c b} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Reverse Golden Triangle.

```

217 \tl_clear:N \l__penrose_tmpa_tl
218 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}

```

```

219 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(18)}{sind(18)}
220 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(18)}{-sind(18)}
221
222 \prop_gput:Nnx \g__penrose_tiles_prop {reverse~ golden~ triangle}
223 {{B C A} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Golden Gnomon

```

224 \tl_clear:N \l__penrose_tmpa_tl
225 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
226 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(36)}{sind(36)}
227 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {2*cosd(36)}{0}
228
229 \prop_gput:Nnx \g__penrose_tiles_prop {golden~ gnomon}
230 {{C b A} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Reverse Golden Gnomon

```

231 \tl_clear:N \l__penrose_tmpa_tl
232 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
233 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {2*cosd(36)}{0}
234 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(36)}{-sind(36)}
235 \prop_gput:Nnx \g__penrose_tiles_prop {reverse~ golden~ gnomon}
236 {{a B c} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Primary Pentagon (pentagon 5)

```

237 \tl_clear:N \l__penrose_tmpa_tl
238 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
239 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(108)}{sind(108)}
240 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl
241 {1+cosd(72)+cosd(144)}{sind(72)+sind(144)}
242 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {1+cosd(72)}{sind(72)}
243 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {1}{0}
244 \prop_gput:Nnx \g__penrose_tiles_prop {pentagon~ 5}
245 {{a a a a a} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Secondary Pentagon (pentagon 3)

```

246 \prop_gput:Nnx \g__penrose_tiles_prop {pentagon~ 3}
247 {{A b a a b} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Tertiary Pentagon (pentagon 2)

```

248 \prop_gput:Nnx \g__penrose_tiles_prop {pentagon~ 2}
249 {{d A e c A} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Pentagram

```

250 \tl_clear:N \l__penrose_tmpa_tl
251 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {1}{0}
252 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {1-cosd(36)}{-sind(36)}
253 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl

```



```

254 {1-cosd(36)-cosd(108)}{-sind(36)-sind(108)}
255 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(108)}{-sind(108)}
256 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl
257 {-1+3*cosd(108)+cosd(36)}{-sind(36)-sind(108)}
258 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl
259 {-1+2*cosd(108)+cosd(36)}{-sind(36)}
260 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {-1+2*cosd(108)}{0}
261 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {2*cosd(108)}{0}
262 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(108)}{sind(108)}
263 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
264 \prop_gput:Nnx \g__penrose_tiles_prop {pentagram}
265 {{C E C E C E C E C E} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Boat

```

266 \tl_clear:N \l__penrose_tmpa_tl
267 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {-1+2*cosd(108)}{0}
268 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {2*cosd(108)}{0}
269 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(108)}{sind(108)}
270 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
271 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {1}{0}
272 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {1-cosd(36)}{-sind(36)}
273 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl
274 {-1+2*cosd(108)+cosd(36)}{-sind(36)}
275 \prop_gput:Nnx \g__penrose_tiles_prop {boat}
276 {{C E C E B D B} {\tl_use:N \l__penrose_tmpa_tl}}

```

- Diamond.

```

277 \tl_clear:N \l__penrose_tmpa_tl
278 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {0}{0}
279 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(18)}{sind(18)}
280 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {2*cosd(18)}{0}
281 \__penrose_add_coordinate:Nnn \l__penrose_tmpa_tl {cosd(18)}{-sind(18)}
282 \prop_gput:Nnx \g__penrose_tiles_prop {diamond}
283 {{D B B D} {\tl_use:N \l__penrose_tmpa_tl}}

```

\BakePenroseTile

This is the user wrapper around the tile creation macros.

```

284 \NewDocumentCommand \BakePenroseTile {m}
285 {
286   \prop_get:NnN \g__penrose_tiles_prop {#1} \l__penrose_tmpa_tl
287   \__penrose_make_tile:nV {#1} \l__penrose_tmpa_tl
288 }

```

For backwards compatibility.

```

289 \cs_set_eq:NN \MakePenroseTile \BakePenroseTile

```

\UsePenroseTile

This is the command that actually places a tile on the page. The first argument is optional and is for styling.

```
290 \NewDocumentCommand \UsePenroseTile {0{ } m}
291 {
```

We need to transform the tile to correspond to the current transformation matrix. To ensure that we only transform the current tile, we clone it first.

```
292 \tl_set_eq:Nc \l__penrose_tmp_tile_path_tl {g__penrose_tile_#2_tl}
```

The transformation matrix returned by PGF appears to be transposed from what it should be. (This needs a little more investigation, it might be that I've implemented the multiplication incorrectly here.)

```
293 \pgfgettransform \l__penrose_tmpa_tl
294 % \tl_clear:N \l__penrose_tmpb_tl
295 % \tl_set:Nx \l__penrose_tmpb_tl {{\tl_head:N \l__penrose_tmpa_tl}}
296 % \tl_set:Nx \l__penrose_tmpa_tl {\tl_tail:N \l__penrose_tmpa_tl}
297 % \tl_put_right:Nx \l__penrose_tmpb_tl
298 % {
299 %   {\tl_item:Nn \l__penrose_tmpa_tl {1}}
300 % }
301 % \tl_put_right:Nx \l__penrose_tmpb_tl
302 % {
303 %   {\tl_item:Nn \l__penrose_tmpa_tl {2}}
304 % }
305 % \tl_set:Nx \l__penrose_tmpa_tl {\tl_tail:N \l__penrose_tmpa_tl}
306 % \tl_set:Nx \l__penrose_tmpa_tl {\tl_tail:N \l__penrose_tmpa_tl}
307 % \tl_put_right:NV \l__penrose_tmpb_tl \l__penrose_tmpa_tl
```

Apply the transformation, protocol the path, and render it.

```
308 \spath_transform:NV \l__penrose_tmp_tile_path_tl \l__penrose_tmpa_tl
309 \spath_tikz_path:nV {#1} \l__penrose_tmp_tile_path_tl
310 }
```

save_Penrose_path

This is a style for a user to take a path and make it into the path for one of the sides. It needs to store both that side and the reverse.

```
311 \tikzset{
312   save~ Penrose~ path/.code={
313     \tikz@addmode{
```

Get the current path.

```
314     \pgfsyssoftpath@getcurrentpath\l__penrose_tmpa_tl
```

Normalise the path and save.

```
315     \__penrose_normalise_path:N \l__penrose_tmpa_tl
316     \tl_gclear_new:c {g__penrose_side_#1_tl}
317     \tl_gset_eq:cN {g__penrose_side_#1_tl} \l__penrose_tmpa_tl
```

Now create the reverse path. The name is the upper case version.

```
318     \tl_set:Nx \l__penrose_tmpb_tl {\str_uppercase:n {#1}}
```

Reverse the path, and relocate to the interval $[0, 1]$.

```
319     \spath_reverse:N \l__penrose_tmpa_tl
320     \spath_transform:Nnnnnn \l__penrose_tmpa_tl {-1} {0} {0} {-1} {1} {0}
321     \tl_gclear_new:c {g__penrose_side_ \tl_use:N \l__penrose_tmpb_tl _tl}
```

```

322     \tl_gset_eq:cN {g__penrose_side_ \tl_use:N \l__penrose_tmpb_tl _tl} \l__penrose_tmpa_tl
323   }
324 },
325 clone~ Penrose~ side~ path/.style~ 2~ args={
326   spath/set~ name=Penrose~side,
327   spath/clone~ global={#1}{#2}
328 },
329 spath/prefix/Penrose~side/.style={
330   spath/set~ prefix=g__penrose_side_,
331 },
332 spath/suffix/Penrose~side/.style={
333   spath/set~ suffix=_tl,
334 },
335 clone~ Penrose~ tile~ path/.style~ 2~ args={
336   spath/set~ name=Penrose~tile,
337   spath/clone~ global={#1}{#2}
338 },
339 spath/prefix/Penrose~tile/.style={
340   spath/set~ prefix=g__penrose_tile_,
341 },
342 spath/suffix/Penrose~tile/.style={
343   spath/set~ suffix=_tl,
344 },
345 expand~ key/.code={
346   \exp_args:NV \pgfkeysalso #1
347 }
348 }

```

(End definition for save Penrose path. This function is documented on page ??.)

Create the basic tile shapes.

```

349 \BakePenroseTile {thin~ rhombus}
350 \BakePenroseTile {thick~ rhombus}
351 \BakePenroseTile {dart}
352 \BakePenroseTile {kite}
353 \BakePenroseTile {golden~ triangle}
354 \BakePenroseTile {reverse~ golden~ triangle}
355 \BakePenroseTile {golden~ gnomon}
356 \BakePenroseTile {reverse~ golden~ gnomon}
357 \BakePenroseTile {pentagon~ 5}
358 \BakePenroseTile {pentagon~ 3}
359 \BakePenroseTile {pentagon~ 2}
360 \BakePenroseTile {pentagram}
361 \BakePenroseTile {boat}
362 \BakePenroseTile {diamond}

```

2.4 User-Definable Tiles

The commands in this section are for enabling a user to define their own tile.

`__penrose_transform_side:nn` Apply a transformation to make a given side lie on the x-axis. First argument is the tile, second is the side.

```

363 \cs_new_nopar:Npn \__penrose_transform_side:nn #1#2
364 {

```

Get our tile data.

```
365 \prop_get:NnN \g__penrose_tiles_prop {#1} \l__penrose_tmpa_tl
```

Start with the edge list.

Initialise the counter.

```
366 \int_zero:N \l__penrose_tmpb_int
```

```
367 \int_incr:N \l__penrose_tmpb_int
```

Get the path type list.

```
368 \tl_set:Nx \l__penrose_tmpc_tl {\tl_head:N \l__penrose_tmpa_tl}
```

Iterate through the path type list, looking for the requested path.

```
369 \tl_map_inline:Nn \l__penrose_tmpc_tl {
```

```
370 \str_if_eq:nnT {#1} {#2} {
```

```
371 \tl_map_break:
```

```
372 }
```

```
373 \int_incr:N \l__penrose_tmpb_int
```

```
374 }
```

Get the coordinate list.

```
375 \tl_set:Nx \l__penrose_tmpc_tl {\tl_tail:N \l__penrose_tmpa_tl}
```

Strip off the outer braces.

```
376 \tl_set:Nx \l__penrose_tmpc_tl {\tl_item:Nn \l__penrose_tmpc_tl {1}}
```

Add the first coordinate at the end.

```
377 \tl_put_right:Nx \l__penrose_tmpc_tl
```

```
378 {\tl_item:Nn \l__penrose_tmpc_tl {1}}
```

Get the coordinates for this edge.

```
379 \tl_set:Nx \l__penrose_tmpa_tl
```

```
380 {\tl_item:Nn \l__penrose_tmpc_tl {\int_use:N \l__penrose_tmpb_int}}
```

```
381 \tl_set:Nx \l__penrose_tmpb_tl
```

```
382 {\tl_item:Nn \l__penrose_tmpc_tl {\int_use:N \l__penrose_tmpb_int + 1}}
```

Translate second to origin.

```
383 \tikz_scan_point:V \l__penrose_tmpb_tl
```

```
384 \dim_set_eq:Nc \l__penrose_xa_dim {pgf@x}
```

```
385 \dim_set_eq:Nc \l__penrose_ya_dim {pgf@y}
```

Rotate and scale first to unit x-vector.

```
386 \tikz_scan_point:V \l__penrose_tmpa_tl
```

```
387 \dim_set_eq:Nc \l__penrose_xb_dim {pgf@x}
```

```
388 \dim_set_eq:Nc \l__penrose_yb_dim {pgf@y}
```

```
389
```

```
390 \dim_sub:Nn \l__penrose_xb_dim {\l__penrose_xa_dim }
```

```
391 \dim_sub:Nn \l__penrose_yb_dim {\l__penrose_ya_dim }
```

And normalise the vector along it.

```
392 \pgfpointnormalised{\pgfpoint{\l__penrose_xb_dim}{\l__penrose_yb_dim}}
```

```
393 \dim_set_eq:Nc \l__penrose_xb_dim {pgf@x}
```

```
394 \dim_set_eq:Nc \l__penrose_yb_dim {pgf@y}
```

Now rotate so that the x -axis lies along the edge.

