1 Introduction

This document presents examples of tiling patterns drawn from various historical sources of inspiration. Apart from the aesthetics of the results, the main interest in the METAPOST source is how each "tile" is painted; the filling of a patch is pretty much the same in all cases, using a rectangular or triangular grid.

2 Moorish

These designs are largely taken from the chapter on Moorish tiling patterns in Owen Jones' *Grammar of Ornament*, 1856. The section names are all mine.

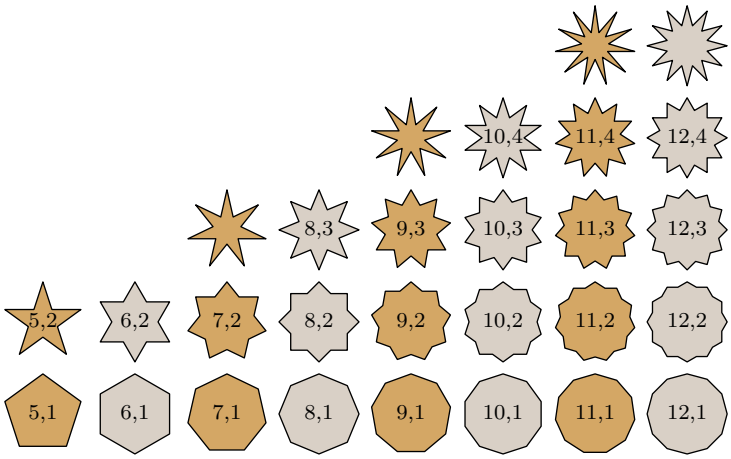
Each picture in this section runs in an METAPOST instance that inherits the definitions of these six ⟨color⟩ variables:

ivory navy gold forest lapiz plum

and this macro, which draws regular polygonal stars:

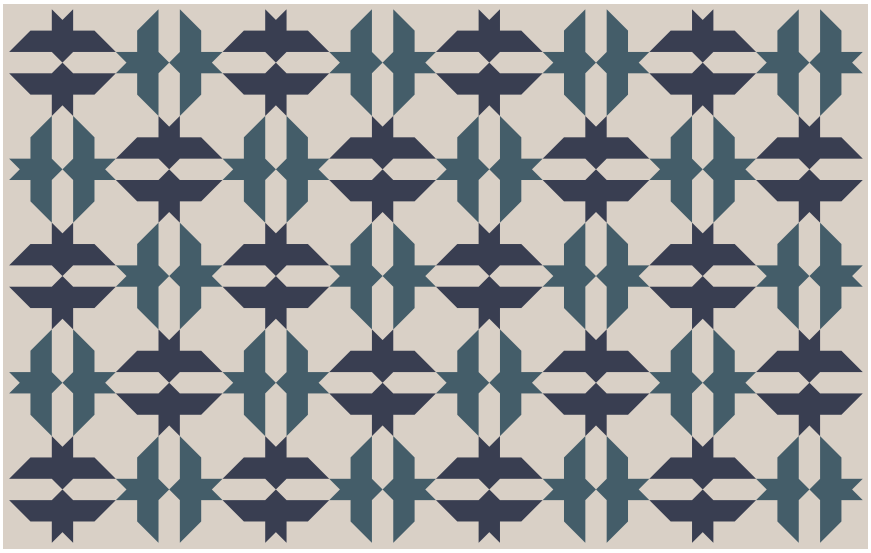
```
vardef star(expr n, s, r) =
  save a, b, t; path a, b; numeric t;
  a = (up -- up rotated (360/n*s)) scaled r;
  b = a rotated -(360/n*(s-1));
  (t,whatever) = a intersectiontimes b;
  for i=0 upto n-1: subpath (0,t) of a rotated (360/n*i)-- endfor cycle
enddef;
```

The parameter r is the radius of the star; n and s determine the shape, like this:



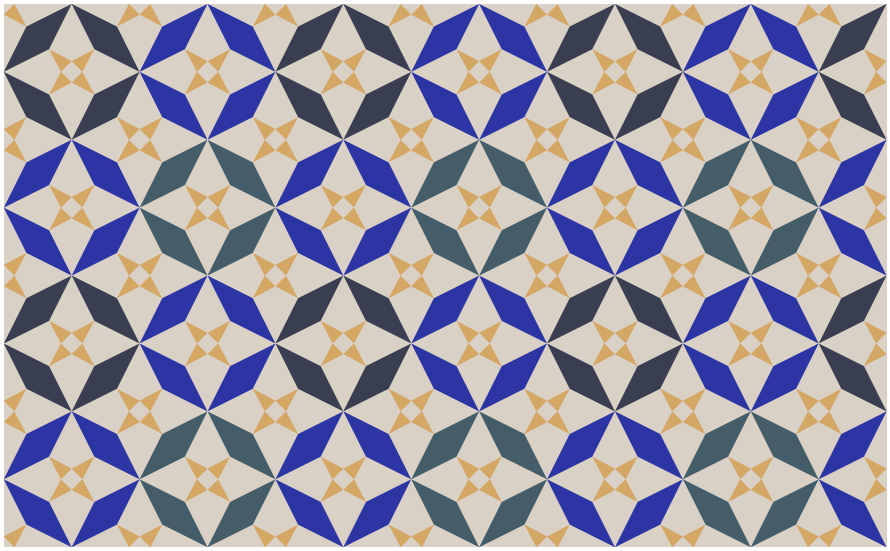
Most of the patterns in the following pages are laid out on a square grid, but the last three use a triangular grid.

Birds



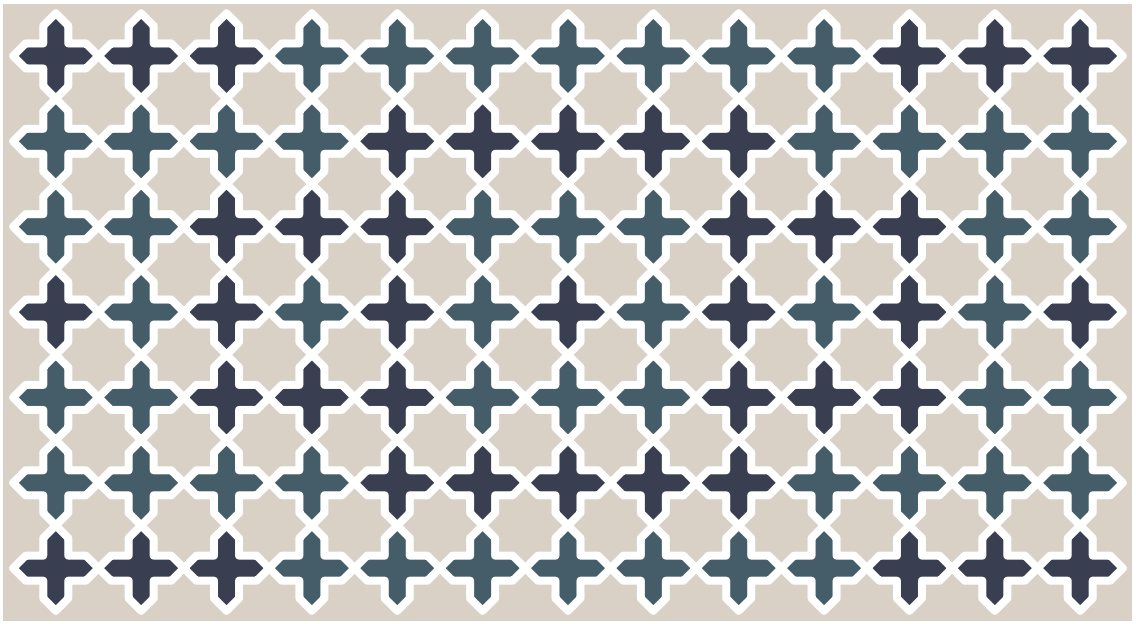
```
path bird;
bird = (0,0)--(1,1)--(5,1)--(3,3)--(1,3)--(1,5)--(0,4);
bird := bird -- subpath (5,1) of bird reflectedabout(up, down) -- cycle;

vardef unit(expr shade) = image(for t=0, 180:
  fill bird scaled 4 rotated t withcolor shade;
endfor) enddef;

picture tile[];
tile0 = unit(navy);
tile1 = unit(forest) rotated 90;

numeric u, v;
(u,v) = urcorner tile0 - llcorner tile0;

for i=1 upto 8:
  for j=1 upto 5:
    draw tile[(i+j) mod 2] shifted (i*u,j*v);
  endfor
endfor

picture p; p = currentpicture; fill bbox p withcolor ivory; draw p;
```