```

395   \pgftransformtriangle%
396   {\pgfpoint{0pt}{0pt}}%
397   {\pgfpoint{\l__penrose_xb_dim}{-\l__penrose_yb_dim}}
398   {\pgfpoint{\l__penrose_yb_dim}{\l__penrose_xb_dim}}
399
400   \pgftransformshift{\pgfpoint{-\l__penrose_xa_dim}{-\l__penrose_ya_dim}}
401
402 }

```

(End definition for `__penrose_transform_side:nn`.)

```

403 \cs_generate_variant:Nn \tl_if_eq:nnT {nVT}
404 \cs_generate_variant:Nn \__penrose_transform_side:nn {nx,nV}

```

`\TransformAlongSide` Make this available outside the L^AT_EX3 environment.

```

405 \DeclareDocumentCommand \TransformAlongSide {m m}
406 {
407   \__penrose_transform_side:nx {#1}{#2}
408 }

```

(End definition for `\TransformAlongSide`.)

`__penrose_coordinates_at_vertices:n` This places TikZ coordinates at the vertices of the tile.

```

409 \cs_new_nopar:Npn \__penrose_coordinates_at_vertices:n #1
410 {

```

Get our tile data

```

411   \prop_get:NnN \g__penrose_tiles_prop {#1} \l__penrose_tmpa_tl

```

Start with the edge list

```

412   \tl_set:Nx \l__penrose_tmpb_tl {\tl_head:N \l__penrose_tmpa_tl}

```

Get the coordinate list

```

413   \tl_set:Nx \l__penrose_tmpc_tl {\tl_tail:N \l__penrose_tmpa_tl}

```

Strip off the outer braces

```

414   \tl_set:Nx \l__penrose_tmpc_tl
415   {\tl_item:Nn \l__penrose_tmpc_tl {1}}

```

Add the first coordinate at the end

```

416   \tl_put_right:Nx \l__penrose_tmpc_tl
417   {\tl_item:Nn \l__penrose_tmpc_tl {1}}

```

Get the first coordinate

```

418   \tl_set:Nx \l__penrose_tmpa_tl {\tl_head:N \l__penrose_tmpc_tl}
419   \tl_set:Nx \l__penrose_tmpc_tl {\tl_tail:N \l__penrose_tmpc_tl}

```

Iterate through the path type list, looking for the requested path

```

420   \tl_map_inline:Nn \l__penrose_tmpb_tl {
421     \tl_set:Nx \l__penrose_tmpd_tl {
422       \exp_not:N \coordinate
423       (-edge~ ##1~ start)~ at \tl_use:N \l__penrose_tmpa_tl;
424     }
425     \tl_use:N \l__penrose_tmpd_tl
426     \tl_set:Nx \l__penrose_tmpa_tl {\tl_head:N \l__penrose_tmpc_tl}
427     \tl_set:Nx \l__penrose_tmpc_tl {\tl_tail:N \l__penrose_tmpc_tl}
428     \tl_set:Nx \l__penrose_tmpd_tl {

```

```

429     \exp_not:N \coordinate
430     (-edge~ ##1~ end)~ at \tl_use:N \l__penrose_tmpa_tl;
431   }
432   \tl_use:N \l__penrose_tmpd_tl
433 }
434 }

```

(End definition for __penrose_coordinates_at_vertices:n.)

`\CoordinatesAtVertices` User-accessible wrapper around the above.

```

435 \DeclareDocumentCommand \CoordinatesAtVertices {m}
436 {
437   \__penrose_coordinates_at_vertices:n {#1}
438 }

```

(End definition for \CoordinatesAtVertices.)

\DefineTile

This is the user function for defining a tile.

```
439 \DeclareDocumentCommand \DefineTile { m m m }
440 {
```

Clear the temporary variable.

```
441 \tl_clear:N \l__penrose_tmpa_tl
```

The 3rd parameter is a list of coordinates at vertices, iterate through them and add them to the list.

```
442 \tl_map_inline:nn {#3} {
443   \__penrose_add_coordinate:w \l__penrose_tmpa_tl ##1 \q_stop
444 }
```

Now we make a list of the edge types (from the 2nd parameter), using a prop to keep track of whether an edge is repeated.

```
445 \prop_clear:N \l__penrose_tmpa_prop
446 \tl_map_inline:nn {#2} {
447   \prop_if_in:NnTF \l__penrose_tmpa_prop {##1}
448   {
449     \prop_put:Nnn \l__penrose_tmpa_prop {##1} {1}
450   }
451   {
452     \prop_put:Nnn \l__penrose_tmpa_prop {##1} {0}
453   }
454 }
```

Having established their multiplicity, we now create the edges with their names, appending numbers to their names if used more than once.

```
455 \tl_clear:N \l__penrose_tmpb_tl
456 \tl_map_inline:nn {#2}
457 {
458   \tl_clear:N \l__penrose_tmpc_tl
459   \tl_put_right:Nn \l__penrose_tmpc_tl {##1}
460   \int_compare:nF {\prop_item:Nn \l__penrose_tmpa_prop {##1} == 0} {
461     \tl_put_right:Nx \l__penrose_tmpc_tl
462     {\prop_item:Nn \l__penrose_tmpa_prop {##1}}
463     \prop_put:Nnx \l__penrose_tmpa_prop {##1}
464     {\int_eval:n {\prop_item:Nn \l__penrose_tmpa_prop {##1} + 1}}
465   }
466   \tl_put_right:Nx \l__penrose_tmpb_tl {{ \l__penrose_tmpc_tl }}
467 }
```

Finally, we can create our tile and add it to the global tile prop.

```
468 \prop_gput:Nnx \g__penrose_tiles_prop {#1}
469 {{\tl_use:N \l__penrose_tmpb_tl} {\tl_use:N \l__penrose_tmpa_tl}}
```

Having created the tile, we make a TikZ pic to place it on the page.

```
470 \tikzset{
471   #1/.pic={
472     \begin{scope}
```

Were we given a tile to align ourselves against?

```
473     \pgfkeysgetvalue{/tikz/Penrose/alignment~ location}{\prloc}
474     \ifx\prloc\pgfutil@empty
475     \else
```

Yes, we were. So we adjust our position accordingly. The first job is to transform so that we're along the edge of the receiving tile. 15

```
476     \begingroup
```

We get the locations of the start and end of the receiving tile. As pic sets the node prefix, we have to temporarily suspend that (hence working in a group).

```
477     \tikzset{name~ prefix~ ..}%
478     \tikz@scan@one@point\pgfutil@firstofone%
479     (\prloc~edge~ \pgfkeysvalueof{/tikz/Penrose/alignment~ edge}~ start)%
480     \global\pgf@xa=\pgf@x
481     \global\pgf@ya=\pgf@y
```

2.5 Lindenmayer System

This is an implementation of the Lindenmayer System description of Penrose tilings as a way of generating tilings from a specific starting seed.

The implementation uses `props` to store *rules* and *actions*. The rules are used to expand the starting seed to a certain level, after which the actions are carried out. The syntax is based on the PGF library, but as we're already using L^AT_EX3 it is reimplemented in that.

These are the rules for generating rhombus tilings.

```

531 \prop_new:N \g__penrose_rhombus_lms_rule_prop
532 \prop_gput:Nnn \g__penrose_rhombus_lms_rule_prop {T} {[f*sT][f>g]}
533 \prop_gput:Nnn \g__penrose_rhombus_lms_rule_prop {t} {[f_st][f>G]}
534 \prop_gput:Nnn \g__penrose_rhombus_lms_rule_prop {G} {[f+sG][sf>g][sf*sT]}
535 \prop_gput:Nnn \g__penrose_rhombus_lms_rule_prop {g} {[f-sg][sf>G][sf_st]}

```

These are the rules for generating kite and dart tilings.

```

536 \prop_new:N \g__penrose_kite_lms_rule_prop
537 \prop_gput:Nnn \g__penrose_kite_lms_rule_prop {T} {[f*sT][f>st][+sg]}
538 \prop_gput:Nnn \g__penrose_kite_lms_rule_prop {t} {[f_st][f>st][-sg]}
539 \prop_gput:Nnn \g__penrose_kite_lms_rule_prop {G} {[f++sG][sT]}
540 \prop_gput:Nnn \g__penrose_kite_lms_rule_prop {g} {[f-sg][st]}

```

These are the rules for generating pentagon tilings.

```

541 \prop_new:N \g__penrose_pentagon_lms_rule_prop
542 \prop_gput:Nnn \g__penrose_pentagon_lms_rule_prop {P}
543 {[s>P][1sF+Q][1+sF+Q][1*sF+Q][1-sF+Q][1_sF+Q]} % pentagon 5
544 \prop_gput:Nnn \g__penrose_pentagon_lms_rule_prop {Q}
545 {[s>P][1+sFR][1*sF*R][1-sF+Q][1_sF+Q][1sF+Q][->fsD]} % pentagon 3
546 \prop_gput:Nnn \g__penrose_pentagon_lms_rule_prop {R}
547 {[s>P][1-sF+Q][1+sF*R][1*sFR][1_sF*R][1sFR][>fsD][>fsD]} % pentagon 2
548 \prop_gput:Nnn \g__penrose_pentagon_lms_rule_prop {G}
549 {
550   [s>G]
551   [se[>d+R][e1B]]
552   [+se[>d+R][e1B]]
553   [-se[>d+R][e1B]]
554   [*se[>d+R][e1B]]
555   [_se[>d+R][e1B]]
556 } % pentagram
557 \prop_gput:Nnn \g__penrose_pentagon_lms_rule_prop {B}
558 {
559   [s>G]
560   [se[>d+R][e1B]]
561   [+se[>d+R][e1B]]
562   [-se[>d+R][e1B]]
563 } % boat
564 \prop_gput:Nnn \g__penrose_pentagon_lms_rule_prop {D}
565 {[s>d+R][s>eG][se1B]} % diamond

```

Each of the standard tilings can also be drawn using triangles using the same rules.

```

566 \prop_gset_eq:NN \g__penrose_rtriangle_lms_rule_prop
567 \g__penrose_rhombus_lms_rule_prop
568 \prop_gset_eq:NN \g__penrose_ktriangle_lms_rule_prop
569 \g__penrose_kite_lms_rule_prop

```


These hold the various actions.

```

570 \prop_new:N \g__penrose_default_lms_action_prop
571 \prop_new:N \g__penrose_rhombus_lms_action_prop
572 \prop_new:N \g__penrose_kite_lms_action_prop
573 \prop_new:N \g__penrose_rtriangle_lms_action_prop
574 \prop_new:N \g__penrose_ktriangle_lms_action_prop
575 \prop_new:N \g__penrose_pentagon_lms_action_prop

```

We need some parameters.

```

576 \dim_new:N \l__penrose_step_dim
577 \dim_set:Nn \l__penrose_step_dim {1cm}

```

These are the defaults, which will be used in all the rule sets.

```

578 \prop_gput:Nnn \g__penrose_default_lms_action_prop {} {\group_begin:}
579 \prop_gput:Nnn \g__penrose_default_lms_action_prop {} {\group_end:}
580 \prop_gput:Nnn \g__penrose_default_lms_action_prop {f}
581 {\pgftransformxshift{\l__penrose_step_dim}}
582 \prop_gput:Nnn \g__penrose_default_lms_action_prop {s} {
583   \fp_set:Nn \l__penrose_tmpa_fp { 2 * sind(18) * \l__penrose_step_dim }
584   \dim_set:Nn \l__penrose_step_dim {\fp_to_dim:N \l__penrose_tmpa_fp}
585 }

```

The rhombus rules need a variety of turns.

```

586 \prop_gput:Nnn \g__penrose_rhombus_lms_action_prop {+}
587 {\pgftransformrotate{144}}
588 \prop_gput:Nnn \g__penrose_rhombus_lms_action_prop {*}
589 {\pgftransformrotate{108}}
590 \prop_gput:Nnn \g__penrose_rhombus_lms_action_prop {-}
591 {\pgftransformrotate{216}}
592 \prop_gput:Nnn \g__penrose_rhombus_lms_action_prop {_}
593 {\pgftransformrotate{252}}
594 \prop_gput:Nnn \g__penrose_rhombus_lms_action_prop {>}
595 {\pgftransformrotate{180}}

```

Up to now, the actions for the rhombus and its triangle replacement are the same.

```

596 \prop_gset_eq:NN \g__penrose_rtriangle_lms_action_prop
597 \g__penrose_rhombus_lms_action_prop

```

Now we do the actions that actually draw something.

```

598 \prop_gput:Nnn \g__penrose_rhombus_lms_action_prop {T} {
599   \group_begin:

```

As we go through, we keep track of how many tiles we've drawn.

```

600   \int_gincr:N \g__penrose_tile_int

```

Set up the position, size, and angle correctly.

```

601   \pgftransformrotate{198}
602   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim*2*cosd(18)}
603   \pgftransformxshift{-\fp_to_dim:N \l__penrose_tmpa_fp}
604   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
605   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}

```

Now we draw the thin rhombus, applying every style we can possibly imagine. The `Penrose tile` style gets the current tile and total tile numbers passed to it.

```

606   \tl_set:Nx \l__penrose_tmpc_tl
607   {
608     {\int_use:N \g__penrose_tile_int}

```

```

609   {\int_use:N \g__penrose_tiles_int}
610 }
611 \UsePenroseTile[
612   every~ Penrose~ tile/.try,
613   every~ thin~ rhombus/.try,
614   Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
615   Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
616 ]{thin~rhombus}
617 \group_end:
618 }

```

Same for the thick rhombus.

```

619 \prop_gput:Nnn \g__penrose_rhombus_lms_action_prop {G} {
620   \group_begin:
621   \int_gincr:N \g__penrose_tile_int
622   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)/(2*cosd(36))}
623   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
624   \tl_set:Nx \l__penrose_tmpc_tl
625   {
626     {\int_use:N \g__penrose_tile_int}
627     {\int_use:N \g__penrose_tiles_int}
628   }
629   \UsePenroseTile[
630     every~ Penrose~ tile/.try,
631     every~ thick~ rhombus/.try,
632     Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
633     Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
634   ]{thick~rhombus}
635   \group_end:
636 }

```

Now we do the same for the kite and dart tiling.

```

637 \prop_gput:Nnn \g__penrose_kite_lms_action_prop {+} {\pgftransformrotate{36}}
638 \prop_gput:Nnn \g__penrose_kite_lms_action_prop {*} {\pgftransformrotate{108}}
639 \prop_gput:Nnn \g__penrose_kite_lms_action_prop {-} {\pgftransformrotate{-36}}
640 \prop_gput:Nnn \g__penrose_kite_lms_action_prop {_}
641 {\pgftransformrotate{-108}}
642 \prop_gput:Nnn \g__penrose_kite_lms_action_prop {>} {\pgftransformrotate{180}}
643 \prop_gset_eq:NN \g__penrose_ktriangle_lms_action_prop
644 \g__penrose_kite_lms_action_prop
645 \prop_gput:Nnn \g__penrose_kite_lms_action_prop {T} {
646   \group_begin:
647   \int_gincr:N \g__penrose_tile_int
648   \pgftransformrotate{36}
649   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
650   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
651   \tl_set:Nx \l__penrose_tmpc_tl
652   {
653     {\int_use:N \g__penrose_tile_int}
654     {\int_use:N \g__penrose_tiles_int}
655   }
656   \UsePenroseTile[
657     every~ Penrose~ tile/.try,
658     every~ kite/.try,

```

```

659 Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
660 Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
661 ]{kite}
662 \group_end:
663 }
664 \prop_gput:Nnn \g__penrose_kite_lms_action_prop {g} {
665 \group_begin:
666 \int_gincr:N \g__penrose_tile_int
667 \pgftransformrotate{144}
668 \pgftransformxshift{-\l__penrose_step_dim * 2 * sin(18)}
669 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
670 \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
671 \tl_set:Nx \l__penrose_tmpc_tl
672 {
673 {\int_use:N \g__penrose_tile_int}
674 {\int_use:N \g__penrose_tiles_int}
675 }
676 \UsePenroseTile[
677 every~ Penrose~ tile/.try,
678 every~ dart/.try,
679 Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
680 Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
681 ]{dart}
682 \group_end:
683 }

```

Now we set up the actions for the triangle variations.

```

684 \prop_gput:Nnn \g__penrose_rtriangle_lms_action_prop {T} {
685 \group_begin:
686 \int_gincr:N \g__penrose_tile_int
687 \pgftransformrotate{18}
688 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
689 \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
690 \tl_set:Nx \l__penrose_tmpc_tl
691 {
692 {\int_use:N \g__penrose_tile_int}
693 {\int_use:N \g__penrose_tiles_int}
694 }
695 \UsePenroseTile[
696 every~ Penrose~ tile/.try,
697 every~ reverse~ golden~ triangle/.try,
698 Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
699 Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
700 ]{reverse~ golden~ triangle}
701 \group_end:
702 }
703 \prop_gput:Nnn \g__penrose_rtriangle_lms_action_prop {t} {
704 \group_begin:
705 \int_gincr:N \g__penrose_tile_int
706 \pgftransformrotate{-18}
707 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
708 \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
709 \tl_set:Nx \l__penrose_tmpc_tl
710 {

```

```

711     {\int_use:N \g__penrose_tile_int}
712     {\int_use:N \g__penrose_tiles_int}
713   }
714   \tl_set:Nx \l__penrose_tmpc_tl
715   {
716     {\int_use:N \g__penrose_tile_int}
717     {\int_use:N \g__penrose_tiles_int}
718   }
719   \UsePenroseTile[
720     every~ Penrose~ tile/.try,
721     every~ golden~ triangle/.try,
722     Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
723     Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
724   ]{golden~ triangle}
725   \group_end:
726 }

727 \prop_gput:Nnn \g__penrose_rtriangle_lms_action_prop {G} {
728   \group_begin:
729   \int_gincr:N \g__penrose_tile_int
730   \pgftransformrotate{180}
731   \pgftransformxshift{-\l__penrose_step_dim}
732   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)/(2*cosd(36))}
733   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
734   \tl_set:Nx \l__penrose_tmpc_tl
735   {
736     {\int_use:N \g__penrose_tile_int}
737     {\int_use:N \g__penrose_tiles_int}
738   }
739   \UsePenroseTile[
740     every~ Penrose~ tile/.try,
741     every~ reverse~ golden~ gnomon/.try,
742     Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
743     Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
744   ]{reverse~ golden~ gnomon}
745   \group_end:
746 }

747 \prop_gput:Nnn \g__penrose_rtriangle_lms_action_prop {g} {
748   \group_begin:
749   \int_gincr:N \g__penrose_tile_int
750   \pgftransformrotate{180}
751   \pgftransformxshift{-\l__penrose_step_dim}
752   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)/(2*cosd(36))}
753   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
754   \tl_set:Nx \l__penrose_tmpc_tl
755   {
756     {\int_use:N \g__penrose_tile_int}
757     {\int_use:N \g__penrose_tiles_int}
758   }
759   \UsePenroseTile[
760     every~ Penrose~ tile/.try,
761     every~ golden~ gnomon/.try,
762     Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
763     Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl

```

```

764   ]{golden~ gnomon}
765   \group_end:
766 }

767 \prop_gput:Nnn \g__penrose_ktriangle_lms_action_prop {T} {
768   \group_begin:
769   \int_gincr:N \g__penrose_tile_int
770   \pgftransformrotate{18}
771   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
772   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
773   \tl_set:Nx \l__penrose_tmpc_tl
774   {
775     {\int_use:N \g__penrose_tile_int}
776     {\int_use:N \g__penrose_tiles_int}
777   }
778   \UsePenroseTile[
779     every~ Penrose~ tile/.try,
780     every~ reverse~ golden~ triangle/.try,
781     Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
782     Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
783   ]{reverse~ golden~ triangle}
784   \group_end:
785 }

786 \prop_gput:Nnn \g__penrose_ktriangle_lms_action_prop {t} {
787   \group_begin:
788   \int_gincr:N \g__penrose_tile_int
789   \pgftransformrotate{-18}
790   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
791   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
792   \tl_set:Nx \l__penrose_tmpc_tl
793   {
794     {\int_use:N \g__penrose_tile_int}
795     {\int_use:N \g__penrose_tiles_int}
796   }
797   \UsePenroseTile[
798     every~ Penrose~ tile/.try,
799     every~ golden~ triangle/.try,
800     Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
801     Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
802   ]{golden~ triangle}
803   \group_end:
804 }

805 \prop_gput:Nnn \g__penrose_ktriangle_lms_action_prop {G} {
806   \group_begin:
807   \int_gincr:N \g__penrose_tile_int
808   \pgftransformrotate{180}
809   \pgftransformxshift{-\l__penrose_step_dim}
810   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)/(2*cosd(36))}
811   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
812   \tl_set:Nx \l__penrose_tmpc_tl
813   {
814     {\int_use:N \g__penrose_tile_int}
815     {\int_use:N \g__penrose_tiles_int}
816   }

```

```

817 \UsePenroseTile[
818   every~ Penrose~ tile/.try,
819   every~ reverse~ golden~ gnomon/.try,
820   Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
821   Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
822 ]{reverse~ golden~ gnomon}
823 \group_end:
824 }

825 \prop_gput:Nnn \g__penrose_ktriangle_lms_action_prop {g} {
826   \group_begin:
827   \int_gincr:N \g__penrose_tile_int
828   \pgftransformrotate{180}
829   \pgftransformxshift{-\l__penrose_step_dim}
830   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)/(2*cosd(36))}
831   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
832   \tl_set:Nx \l__penrose_tmpc_tl
833   {
834     {\int_use:N \g__penrose_tile_int}
835     {\int_use:N \g__penrose_tiles_int}
836   }
837   \UsePenroseTile[
838     every~ Penrose~ tile/.try,
839     every~ golden~ gnomon/.try,
840     Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
841     Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
842   ]{golden~ gnomon}
843   \group_end:
844 }

```

Now we do the same for the pentagonal tilings.

The rules need a variety of turns.

```

845 \int_new:N \l__penrose_pentagon_parity_int
846 \seq_new:N \l__penrose_pentagon_parity_seq
847 \seq_set_from_clist:Nn \l__penrose_pentagon_parity_seq {odd,even}
848 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {1} {
849   \int_set:Nn \l__penrose_pentagon_parity_int
850   {3 - \l__penrose_pentagon_parity_int}
851 }
852 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {+}
853 {\pgftransformrotate{72}}
854 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {*}
855 {\pgftransformrotate{144}}
856 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {-}
857 {\pgftransformrotate{288}}
858 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {_}
859 {\pgftransformrotate{216}}
860 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {>}
861 {\pgftransformrotate{180}}
862 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {|}
863 {\pgftransformxscale{-1}}

```

The scale factor is different.

```

864 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {s} {
865   \fp_set:Nn \l__penrose_tmpa_fp

```

```

866 {
867   1/(2 + 2 * cosd(72) ) * \l__penrose_step_dim
868 }
869 \dim_set:Nn \l__penrose_step_dim {\fp_to_dim:N \l__penrose_tmpa_fp}
870 }

```

And we tend to work better vertically.