Carré



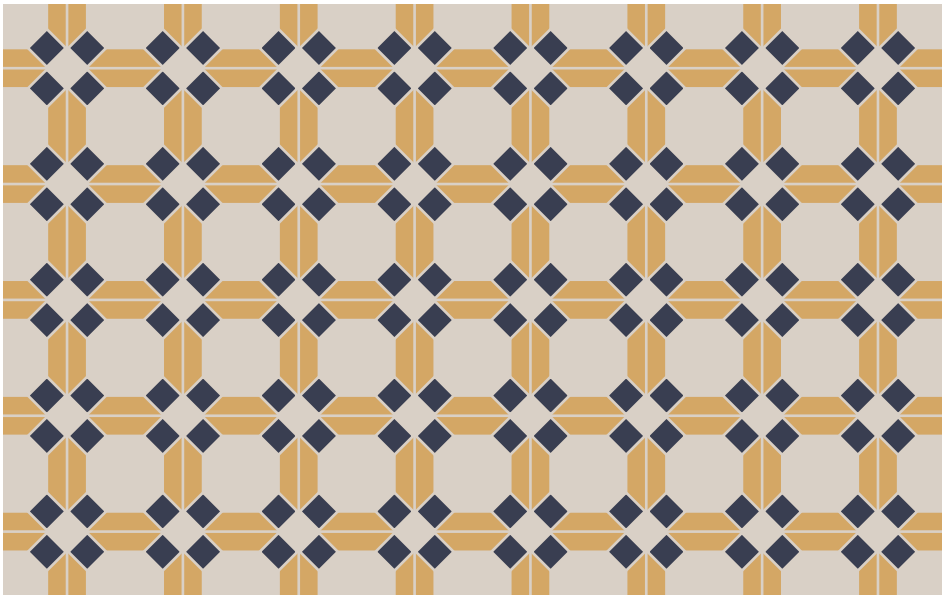
```
path lozenge, triangle, box;
lozenge = (left -- down -- right -- up -- cycle)
  yscaled 1/3 rotated -45 scaled 18;
interim bboxmargin := 0; box = bbox lozenge;
triangle = point 0 of box + 4 right -- point 1 of lozenge
        -- point 0 of box + 4 up -- cycle;

vardef paint_tile(expr s) = image(
  fill box withcolor ivory;
  fill triangle withcolor gold;
  fill triangle rotated 180 withcolor gold;
  fill lozenge withcolor s;
) enddef;

picture tile[];
tile0 = paint_tile(lapiz);   tile2 = paint_tile(navy);
tile1 = paint_tile(forest);  tile3 = paint_tile(lapiz);

numeric u, v;
(u, v) = urcorner tile0 - llcorner tile0;

for i = 0 upto 12:
  for j = 0 upto 7:
    t := (floor (i/2) mod 2) + 2(floor (j/2) mod 2);
    draw tile[t] if odd (i+j): rotated 90 fi shifted (i*u, j*v);
  endfor
endfor
```