```

871 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {f} {
872   \fp_set:Nn \l__penrose_tmpa_fp { tand(54)/2 * \l__penrose_step_dim }
873   \pgftransformyshift{\fp_to_dim:N \l__penrose_tmpa_fp}
874 }
875 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {F} {
876   \fp_set:Nn \l__penrose_tmpa_fp { tand(54) * \l__penrose_step_dim }
877   \pgftransformyshift{\fp_to_dim:N \l__penrose_tmpa_fp}
878 }
879 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {d} {
880   \fp_set:Nn \l__penrose_tmpa_fp
881   {
882     (tand(54)/2 - tand(72)/2 + sind(36) ) * \l__penrose_step_dim
883   }
884   \pgftransformyshift{\fp_to_dim:N \l__penrose_tmpa_fp}
885 }
886 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {e} {
887   \fp_set:Nn \l__penrose_tmpa_fp
888   {
889     tand(54) * cosd(36) * \l__penrose_step_dim
890   }
891   \pgftransformyshift{\fp_to_dim:N \l__penrose_tmpa_fp}
892 }
893 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {P} {
894   \group_begin:
895   \int_gincr:N \g__penrose_tile_int
896   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/2}
897   \pgftransformxshift{-\fp_to_dim:N \l__penrose_tmpa_fp}
898   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim*tand(54)/2}
899   \pgftransformyshift{-\fp_to_dim:N \l__penrose_tmpa_fp}
900   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
901   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
902   \tl_set:Nx \l__penrose_tmpe_tl
903   {
904     {\int_use:N \g__penrose_tile_int}
905     {\int_use:N \g__penrose_tmpe_int}
906   }
907   \UsePenroseTile[
908     every~ Penrose~ tile/.try,
909     every~ pentagon/.try,
910     every~
911     \seq_item:Nn \l__penrose_pentagon_parity_seq
912     {\l__penrose_pentagon_parity_int}
913     \space pentagon/.try,
914     every~ pentagon~ 5/.try,
915     Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
916     Penrose~ tile/.try/.expand~ once=\l__penrose_tmpe_tl
917   ]{pentagon~5}

```

```

918 \group_end:
919 }

920 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {Q} {
921 \group_begin:
922 \int_gincr:N \g__penrose_tile_int
923 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/2}
924 \pgftransformxshift{-\fp_to_dim:N \l__penrose_tmpa_fp}
925 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim*tand(54)/2}
926 \pgftransformyshift{-\fp_to_dim:N \l__penrose_tmpa_fp}
927 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
928 \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
929 \tl_set:Nx \l__penrose_tmpc_tl
930 {
931 {\int_use:N \g__penrose_tile_int}
932 {\int_use:N \g__penrose_tiles_int}
933 }
934 \UsePenroseTile[
935 every~ Penrose~ tile/.try,
936 every~ pentagon/.try,
937 every~
938 \seq_item:Nn \l__penrose_pentagon_parity_seq
939 {\l__penrose_pentagon_parity_int}
940 \space pentagon/.try,
941 every~ pentagon~ 3/.try,
942 Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
943 Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
944 ]{pentagon~3}
945 \group_end:
946 }

947 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {R} {
948 \group_begin:
949 \int_gincr:N \g__penrose_tile_int
950 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/2}
951 \pgftransformxshift{-\fp_to_dim:N \l__penrose_tmpa_fp}
952 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim*tand(54)/2}
953 \pgftransformyshift{-\fp_to_dim:N \l__penrose_tmpa_fp}
954 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
955 \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
956 \tl_set:Nx \l__penrose_tmpc_tl
957 {
958 {\int_use:N \g__penrose_tile_int}
959 {\int_use:N \g__penrose_tiles_int}
960 }
961 \UsePenroseTile[
962 every~ Penrose~ tile/.try,
963 every~ pentagon/.try,
964 every~
965 \seq_item:Nn \l__penrose_pentagon_parity_seq
966 {\l__penrose_pentagon_parity_int}
967 \space pentagon/.try,
968 every~ pentagon~ 2/.try,
969 Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
970 Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl

```



```

971   ]{pentagon~2}
972   \group_end:
973 }

974 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {G} {
975   \group_begin:
976   \int_gincr:N \g__penrose_tile_int
977   % \pgftransformrotate{198}
978   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim*cosd(72)}
979   \pgftransformxshift{\fp_to_dim:N \l__penrose_tmpa_fp}
980   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim*tand(54)*cosd(72)}
981   \pgftransformyshift{\fp_to_dim:N \l__penrose_tmpa_fp}
982   \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
983   \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
984   \tl_set:Nx \l__penrose_tmpc_tl
985   {
986     {\int_use:N \g__penrose_tile_int}
987     {\int_use:N \g__penrose_tiles_int}
988   }
989   \UsePenroseTile[
990     every~ Penrose~ tile/.try,
991     every~ pentagram/.try,
992     Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
993     Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
994   ]{pentagram}
995   \group_end:
996 }

997 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {B} {
998   \group_begin:
999   \int_gincr:N \g__penrose_tile_int
1000  % \pgftransformrotate{198}
1001  \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim*cosd(72)}
1002  \pgftransformxshift{\fp_to_dim:N \l__penrose_tmpa_fp}
1003  \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim*tand(54)*cosd(72)}
1004  \pgftransformyshift{\fp_to_dim:N \l__penrose_tmpa_fp}
1005  \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
1006  \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
1007  \tl_set:Nx \l__penrose_tmpc_tl
1008  {
1009    {\int_use:N \g__penrose_tile_int}
1010    {\int_use:N \g__penrose_tiles_int}
1011  }
1012  \UsePenroseTile[
1013    every~ Penrose~ tile/.try,
1014    every~ boat/.try,
1015    Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
1016    Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
1017  ]{boat}
1018  \group_end:
1019 }

1020 \prop_gput:Nnn \g__penrose_pentagon_lms_action_prop {D} {
1021   \group_begin:
1022   \int_gincr:N \g__penrose_tile_int
1023   \pgftransformrotate{90}

```

```

1024 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim*cosd(18)}
1025 \pgftransformxshift{-\fp_to_dim:N \l__penrose_tmpa_fp}
1026 \fp_set:Nn \l__penrose_tmpa_fp {\l__penrose_step_dim/(1cm)}
1027 \pgftransformscale{\fp_use:N \l__penrose_tmpa_fp}
1028 \tl_set:Nx \l__penrose_tmpc_tl
1029 {
1030   {\int_use:N \g__penrose_tile_int}
1031   {\int_use:N \g__penrose_tiles_int}
1032 }
1033 \UsePenroseTile[
1034   every~ Penrose~ tile/.try,
1035   every~ diamond/.try,
1036   Penrose~ tile~ \int_use:N \g__penrose_tile_int/.try,
1037   Penrose~ tile/.try/.expand~ once=\l__penrose_tmpc_tl
1038 ]{diamond}
1039 \group_end:
1040 }

```

`__penrose_make_lms:Nnnn` This creates the token list of actions, starting with the seed. The arguments are: a token list to store the result in, the name of the system, the number of iterations, and the initial state.

```

1041 \cs_new_nopar:Npn \__penrose_make_lms:Nnnn #1#2#3#4
1042 {
1043   \group_begin:

```

On the first time round, we start with the given seed.

```
1044 \tl_set:Nn \l__penrose_tmpb_tl {#4}
```

We repeat the specified number of times.

```
1045 \prg_replicate:nm {#3} {
```

Duplicate the current state.

```
1046 \tl_set_eq:NN \l__penrose_tmpa_tl \l__penrose_tmpb_tl
```

Clear the receiving token list.

```
1047 \tl_clear:N \l__penrose_tmpb_tl
```

Walk through the current list, appending to the receiving list according to the rules.

```
1048 \tl_map_inline:Nn \l__penrose_tmpa_tl
1049 {
```

If a rule exists, copy that.

```

1050   \prop_if_in:cnTF {g__penrose_#2_lms_rule_prop} {##1}
1051   {
1052     \tl_put_right:Nx \l__penrose_tmpb_tl
1053     {\prop_item:cn {g__penrose_#2_lms_rule_prop} {##1} }
1054   }
1055   {

```

Otherwise, just copy the token.

```

1056     \tl_put_right:Nn \l__penrose_tmpb_tl {##1}
1057   }
1058 }
1059 }

```

We've done all this inside a group, now pass the result outside.

```

1060 \tl_set:Nn \l__penrose_tmpa_tl {
1061   \group_end:
1062   \tl_set:Nn #1
1063 }
1064 \tl_put_right:Nx \l__penrose_tmpa_tl {{\tl_use:N \l__penrose_tmpb_tl}}
1065 \tl_use:N \l__penrose_tmpa_tl
1066 }
1067 \cs_generate_variant:Nn \__penrose_make_lms:Nnnn {Nnnx}

```

(End definition for `__penrose_make_lms:Nnnn`.)

`__penrose_invoke_lms:Nn` This carries out the actions specified by the resulting rules.

```

1068 \cs_new_nopar:Npn \__penrose_invoke_lms:Nn #1#2
1069 {
1070   \group_begin:

```

Walk through the given list, carrying out the corresponding action if it exists. If not, look at the default. Otherwise, just do nothing.

```

1071   \tl_map_inline:Nn #1 {
1072     \prop_if_in:cnTF {g__penrose_#2_lms_action_prop} {##1}
1073     {
1074       \prop_item:cn {g__penrose_#2_lms_action_prop} {##1}
1075     }
1076     {
1077       \prop_if_in:cnT {g__penrose_default_lms_action_prop} {##1}
1078       {
1079         \prop_item:cn {g__penrose_default_lms_action_prop} {##1}
1080       }
1081     }
1082   }
1083   \group_end:
1084 }

```

(End definition for `__penrose_invoke_lms:Nn`.)

We keep track of the number of tiles.

```

1085 \int_new:N \g__penrose_tile_int
1086 \int_new:N \g__penrose_tiles_int

```

`\PenroseDecomposition`

This is the user macro to invoke the decomposition. The arguments are: optional styles, the name, number of iterations, and starting seed.

```

1087 \NewDocumentCommand \PenroseDecomposition { 0{} m m m }
1088 {
1089   \group_begin:
1090   \tikzset{#1}
1091   \__penrose_make_lms:Nnnx \l__penrose_tmpa_tl {#2} {#3} {#4}
1092   \__penrose_count_lms:N \l__penrose_tmpa_tl
1093   \int_gzero:N \g__penrose_tile_int
1094   \int_set:Nn \l__penrose_pentagon_parity_int {2}
1095   \__penrose_invoke_lms:Nn \l__penrose_tmpa_tl {#2}
1096   \group_end:
1097 }

```

```

\__penrose_count_lms:N This counts the number of tiles in the string.
1098 \cs_new_nopar:Npn \__penrose_count_lms:N #1
1099 {
1100   \int_gzero:N \g__penrose_tiles_int
1101   \tl_map_inline:Nn #1
1102   {
1103     \tl_if_eq:nnT {##1} {T}
1104     {
1105       \int_gincr:N \g__penrose_tiles_int
1106     }
1107     \tl_if_eq:nnT {##1} {t}
1108     {
1109       \int_gincr:N \g__penrose_tiles_int
1110     }
1111     \tl_if_eq:nnT {##1} {G}
1112     {
1113       \int_gincr:N \g__penrose_tiles_int
1114     }
1115     \tl_if_eq:nnT {##1} {g}
1116     {
1117       \int_gincr:N \g__penrose_tiles_int
1118     }
1119   }
1120 }

```

(End definition for `__penrose_count_lms:N`.)

This is a `\tikzset` mechanism for setting the dimensions of the tiling.

```

1121 \tikzset{
1122   Penrose~ step/.code={
1123     \dim_set:Nn \l__penrose_step_dim {#1}
1124   }
1125 }

```

We're done with L^AT_EX3, so turn off the syntax.

```

1126 \ExplSyntaxOff

```

2.6 TikZ Pictures

New in TikZ3.0 is the ability to make pictures that can be reused. Penrose tiles seems an obvious use for this. These pictures can be placed alongside other tiles, matching by edge type.

There are a variety of constants that are frequently used and reused, so we define them all here. These are the PGF versions.

```

1127 \pgfmathsetmacro\pr@chphi{\cos(18)}
1128 \pgfmathsetmacro\pr@shphi{\sin(18)}
1129 \pgfmathsetmacro\pr@cphi{\cos(36)}
1130 \pgfmathsetmacro\pr@sphi{\sin(36)}
1131 \pgfmathsetmacro\pr@invphi{2/(sqrt(5)+1)}
1132 \pgfmathsetmacro\pr@phi{(sqrt(5)+1)/2}
1133 \pgfmathsetmacro\pr@invphisq{\pr@invphi*\pr@invphi}
1134 \pgfmathsetmacro\pr@ominvphisq{\pr@invphi - \pr@invphisq}
1135 \pgfmathsetmacro\pr@ominvphi{1 - \pr@invphi}

```

```

1136 \newif\if@edgealign
1137 \def\pr@getfirst#1#2\pr@stop{#1}%
1138 \def\test@edgealign#1{%
1139   \pgfkeysgetvalue{/tikz/Penrose/alignment edge}{\@penrose@tmpa}%
1140   \edef\@penrose@tmpb{#1}%
1141   \edef\@penrose@tmpa{\expandafter\pr@getfirst\@penrose@tmpa.\pr@stop}%
1142   \ifx\@penrose@tmpa\@penrose@tmpb
1143     \@edgealigntrue
1144   \else
1145     \@edgealignfalse
1146   \fi
1147 }%

1148 \newif\if@newedge
1149 \def\test@newedge#1{%
1150   \pgfkeysgetvalue{/tikz/Penrose/alignment new edge}{\@penrose@tmpa}%
1151   \edef\@penrose@tmpb{#1}%
1152   \edef\@penrose@tmpa{\@penrose@tmpa}%
1153   \ifx\@penrose@tmpa\@penrose@tmpb
1154     \@newedgetrue
1155   \else
1156     \@newedgefalse
1157   \fi
1158 }%

```

The implementation is essentially the same for each, so only the first will be commented.

```

1159 \tikzset{

```

The key `align with=<tile> along <edge>` is used to set the parameters for placing a tile next to an existing one. For most tiles, that's enough to specify how the new tile should be placed. Some, though, need more information. For those, use `align with=<tile> along <edge> using <number>`.

```

1160 align with/.code args={#1 along #2}{%
1161   \pgfutil@in@{ using }{#2}
1162   \ifpgfutil@in@%
1163     \tikzset{
1164       Penrose/alignment location=#1,
1165       Penrose/alignment set edges=#2,
1166     }%
1167   \else
1168     \tikzset{
1169       Penrose/alignment location=#1,
1170       Penrose/alignment edge=#2,
1171     }%
1172   \fi
1173 },
1174 Penrose/alignment set edges/.code args={#1 using #2}{%
1175   \tikzset{
1176     Penrose/alignment edge=#1,
1177     Penrose/alignment new edge=#2
1178   },
1179 },
1180 Penrose/alignment location/.initial={},
1181 Penrose/alignment edge/.initial=a,

```

1182 `Penrose/alignment new edge/.initial={},`

Default clipping style.

1183 `every Penrose tile clip/.style={clip},`

This is the code for setting up a `pic`.

1184 `thin rhombus/.pic={`

1185 `\begin{scope}`

Were we given a tile to align ourselves against?

1186 `\pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}`

1187 `\ifx\prloc\pgfutil@empty`

1188 `\else`

Yes, we were. So we adjust our position accordingly. The first job is to transform so that we're along the edge of the receiving tile.

1189 `\begin{group}`

We get the locations of the start and end of the receiving tile. As `pic` sets the node prefix, we have to temporarily suspend that (hence working in a `group`).

1190 `\tikzset{name prefix .}%`

1191 `\tikz@scan@one@point\pgfutil@firstofone%`

1192 `(\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%`

1193 `\global\pgf@xa=\pgf@x`

1194 `\global\pgf@ya=\pgf@y`

1195 `\tikz@scan@one@point\pgfutil@firstofone%`

1196 `(\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%`

1197 `\global\pgf@xb=\pgf@x`

1198 `\global\pgf@yb=\pgf@y`

1199 `\endgroup`

We store the initial points in `\pgf@xa` and `\pgf@ya` but we want `\pgf@xb` and `\pgf@yb` to be a vector along the edge.

1200 `\advance\pgf@xb by -\pgf@xa`

1201 `\advance\pgf@yb by -\pgf@ya`

We shift to the start of the edge.

1202 `\pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%`

And normalise the vector along it.

1203 `\pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}%`

1204 `\pgf@xb=\pgf@x`

1205 `\pgf@yb=\pgf@y`

Now rotate so that the x -axis lies along the edge.

1206 `\pgftransformtriangle%`

1207 `{\pgfpoint{0pt}{0pt}}%`

1208 `{\pgfpoint{\pgf@xb}{\pgf@yb}}%`

1209 `{\pgfpoint{-\pgf@yb}{\pgf@xb}}`

The next job is to shift and rotate the current tile so that the correct edge ends up against the receiving tile.

1210 `\if\pgfkeysvalueof{/tikz/Penrose/alignment edge}b\relax`

1211 `\pgftransformrotate{-18}%`

1212 `\pgftransformshift{\pgfpoint{-\pr@chphi cm}{\pr@shphi cm}}%`

1213 `\else`

1214 `\if\pgfkeysvalueof{/tikz/Penrose/alignment edge}B\relax`

1215 `\pgftransformrotate{18}%`

```

1216 \else
1217 \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}a\relax
1218 \pgftransformrotate{198}%
1219 \pgftransformshift{\pgfpoint{-2*\pr@chphi cm}{0 cm}}%
1220 \else
1221 \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}A\relax
1222 \pgftransformrotate{162}%
1223 \pgftransformshift{\pgfpoint{-\pr@chphi cm}{-\pr@shphi cm}}%
1224 \fi\fi\fi\fi
1225 \fi

```

Now that the transformation is finalised, we can render the tile. We clip against the tile path so that the tiles don't "bleed". If we didn't do this, drawing the tile would result in overlaps which can look a bit ugly. On the other hand, tight clipping can lead to "gaps" between the tiles so we make this optional by enclosing it in a style.

```

1226 \UsePenroseTile[
1227   every Penrose tile clip/.try,
1228   every thin rhombus clip/.try
1229 ]{thin rhombus}
1230 \UsePenroseTile[
1231   every Penrose tile/.try,
1232   every thin rhombus/.try,
1233   pic actions
1234 ]{thin rhombus}

```

These draw the arcs that designate the joining rules. We draw full circles so that it doesn't matter what shape the tiles are.

```

1235 \UsePenroseTile[
1236   every Penrose arc clip/.try,
1237   every thin rhombus arc clip/.try
1238 ]{thin rhombus}
1239 \path[every circle arc/.try] (18:1) circle[radius=1/4];
1240 \path[every long arc/.try] (-18:1) circle[radius=1/4];

```

Lastly, we put coordinates at each vertex, labelled by which edge they are.

```

1241 \coordinate (-edge a start) at (0,0);
1242 \coordinate (-edge a end) at (18:1);
1243 \coordinate (-edge A start) at (18:1);
1244 \coordinate (-edge A end) at (2*\pr@chphi,0);
1245 \coordinate (-edge B start) at (2*\pr@chphi,0);
1246 \coordinate (-edge B end) at (-18:1);
1247 \coordinate (-edge b start) at (-18:1);
1248 \coordinate (-edge b end) at (0,0);
1249 \end{scope}
1250 },

```

This is a shortcut for installing the pic type.

```

1251 thin rhombus/.style={
1252   every Penrose pic/.try,
1253   pic type=thin rhombus,
1254 },

```

Same again, but for the thick rhombus.

```

1255 thick rhombus/.pic={
1256   \begin{scope}
1257     \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}

```

```

1258 \ifx\prloc\pgfutil@empty
1259 \else
1260 \beginpgfgroup
1261 \tikzset{name prefix ..}%
1262 \tikz@scan@one@point\pgfutil@firstofone%
1263 (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1264 \global\pgf@xa=\pgf@x
1265 \global\pgf@ya=\pgf@y
1266 \tikz@scan@one@point\pgfutil@firstofone%
1267 (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1268 \global\pgf@xb=\pgf@x
1269 \global\pgf@yb=\pgf@y
1270 \endpgfgroup
1271 \advance\pgf@xb by -\pgf@xa
1272 \advance\pgf@yb by -\pgf@ya
1273 \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1274 \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1275 \pgf@xb=\pgf@x
1276 \pgf@yb=\pgf@y
1277 \pgftransformtriangle%
1278 {\pgfpoint{0pt}{0pt}}%
1279 {\pgfpoint{\pgf@xb}{\pgf@yb}}%
1280 {\pgfpoint{-\pgf@yb}{\pgf@xb}}
1281 \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}b\relax
1282 \pgftransformrotate{144}%
1283 \pgftransformshift{\pgfpoint{-\pr@cphi cm}{-\pr@sphi cm}}%
1284 \else
1285 \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}B\relax
1286 \pgftransformrotate{36}%
1287 \else
1288 \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}a\relax
1289 \pgftransformrotate{-36}%
1290 \pgftransformshift{\pgfpoint{-\pr@cphi cm}{\pr@sphi cm}}%
1291 \else
1292 \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}A\relax
1293 \pgftransformrotate{216}%
1294 \pgftransformshift{\pgfpoint{-2*\pr@cphi cm}{0 cm}}%
1295 \fi\fi\fi\fi
1296 \fi
1297 \UsePenroseTile[
1298   every Penrose tile clip/.try,
1299   every thick rhombus clip/.try
1300 ]{thick rhombus}
1301 \UsePenroseTile[
1302   every Penrose tile/.try,
1303   every thick rhombus/.try,
1304   pic actions
1305 ]{thick rhombus}
1306 \UsePenroseTile[
1307   every Penrose arc clip/.try,
1308   every thick rhombus arc clip/.try
1309 ]{thick rhombus}
1310 \path[every circle arc/.try] (2*\pr@cphi,0) circle[radius=1/4];
1311 \path[every long arc/.try] (0,0) circle[radius=3/4];

```



```

1312 \coordinate (-edge B start) at (0,0);
1313 \coordinate (-edge B end) at (36:1);
1314 \coordinate (-edge a start) at (36:1);
1315 \coordinate (-edge a end) at (2*\pr@cphi,0);
1316 \coordinate (-edge A start) at (2*\pr@cphi,0);
1317 \coordinate (-edge A end) at (-36:1);
1318 \coordinate (-edge b start) at (-36:1);
1319 \coordinate (-edge b end) at (0,0);
1320 \end{scope}
1321 },
1322 thick rhombus/.style={
1323   every Penrose pic/.try,
1324   pic type=thick rhombus,
1325 },

```

Now the kite.

```

1326 kite/.pic={
1327   \begin{scope}
1328     \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
1329     \ifx\prloc\pgfutil@empty
1330       \else
1331         \begingroup
1332         \tikzset{name prefix ..}%
1333         \tikz@scan@one@point\pgfutil@firstofone%
1334         (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1335         \global\pgf@xa=\pgf@x
1336         \global\pgf@ya=\pgf@y
1337         \tikz@scan@one@point\pgfutil@firstofone%
1338         (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1339         \global\pgf@xb=\pgf@x
1340         \global\pgf@yb=\pgf@y
1341         \endgroup
1342         \advance\pgf@xb by -\pgf@xa
1343         \advance\pgf@yb by -\pgf@ya
1344         \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1345         \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1346         \pgf@xb=\pgf@x
1347         \pgf@yb=\pgf@y
1348         \pgftransformtriangle%
1349         {\pgfpoint{0pt}{0pt}}%
1350         {\pgfpoint{\pgf@xb}{\pgf@yb}}%
1351         {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
1352         \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}c\relax
1353         \pgftransformrotate{-72}%
1354         \pgftransformshift{\pgfpoint{-\pr@cphi cm}{\pr@sphi cm}}%
1355         \else
1356         \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}C\relax
1357         \pgftransformrotate{-108}%
1358         \pgftransformshift{\pgfpoint{-1 cm}{0 cm}}%
1359         \else
1360         \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}a\relax
1361         \pgftransformrotate{36}%
1362         \else
1363         \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}A\relax
1364         \pgftransformrotate{144}%

```

```

1365 \pgftransformshift{\pgfpoint{-\pr@cphi cm}{-\pr@sphi cm}}%
1366 \fi\fi\fi\fi
1367 \fi
1368 \UsePenroseTile[
1369     every Penrose tile clip/.try,
1370     every kite clip/.try
1371 ]{kite}
1372 \UsePenroseTile[
1373     every Penrose tile/.try,
1374     every kite/.try,
1375     pic actions
1376 ]{kite}
1377 \UsePenroseTile[
1378     every Penrose arc clip/.try,
1379     every kite arc clip/.try
1380 ]{kite}
1381 \path[every circle arc/.try] (0,0) circle[radius=\pr@invphi];
1382 \path[every long arc/.try] (1,0) circle[radius=\pr@invphisq];
1383 \coordinate (-edge a start) at (0,0);
1384 \coordinate (-edge a end) at (36:1);
1385 \coordinate (-edge c start) at (36:1);
1386 \coordinate (-edge c end) at (1,0);
1387 \coordinate (-edge C start) at (1,0);
1388 \coordinate (-edge C end) at (-36:1);
1389 \coordinate (-edge A start) at (-36:1);
1390 \coordinate (-edge A end) at (0,0);
1391 \end{scope}
1392 },

```

The dart is next.