Cross



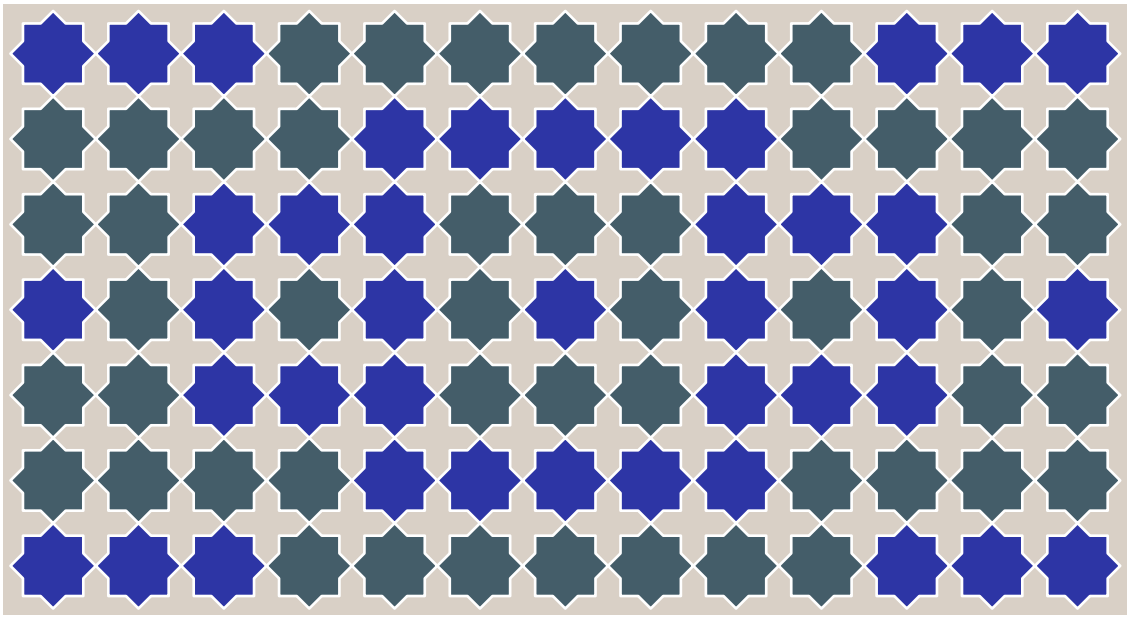
```
path cross;
numeric r, s, t; t = 2; s = sqrt(2); r = t - s;
cross = (for i=0 upto 3:
  ((t,0) -- (s, r) -- (r, r) -- (r, s)) rotated 90i --
endfor cycle) scaled 8;

pair u, v;
u = point 0 of cross - point 8 of cross;
v = point 4 of cross - point 12 of cross;

for i=-6 upto 6:
  for j=-3 upto 3:
    k := j + floor ((i + abs(j)) / (2 abs(j) + 1));
    fill cross shifted (i*u+j*v)
      withcolor if odd k: forest else: navy fi;
    draw cross shifted (i*u+j*v)
      withpen pencircle scaled 3 withcolor  white;
  endfor
endfor
picture P; P = currentpicture; fill bbox P withcolor ivory; draw P;
```

Lifebelt



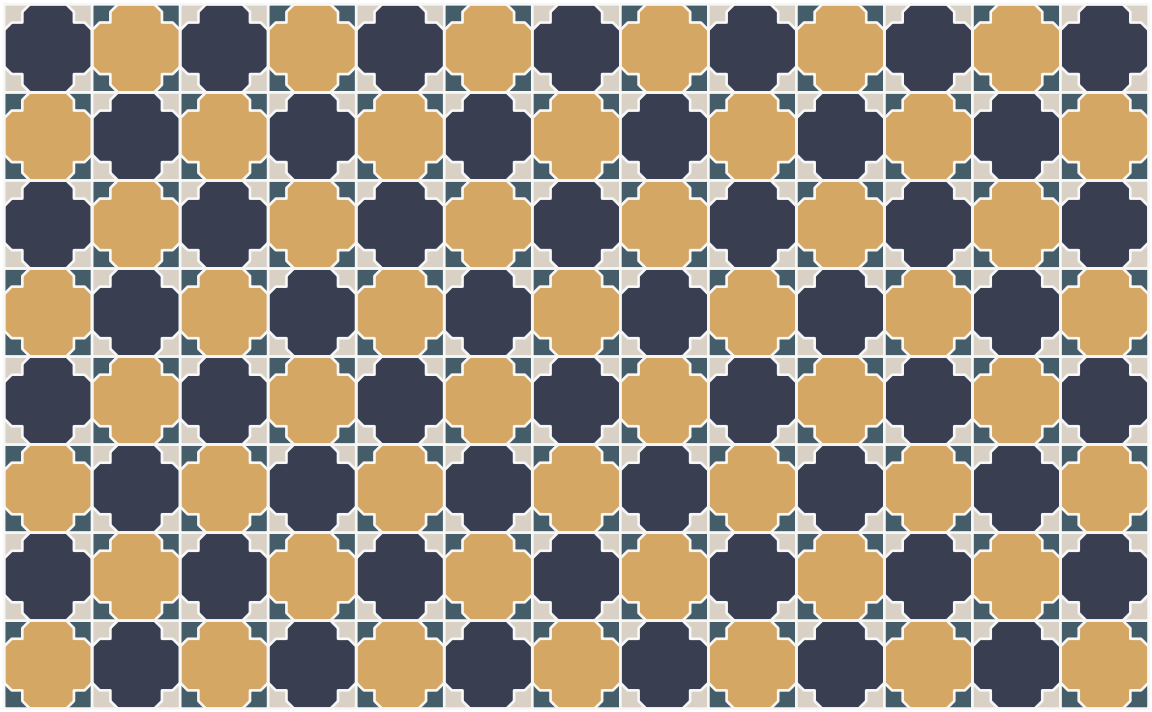
```
background := ivory;
numeric u; u = 10;
path s; s = unitsquare scaled u shifted 1/2(3u, -u) rotated 45;
path t; t = subpath (2,3) of s -- subpath (0,1) of s rotated 90 -- cycle;
picture unit; unit = image(
  pickup pencircle scaled 1;
  for i=0 upto 3:
    fill s rotated 90i withcolor navy;
    fill t rotated 90i withcolor gold;
  endfor
  for i=0 upto 3:
    undraw s rotated 90i;
    undraw subpath (1,2) of t rotated 90i;
  endfor
  pickup defaultpen;
);
numeric u, v; (u,v) = urcorner unit - llcorner unit;
for i=0 upto 8:
  for j= 0 upto 5:
    draw unit shifted (i*u, j*v);
  endfor
endfor
picture P; P = currentpicture; unfill bbox P; draw P;
begingroup;  interim bboxmargin := -1/2 u;
clip currentpicture to bbox currentpicture;
endgroup;
```