```

1393 dart/.pic={
1394     \begin{scope}
1395     \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
1396     \ifx\prloc\pgfutil@empty
1397     \else
1398     \begingroup
1399     \tikzset{name prefix ..}%
1400     \tikz@scan@one@point\pgfutil@firstofone%
1401     (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1402     \global\pgf@xa=\pgf@x
1403     \global\pgf@ya=\pgf@y
1404     \tikz@scan@one@point\pgfutil@firstofone%
1405     (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1406     \global\pgf@xb=\pgf@x
1407     \global\pgf@yb=\pgf@y
1408     \endgroup
1409     \advance\pgf@xb by -\pgf@xa
1410     \advance\pgf@yb by -\pgf@ya
1411     \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1412     \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1413     \pgf@xb=\pgf@x
1414     \pgf@yb=\pgf@y
1415     \pgftransformtriangle%
1416     {\pgfpoint{0pt}{0pt}}%
1417     {\pgfpoint{\pgf@xb}{\pgf@yb}}%

```

```

1418   {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
1419   \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}c\relax
1420   \pgftransformrotate{108}%
1421   \else
1422   \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}C\relax
1423   \pgftransformrotate{72}%
1424   \pgftransformshift%
1425   {\pgfpoint{\pr@invphi*\pr@shphi cm}{-\pr@invphi*\pr@chphi cm}}%
1426   \else
1427   \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}a\relax
1428   \pgftransformrotate{-36}%
1429   \pgftransformshift%
1430   {\pgfpoint{\pr@invphi*\pr@shphi cm}{\pr@invphi*\pr@chphi cm}}%
1431   \else
1432   \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}A\relax
1433   \pgftransformrotate{216}%
1434   \pgftransformshift{\pgfpoint{-\pr@invphi cm}{0 cm}}%
1435   \fi\fi\fi\fi
1436   \fi
1437   \UsePenroseTile[
1438     every Penrose tile clip/.try,
1439     every dart clip/.try
1440   ]{dart}
1441   \UsePenroseTile[
1442     every Penrose tile/.try,
1443     every dart/.try,
1444     pic actions
1445   ]{dart}
1446   \UsePenroseTile[
1447     every Penrose arc clip/.try,
1448     every dart arc clip/.try
1449   ]{dart}
1450   \path[every circle arc/.try] (\pr@invphi,0) circle[radius=\pr@ominvphi];
1451   \path[every long arc/.try] (0,0) circle[radius=\pr@ominvphisq];
1452   \coordinate (-edge c start) at (0,0);
1453   \coordinate (-edge c end) at (108:\pr@invphi);
1454   \coordinate (-edge a start) at (108:\pr@invphi);
1455   \coordinate (-edge a end) at (\pr@invphi,0);
1456   \coordinate (-edge A start) at (\pr@invphi,0);
1457   \coordinate (-edge A end) at (-108:\pr@invphi);
1458   \coordinate (-edge C start) at (-108:\pr@invphi);
1459   \coordinate (-edge C end) at (0,0);
1460   \end{scope}
1461   },
1462   kite/.style={
1463     every Penrose pic/.try,
1464     pic type=kite,
1465   },
1466   dart/.style={
1467     every Penrose pic/.try,
1468     pic type=dart,
1469   },

```

The golden triangle.

```

1470 golden triangle/.pic={
1471   \begin{scope}
1472     \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
1473     \ifx\prloc\pgfutil@empty
1474       \else
1475         \begingroup
1476         \tikzset{name prefix ..}%
1477         \tikz@scan@one@point\pgfutil@firstofone%
1478         (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1479         \global\pgf@xa=\pgf@x
1480         \global\pgf@ya=\pgf@y
1481         \tikz@scan@one@point\pgfutil@firstofone%
1482         (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1483         \global\pgf@xb=\pgf@x
1484         \global\pgf@yb=\pgf@y
1485         \endgroup
1486         \advance\pgf@xb by -\pgf@xa
1487         \advance\pgf@yb by -\pgf@ya
1488         \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1489         \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1490         \pgf@xb=\pgf@x
1491         \pgf@yb=\pgf@y
1492         \pgftransformtriangle%
1493         {\pgfpoint{0pt}{0pt}}%
1494         {\pgfpoint{\pgf@xb}{\pgf@yb}}%
1495         {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
1496         \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}B\relax
1497         \pgftransformrotate{18}%
1498         \else
1499         \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}C\relax
1500         \pgftransformrotate{-90}%
1501         \pgftransformshift{\pgfpoint{-\pr@chphi cm}{\pr@shphi cm}}%
1502         \else
1503         \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}A\relax
1504         \pgftransformrotate{162}%
1505         \pgftransformshift{\pgfpoint{-\pr@chphi cm}{-\pr@shphi cm}}%
1506         \fi\fi\fi
1507         \fi
1508         \UsePenroseTile[
1509           every Penrose tile clip/.try,
1510           every golden triangle clip/.try
1511         ]{golden triangle}
1512         \UsePenroseTile[
1513           every Penrose tile/.try,
1514           every golden triangle/.try,
1515           pic actions
1516         ]{golden triangle}
1517         \coordinate (-edge a start) at (0,0);
1518         \coordinate (-edge a end) at (18:1);
1519         \coordinate (-edge c start) at (18:1);
1520         \coordinate (-edge c end) at (-18:1);
1521         \coordinate (-edge b start) at (-18:1);
1522         \coordinate (-edge b end) at (0,0);
1523         \end{scope}

```

```

1524 },
1525 golden triangle/.style={
1526   every Penrose pic/.try,
1527   pic type=golden triangle,
1528 },

```

The reverse golden triangle (is there a better name?).

```

1529 reverse golden triangle/.pic={
1530   \begin{scope}
1531     \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
1532     \ifx\prloc\pgfutil@empty
1533     \else
1534     \begingroup
1535     \tikzset{name prefix ..}%
1536     \tikz@scan@one@point\pgfutil@firstofone%
1537     (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1538     \global\pgf@xa=\pgf@x
1539     \global\pgf@ya=\pgf@y
1540     \tikz@scan@one@point\pgfutil@firstofone%
1541     (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1542     \global\pgf@xb=\pgf@x
1543     \global\pgf@yb=\pgf@y
1544     \endgroup
1545     \advance\pgf@xb by -\pgf@xa
1546     \advance\pgf@yb by -\pgf@ya
1547     \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1548     \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1549     \pgf@xb=\pgf@x
1550     \pgf@yb=\pgf@y
1551     \pgftransformtriangle%
1552     {\pgfpoint{0pt}{0pt}}%
1553     {\pgfpoint{\pgf@xb}{\pgf@yb}}%
1554     {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
1555     \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}b\relax
1556     \pgftransformrotate{162}%
1557     \pgftransformshift{\pgfpoint{-\pr@chphi cm}{-\pr@shphi cm}}%
1558     \else
1559     \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}c\relax
1560     \pgftransformrotate{-90}%
1561     \pgftransformshift{\pgfpoint{-\pr@chphi cm}{\pr@shphi cm}}%
1562     \else
1563     \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}a\relax
1564     \pgftransformrotate{18}%
1565     \fi\fi\fi
1566     \fi
1567     \UsePenroseTile[
1568       every Penrose tile clip/.try,
1569       every reverse golden triangle clip/.try
1570     ]{reverse golden triangle}
1571     \UsePenroseTile[
1572       every Penrose tile/.try,
1573       every reverse golden triangle/.try,
1574       pic actions
1575     ]{reverse golden triangle}
1576     \coordinate (-edge B start) at (0,0);

```

```

1577 \coordinate (-edge B end) at (18:1);
1578 \coordinate (-edge C start) at (18:1);
1579 \coordinate (-edge C end) at (-18:1);
1580 \coordinate (-edge A start) at (-18:1);
1581 \coordinate (-edge A end) at (0,0);
1582   \end{scope}
1583 },
1584 reverse golden triangle/.style={
1585   every Penrose pic/.try,
1586   pic type=reverse golden triangle,
1587 },
The golden gnomon.
1588 golden gnomon/.pic={
1589   \begin{scope}
1590     \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
1591     \ifx\prloc\pgfutil@empty
1592     \else
1593     \begin{group}
1594       \tikzset{name prefix ..}%
1595       \tikz@scan@one@point\pgfutil@firstofone%
1596       (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1597       \global\pgf@xa=\pgf@x
1598       \global\pgf@ya=\pgf@y
1599       \tikz@scan@one@point\pgfutil@firstofone%
1600       (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1601       \global\pgf@xb=\pgf@x
1602       \global\pgf@yb=\pgf@y
1603     \end{group}
1604     \advance\pgf@xb by -\pgf@xa
1605     \advance\pgf@yb by -\pgf@ya
1606     \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1607     \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1608     \pgf@xb=\pgf@x
1609     \pgf@yb=\pgf@y
1610     \pgftransformtriangle%
1611     {\pgfpoint{0pt}{0pt}}%
1612     {\pgfpoint{\pgf@xb}{\pgf@yb}}%
1613     {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
1614     \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}c\relax
1615     \pgftransformrotate{144}%
1616     \pgftransformshift{\pgfpoint{-\pr@cphi cm}{-\pr@sphi cm}}%
1617     \else
1618     \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}B\relax
1619     \pgftransformrotate{-144}%
1620     \pgftransformshift{\pgfpoint{-2*\pr@cphi cm}{0 cm}}%
1621     \else
1622     \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}a\relax
1623     \fi\fi\fi
1624     \fi
1625     \UsePenroseTile[
1626       every Penrose tile clip/.try,
1627       every golden gnomon clip/.try
1628     ]{golden gnomon}
1629     \UsePenroseTile[

```

```

1630     every Penrose tile/.try,
1631     every golden gnomon/.try,
1632     pic actions
1633   ]{golden gnomon}
1634 \coordinate (-edge C start) at (0,0);
1635 \coordinate (-edge C end) at (36:1);
1636 \coordinate (-edge b start) at (36:1);
1637 \coordinate (-edge b end) at (2*\pr@cphi,0);
1638 \coordinate (-edge A start) at (2*\pr@cphi,0);
1639 \coordinate (-edge A end) at (0,0);
1640 \end{scope}
1641 },
1642 golden gnomon/.style={
1643   every Penrose pic/.try,
1644   pic type=golden gnomon,
1645 },

```

The reverse golden gnomon.

```

1646 reverse golden gnomon/.pic={
1647   \begin{scope}
1648     \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
1649     \ifx\prloc\pgfutil@empty
1650     \else
1651     \begingroup
1652     \tikzset{name prefix ..}%
1653     \tikz@scan@one@point\pgfutil@firstofone%
1654     (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1655     \global\pgf@xa=\pgf@x
1656     \global\pgf@ya=\pgf@y
1657     \tikz@scan@one@point\pgfutil@firstofone%
1658     (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1659     \global\pgf@xb=\pgf@x
1660     \global\pgf@yb=\pgf@y
1661     \endgroup
1662     \advance\pgf@xb by -\pgf@xa
1663     \advance\pgf@yb by -\pgf@ya
1664     \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1665     \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1666     \pgf@xb=\pgf@x
1667     \pgf@yb=\pgf@y
1668     \pgftransformtriangle%
1669     {\pgfpoint{0pt}{0pt}}%
1670     {\pgfpoint{\pgf@xb}{\pgf@yb}}%
1671     {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
1672     \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}C\relax
1673     \pgftransformrotate{36}%
1674     \else
1675     \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}b\relax
1676     \pgftransformrotate{-36}%
1677     \pgftransformshift{\pgfpoint{-\pr@cphi cm}{\pr@sphi cm}}%
1678     \else
1679     \if\pgfkeysvalueof{/tikz/Penrose/alignment edge}A\relax
1680     \pgftransformrotate{180}%
1681     \pgftransformshift{\pgfpoint{-2*\pr@cphi cm}{0 cm}}%
1682     \fi\fi\fi

```

```

1683 \fi
1684 \UsePenroseTile[
1685     every Penrose tile clip/.try,
1686     every reverse golden gnomon clip/.try
1687 ]{reverse golden gnomon}
1688 \UsePenroseTile[
1689     every Penrose tile/.try,
1690     every reverse golden gnomon/.try,
1691     pic actions
1692 ]{reverse golden gnomon}
1693 \coordinate (-edge a start) at (0,0);
1694 \coordinate (-edge a end) at (2*\pr@cphi,0);
1695 \coordinate (-edge B start) at (2*\pr@cphi,0);
1696 \coordinate (-edge B end) at (-36:1);
1697 \coordinate (-edge c start) at (-36:1);
1698 \coordinate (-edge c end) at (0,0);
1699 \end{scope}
1700 },
1701 reverse golden gnomon/.style={
1702     every Penrose pic/.try,
1703     pic type=reverse golden gnomon,
1704 },

```

The primary pentagon.

```

1705 pentagon 5/.pic={
1706     \begin{scope}
1707         \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
1708         \ifx\prloc\pgfutil@empty
1709             \else
1710                 \begin{group}
1711                     \tikzset{name prefix ..}%
1712                     \tikz@scan@one@point\pgfutil@firstofone%
1713                     (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1714                     \global\pgf@xa=\pgf@x
1715                     \global\pgf@ya=\pgf@y
1716                     \tikz@scan@one@point\pgfutil@firstofone%
1717                     (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1718                     \global\pgf@xb=\pgf@x
1719                     \global\pgf@yb=\pgf@y
1720                 \end{group}
1721                 \advance\pgf@xb by -\pgf@xa
1722                 \advance\pgf@yb by -\pgf@ya
1723                 \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1724                 \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1725                 \pgf@xb=\pgf@x
1726                 \pgf@yb=\pgf@y
1727                 \pgftransformtriangle%
1728                 {\pgfpoint{0pt}{0pt}}%
1729                 {\pgfpoint{\pgf@xb}{\pgf@yb}}%
1730                 {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
1731                 \test@newedge{1}%
1732                 \if@newedge
1733                     \pgftransformrotate{180}%
1734                     \pgftransformshift{\pgfpoint{-1 cm}{0 cm}}%
1735                 \else

```



```

1736 \test@newedge{2}%
1737 \if@newedge
1738 \pgftransformshift{\pgfpoint{1 cm}{0 cm}}%
1739 \pgftransformrotate{108}%
1740 \pgftransformshift{\pgfpoint{-1 cm}{0 cm}}%
1741 \else
1742 \test@newedge{3}%
1743 \if@newedge
1744 \pgftransformshift%
1745 {\pgfpoint{\pr@phi * \pr@shphi cm}{- \pr@phi * \pr@chphi cm}}%
1746 \pgftransformrotate{36}%
1747 \else
1748 \test@newedge{4}%
1749 \if@newedge
1750 \pgftransformshift{\pgfpoint{- \pr@shphi cm}{- \pr@chphi cm}}%
1751 \pgftransformrotate{-36}%
1752 \else
1753 \test@newedge{5}%
1754 \if@newedge
1755 \pgftransformrotate{-108}%
1756 \fi\fi\fi\fi\fi
1757 \fi
1758 \UsePenroseTile[
1759     every Penrose tile clip/.try,
1760     every pentagon clip/.try,
1761     every pentagon 5 clip/.try
1762 ]{pentagon 5}
1763 \UsePenroseTile[
1764     every Penrose tile/.try,
1765     every pentagon/.try,
1766     every pentagon 5/.try,
1767     pic actions
1768 ]{pentagon 5}
1769 \coordinate (-edge a1 start) at (0,0);
1770 \coordinate (-edge a1 end) at (1,0);
1771 \coordinate (-edge a2 start) at (1,0);
1772 \coordinate (-edge a2 end) at (1+\pr@shphi,\pr@chphi);
1773 \coordinate (-edge a3 start) at (1+\pr@shphi,\pr@chphi);
1774 \coordinate (-edge a3 end) at (\pr@cphi-\pr@shphi,\pr@sphi+\pr@chphi);
1775 \coordinate (-edge a4 start) at (\pr@cphi-\pr@shphi,\pr@sphi+\pr@chphi);
1776 \coordinate (-edge a4 end) at (-\pr@shphi,\pr@chphi);
1777 \coordinate (-edge a5 start) at (-\pr@shphi,\pr@chphi);
1778 \coordinate (-edge a5 end) at (0,0);
1779 \end{scope}
1780 },
1781 pentagon 5/.style={
1782     every Penrose pic/.try,
1783     pic type=pentagon 5,
1784 },

```

The secondary pentagon.

```

1785 pentagon 3/.pic={
1786     \begin{scope}
1787         \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
1788         \ifx\prloc\pgfutil@empty

```

```

1789 \else
1790 \beginpgfgroup
1791 \tikzset{name prefix ..}%
1792 \tikz@scan@one@point\pgfutil@firstofone%
1793 (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1794 \global\pgf@xa=\pgf@x
1795 \global\pgf@ya=\pgf@y
1796 \tikz@scan@one@point\pgfutil@firstofone%
1797 (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1798 \global\pgf@xb=\pgf@x
1799 \global\pgf@yb=\pgf@y
1800 \endpgfgroup
1801 \advance\pgf@xb by -\pgf@xa
1802 \advance\pgf@yb by -\pgf@ya
1803 \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1804 \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1805 \pgf@xb=\pgf@x
1806 \pgf@yb=\pgf@y
1807 \pgftransformtriangle%
1808 {\pgfpoint{0pt}{0pt}}%
1809 {\pgfpoint{\pgf@xb}{\pgf@yb}}%
1810 {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
1811 \test@edgealign{a}%
1812 \if@edgealign
1813 \pgftransformrotate{180}%
1814 \pgftransformshift{\pgfpoint{-1 cm}{0 cm}}%
1815 \else
1816 \test@edgealign{B}%
1817 \if@edgealign%
1818 \test@newedge{1}%
1819 \if@newedge
1820 \pgftransformshift{\pgfpoint{1 cm}{0 cm}}%
1821 \pgftransformrotate{108}%
1822 \pgftransformshift{\pgfpoint{-1 cm}{0 cm}}%
1823 \else
1824 \pgftransformrotate{-108}%
1825 \fi
1826 \else
1827 \test@newedge{1}%
1828 \if@newedge
1829 \pgftransformshift%
1830 {\pgfpoint{\pr@phi * \pr@shphi cm}{-\pr@phi * \pr@chphi cm}}%
1831 \pgftransformrotate{36}%
1832 \else
1833 \pgftransformshift{\pgfpoint{-\pr@shphi cm}{-\pr@chphi cm}}%
1834 \pgftransformrotate{-36}%
1835 \fi\fi\fi
1836 \fi
1837 \UsePenroseTile[
1838 every Penrose tile clip/.try,
1839 every pentagon clip/.try,
1840 every pentagon 3 clip/.try
1841 ]{pentagon 3}
1842 \UsePenroseTile[

```

```

1843     every Penrose tile/.try,
1844     every pentagon/.try,
1845     every pentagon 3/.try,
1846     pic actions
1847   ]{pentagon 3}
1848   \coordinate (-edge A start) at (0,0);
1849   \coordinate (-edge A end) at (1,0);
1850   \coordinate (-edge b1 start) at (1,0);
1851   \coordinate (-edge b1 end) at (1+\pr@shphi,\pr@chphi);
1852   \coordinate (-edge a1 start) at (1+\pr@shphi,\pr@chphi);
1853   \coordinate (-edge a1 end) at (\pr@cphi-\pr@shphi,\pr@sphi+\pr@chphi);
1854   \coordinate (-edge a2 start) at (\pr@cphi-\pr@shphi,\pr@sphi+\pr@chphi);
1855   \coordinate (-edge a2 end) at (-\pr@shphi,\pr@chphi);
1856   \coordinate (-edge b2 start) at (-\pr@shphi,\pr@chphi);
1857   \coordinate (-edge b2 end) at (0,0);
1858   \end{scope}
1859 },
1860 pentagon 3/.style={
1861   every Penrose pic/.try,
1862   pic type=pentagon 3,
1863 },

```

The tertiary pentagon.

```

1864 pentagon 2/.pic={
1865   \begin{scope}
1866     \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
1867     \ifx\prloc\pgfutil@empty
1868     \else
1869     \begin{group}
1870       \tikzset{name prefix ..}%
1871       \tikz@scan@one@point\pgfutil@firstofone%
1872       (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1873       \global\pgf@xa=\pgf@x
1874       \global\pgf@ya=\pgf@y
1875       \tikz@scan@one@point\pgfutil@firstofone%
1876       (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1877       \global\pgf@xb=\pgf@x
1878       \global\pgf@yb=\pgf@y
1879     \end{group}
1880     \advance\pgf@xb by -\pgf@xa
1881     \advance\pgf@yb by -\pgf@ya
1882     \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1883     \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1884     \pgf@xb=\pgf@x
1885     \pgf@yb=\pgf@y
1886     \pgftransformtriangle%
1887     {\pgfpoint{0pt}{0pt}}%
1888     {\pgfpoint{\pgf@xb}{\pgf@yb}}%
1889     {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
1890     \test@edgealign{D}%
1891     \if@edgealign
1892     \pgftransformrotate{180}%
1893     \pgftransformshift{\pgfpoint{-1 cm}{0 cm}}%
1894     \else
1895     \test@edgealign{a}%

```

```

1896 \if@edgealign%
1897 \test@newedge{1}%
1898 \if@newedge
1899 \pgftransformshift{\pgfpoint{1 cm}{0 cm}}%
1900 \pgftransformrotate{108}%
1901 \pgftransformshift{\pgfpoint{-1 cm}{0 cm}}%
1902 \else
1903 \pgftransformrotate{-108}%
1904 \fi
1905 \else
1906 \test@newedge{1}%
1907 \if@newedge
1908 \pgftransformshift%
1909 {\pgfpoint{\pr@phi * \pr@shphi cm}{- \pr@phi * \pr@chphi cm}}%
1910 \pgftransformrotate{36}%
1911 \else
1912 \pgftransformshift{\pgfpoint{- \pr@shphi cm}{- \pr@chphi cm}}%
1913 \pgftransformrotate{-36}%
1914 \fi\fi\fi
1915 \fi
1916 \UsePenroseTile[
1917     every Penrose tile clip/.try,
1918     every pentagon clip/.try,
1919     every pentagon 2 clip/.try
1920 ]{pentagon 2}
1921 \UsePenroseTile[
1922     every Penrose tile/.try,
1923     every pentagon/.try,
1924     every pentagon 2/.try,
1925     pic actions
1926 ]{pentagon 2}
1927 \coordinate (-edge d start) at (0,0);
1928 \coordinate (-edge d end) at (1,0);
1929 \coordinate (-edge A1 start) at (1,0);
1930 \coordinate (-edge A1 end) at (1+\pr@shphi,\pr@chphi);
1931 \coordinate (-edge c1 start) at (1+\pr@shphi,\pr@chphi);
1932 \coordinate (-edge c1 end) at (\pr@cphi-\pr@shphi,\pr@sphi+\pr@chphi);
1933 \coordinate (-edge c2 start) at (\pr@cphi-\pr@shphi,\pr@sphi+\pr@chphi);
1934 \coordinate (-edge c2 end) at (-\pr@shphi,\pr@chphi);
1935 \coordinate (-edge A2 start) at (-\pr@shphi,\pr@chphi);
1936 \coordinate (-edge A2 end) at (0,0);
1937 \end{scope}
1938 },
1939 pentagon 2/.style={
1940     every Penrose pic/.try,
1941     pic type=pentagon 2,
1942 },

```

The pentagram.

```

1943 pentagram/.pic={
1944     \begin{scope}
1945     \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
1946     \ifx\prloc\pgfutil@empty
1947     \else
1948     \begingroup

```

```

1949 \tikzset{name prefix ..}%
1950 \tikz@scan@one@point\pgfutil@firstofone%
1951 (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
1952 \global\pgf@xa=\pgf@x
1953 \global\pgf@ya=\pgf@y
1954 \tikz@scan@one@point\pgfutil@firstofone%
1955 (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
1956 \global\pgf@xb=\pgf@x
1957 \global\pgf@yb=\pgf@y
1958 \endgroup
1959 \advance\pgf@xb by -\pgf@xa
1960 \advance\pgf@yb by -\pgf@ya
1961 \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
1962 \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
1963 \pgf@xb=\pgf@x
1964 \pgf@yb=\pgf@y
1965 \pgftransformtriangle%
1966 {\pgfpoint{0pt}{0pt}}%
1967 {\pgfpoint{\pgf@xb}{\pgf@yb}}%
1968 {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
1969 \test@newedge{2}%
1970 \if@newedge
1971 \pgftransformshift{\pgfpoint{1 cm}{0 cm}}%
1972 \pgftransformrotate{72}%
1973 \else
1974 \test@newedge{3}%
1975 \if@newedge
1976 \pgftransformrotate{-72}%
1977 \pgftransformshift{\pgfpoint{2 * \pr@shphi cm}{0 cm}}%
1978 \else
1979 \test@newedge{4}%
1980 \if@newedge
1981 \pgftransformshift{\pgfpoint{1 cm + 2 * \pr@shphi cm}{0 cm}}%
1982 \else
1983 \test@newedge{5}%
1984 \if@newedge
1985 \pgftransformrotate{216}%
1986 \pgftransformshift{\pgfpoint{\pr@cphi cm}{\pr@sphi cm}}%
1987 \else
1988 \test@newedge{6}%
1989 \if@newedge
1990 \pgftransformshift{\pgfpoint{1cm}{0cm}}%
1991 \pgftransformrotate{-72}%
1992 \pgftransformshift{\pgfpoint{\pr@cphi cm}{\pr@sphi cm}}%
1993 \else
1994 \test@newedge{7}%
1995 \if@newedge
1996 \pgftransformrotate{144}%
1997 \pgftransformshift{\pgfpoint{\pr@shphi cm}{\pr@chphi cm}}%
1998 \else
1999 \test@newedge{8}%
2000 \if@newedge
2001 \pgftransformshift{\pgfpoint{1 cm + 2*\pr@shphi cm}{0cm}}%
2002 \pgftransformrotate{216}%

```

```

2003 \pgftransformshift{\pgfpoint{\pr@cphi cm}{\pr@sphi cm}}%
2004 \else
2005 \test@newedge{9}%
2006 \if@newedge
2007 \pgftransformshift{\pgfpoint{-2*\pr@shphi cm}{0cm}}%
2008 \pgftransformrotate{72}%
2009 \else
2010 \test@newedge{10}%
2011 \if@newedge
2012 \pgftransformrotate{144}%
2013 \pgftransformshift{\pgfpoint{-1 cm}{0 cm}}%
2014 \else
2015 \fi\fi\fi\fi\fi\fi\fi\fi\fi\fi
2016 \fi
2017 \UsePenroseTile[
2018     every Penrose tile clip/.try,
2019     every pentagram clip/.try
2020 ]{pentagram}
2021 \UsePenroseTile[
2022     every Penrose tile/.try,
2023     every pentagram/.try,
2024     pic actions
2025 ]{pentagram}
2026 \coordinate (-edge C1 start) at (1,0);
2027 \coordinate (-edge C1 end) at (0,0);
2028 \coordinate (-edge C2 start) at (0,0);
2029 \coordinate (-edge C2 end) at (-\pr@shphi,\pr@chphi);
2030 \coordinate (-edge C3 start) at (-\pr@shphi,\pr@chphi);
2031 \coordinate (-edge C3 end) at (-2*\pr@shphi,0);
2032 \coordinate (-edge C4 start) at (-2*\pr@shphi,0);
2033 \coordinate (-edge C4 end) at (-1-2*\pr@shphi,0);
2034 \coordinate (-edge C5 start) at (-1-2*\pr@shphi,0);
2035 \coordinate (-edge C5 end) at (-\pr@cphi,-\pr@sphi);
2036 \coordinate (-edge C6 start) at (-\pr@cphi,-\pr@sphi);
2037 \coordinate (-edge C6 end) at (-\pr@cphi-\pr@shphi,-\pr@sphi-\pr@chphi);
2038 \coordinate (-edge C7 start) at (-\pr@cphi-\pr@shphi,-\pr@sphi-\pr@chphi);
2039 \coordinate (-edge C7 end) at (-\pr@shphi,-\pr@chphi);
2040 \coordinate (-edge C8 start) at (-\pr@shphi,-\pr@chphi);
2041 \coordinate (-edge C8 end) at (1-\pr@cphi+\pr@shphi,-\pr@sphi-\pr@chphi);
2042 \coordinate (-edge C9 start) at (1-\pr@cphi+\pr@shphi,-\pr@sphi-\pr@chphi);
2043 \coordinate (-edge C9 end) at (1-\pr@cphi,-\pr@sphi);
2044 \coordinate (-edge C10 start) at (1-\pr@cphi,-\pr@sphi);
2045 \coordinate (-edge C10 end) at (1,0);
2046 \end{scope}
2047 },
2048 pentagram/.style={
2049     every Penrose pic/.try,
2050     pic type=pentagram,
2051 },
The boat.
2052 boat/.pic={
2053     \begin{scope}
2054         \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
2055         \ifx\prloc\pgfutil@empty

```

```

2056 \else
2057 \beginpgfgroup
2058 \tikzset{name prefix ..}%
2059 \tikz@scan@one@point\pgfutil@firstofone%
2060 (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
2061 \global\pgf@xa=\pgf@x
2062 \global\pgf@ya=\pgf@y
2063 \tikz@scan@one@point\pgfutil@firstofone%
2064 (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
2065 \global\pgf@xb=\pgf@x
2066 \global\pgf@yb=\pgf@y
2067 \endpgfgroup
2068 \advance\pgf@xb by -\pgf@xa
2069 \advance\pgf@yb by -\pgf@ya
2070 \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
2071 \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
2072 \pgf@xb=\pgf@x
2073 \pgf@yb=\pgf@y
2074 \pgftransformtriangle%
2075 {\pgfpoint{0pt}{0pt}}%
2076 {\pgfpoint{\pgf@xb}{\pgf@yb}}%
2077 {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
2078 \test@edgealign{d}%
2079 \if@edgealign
2080 \pgftransformrotate{180}%
2081 \pgftransformshift{\pgfpoint{\pr@cphi cm - 1cm}{\pr@sphi cm}}%
2082 \else
2083 \test@edgealign{b}%
2084 \if@edgealign%
2085 \test@newedge{2}%
2086 \if@newedge
2087 \pgftransformrotate{144}%
2088 \pgftransformshift{\pgfpoint{-1 cm}{0 cm}}%
2089 \else
2090 \pgftransformrotate{216}%
2091 \pgftransformshift{\pgfpoint{\pr@cphi cm}{\pr@sphi cm}}%
2092 \fi
2093 \else
2094 \test@edgealign{c}%
2095 \if@edgealign%
2096 \test@newedge{2}%
2097 \if@newedge
2098 \pgftransformshift{\pgfpoint{1 cm}{0 cm}}%
2099 \pgftransformrotate{72}%
2100 \else
2101 \test@newedge{3}%
2102 \if@newedge
2103 \pgftransformrotate{-72}%
2104 \pgftransformshift{\pgfpoint{2 * \pr@shphi cm}{0 cm}}%
2105 \else
2106 \test@newedge{4}%
2107 \if@newedge
2108 \pgftransformshift{\pgfpoint{1 cm + 2 * \pr@shphi cm}{0 cm}}%
2109 \else

```

```

2110 \fi\fi\fi\fi\fi\fi
2111 \fi
2112 \UsePenroseTile[
2113     every Penrose tile clip/.try,
2114     every boat clip/.try
2115 ]{boat}
2116 \UsePenroseTile[
2117     every Penrose tile/.try,
2118     every boat/.try,
2119     pic actions
2120 ]{boat}
2121 \coordinate (-edge C1 start) at (1,0);
2122 \coordinate (-edge C1 end) at (0,0);
2123 \coordinate (-edge C2 start) at (0,0);
2124 \coordinate (-edge C2 end) at (-\pr@shphi,\pr@chphi);
2125 \coordinate (-edge C3 start) at (-\pr@shphi,\pr@chphi);
2126 \coordinate (-edge C3 end) at (-2*\pr@shphi,0);
2127 \coordinate (-edge C4 start) at (-2*\pr@shphi,0);
2128 \coordinate (-edge C4 end) at (-1-2*\pr@shphi,0);
2129 \coordinate (-edge B1 start) at (-1-2*\pr@shphi,0);
2130 \coordinate (-edge B1 end) at (-\pr@cphi,-\pr@sphi);
2131 \coordinate (-edge D start) at (-\pr@cphi,-\pr@sphi);
2132 \coordinate (-edge D end) at (1-\pr@cphi,-\pr@sphi);
2133 \coordinate (-edge B2 start) at (1-\pr@cphi,-\pr@sphi);
2134 \coordinate (-edge B2 end) at (1,0);
2135 \end{scope}
2136 },
2137 boat/.style={
2138     every Penrose pic/.try,
2139     pic type=boat,
2140 },

```

The diamond.

```

2141 diamond/.pic={
2142     \begin{scope}
2143         \pgfkeysgetvalue{/tikz/Penrose/alignment location}{\prloc}
2144         \ifx\prloc\pgfutil@empty
2145             \else
2146                 \begin{group}
2147                     \tikzset{name prefix ..}%
2148                     \tikz@scan@one@point\pgfutil@firstofone%
2149                     (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} start)%
2150                     \global\pgf@xa=\pgf@x
2151                     \global\pgf@ya=\pgf@y
2152                     \tikz@scan@one@point\pgfutil@firstofone%
2153                     (\prloc-edge \pgfkeysvalueof{/tikz/Penrose/alignment edge} end)%
2154                     \global\pgf@xb=\pgf@x
2155                     \global\pgf@yb=\pgf@y
2156                 \end{group}
2157                 \advance\pgf@xb by -\pgf@xa
2158                 \advance\pgf@yb by -\pgf@ya
2159                 \pgftransformshift{\pgfpoint{\pgf@xa}{\pgf@ya}}%
2160                 \pgfpointnormalised{\pgfpoint{\pgf@xb}{\pgf@yb}}
2161                 \pgf@xb=\pgf@x
2162                 \pgf@yb=\pgf@y

```



```

2163 \pgftransformtriangle%
2164 {\pgfpoint{0pt}{0pt}}%
2165 {\pgfpoint{\pgf@xb}{\pgf@yb}}%
2166 {\pgfpoint{-\pgf@yb}{\pgf@xb}}%
2167 \test@edgealign{d}%
2168 \if@edgealign%
2169 \test@newedge{2}%
2170 \if@newedge
2171 \pgftransformshift{\pgfpoint{1cm}{0cm}}%
2172 \pgftransformrotate{-162}%
2173 \else
2174 \pgftransformrotate{-18}%
2175 \fi
2176 \else
2177 \test@newedge{2}%
2178 \if@newedge
2179 \pgftransformrotate{162}%
2180 \pgftransformshift{\pgfpoint{- 2 * \pr@chphi cm}{0cm}}%
2181 \else
2182 \pgftransformshift{\pgfpoint{- \pr@cphi cm}{-\pr@sphi cm}}%
2183 \pgftransformrotate{18}%
2184 \fi\fi
2185 \fi
2186 \UsePenroseTile[
2187   every Penrose tile clip/.try,
2188   every diamond clip/.try
2189 ]{diamond}
2190 \UsePenroseTile[
2191   every Penrose tile/.try,
2192   every diamond/.try,
2193   pic actions
2194 ]{diamond}
2195 \coordinate (-edge D1 start) at (0,0);
2196 \coordinate (-edge D1 end) at (\pr@chphi,\pr@shphi);
2197 \coordinate (-edge B1 start) at (\pr@chphi,\pr@shphi);
2198 \coordinate (-edge B1 end) at (2*\pr@chphi,0);
2199 \coordinate (-edge B2 start) at (2*\pr@chphi,0);
2200 \coordinate (-edge B2 end) at (\pr@chphi,-\pr@shphi);
2201 \coordinate (-edge D2 start) at (\pr@chphi,-\pr@shphi);
2202 \coordinate (-edge D2 end) at (0,0);
2203 \end{scope}
2204 },
2205 diamond/.style={
2206   every Penrose pic/.try,
2207   pic type=diamond,
2208 },
2209 }
2210 </package>

```