Octa-star



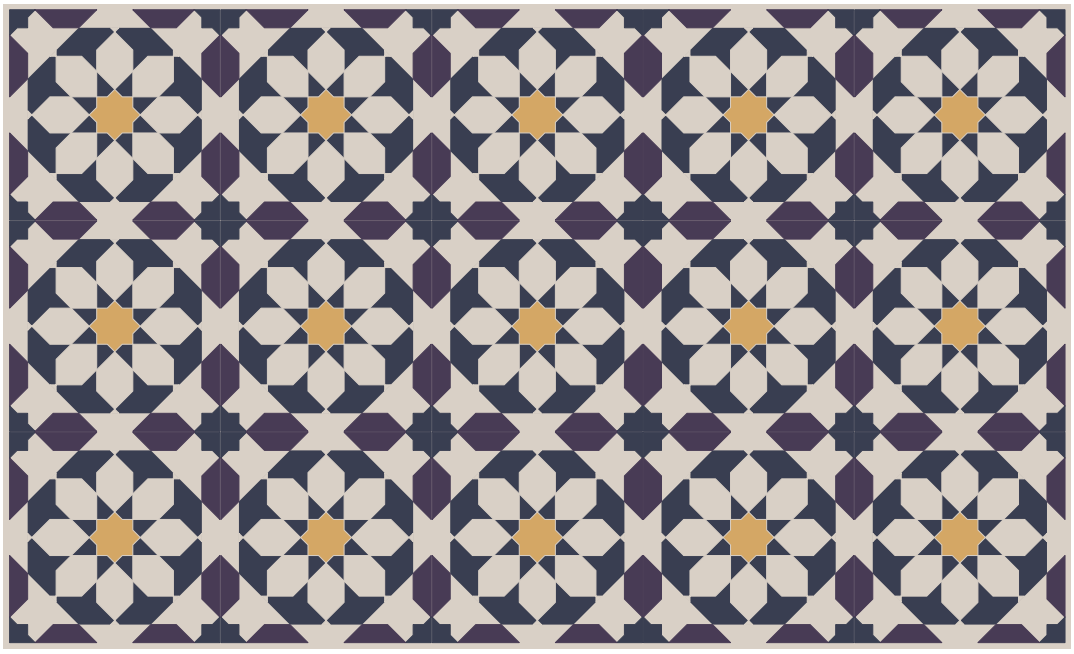
```
background := ivory;
path s; s = star(8,2,16);
numeric u, v; (u, v) = urcorner s - llcorner s;
for i=-6 upto 6:
  for j=-3 upto 3:
    pair w; w = (i*u, j*v);
    numeric k; k = j + floor ((i + abs j) / (1 + 2 abs j));
    fill s shifted w withcolor if odd k: forest else: lapiz fi;
    draw s shifted w withpen pencircle scaled 1 withcolor white;
  endfor
endfor
picture P; P = currentpicture; unfill bbox P; draw P;
```

Quad



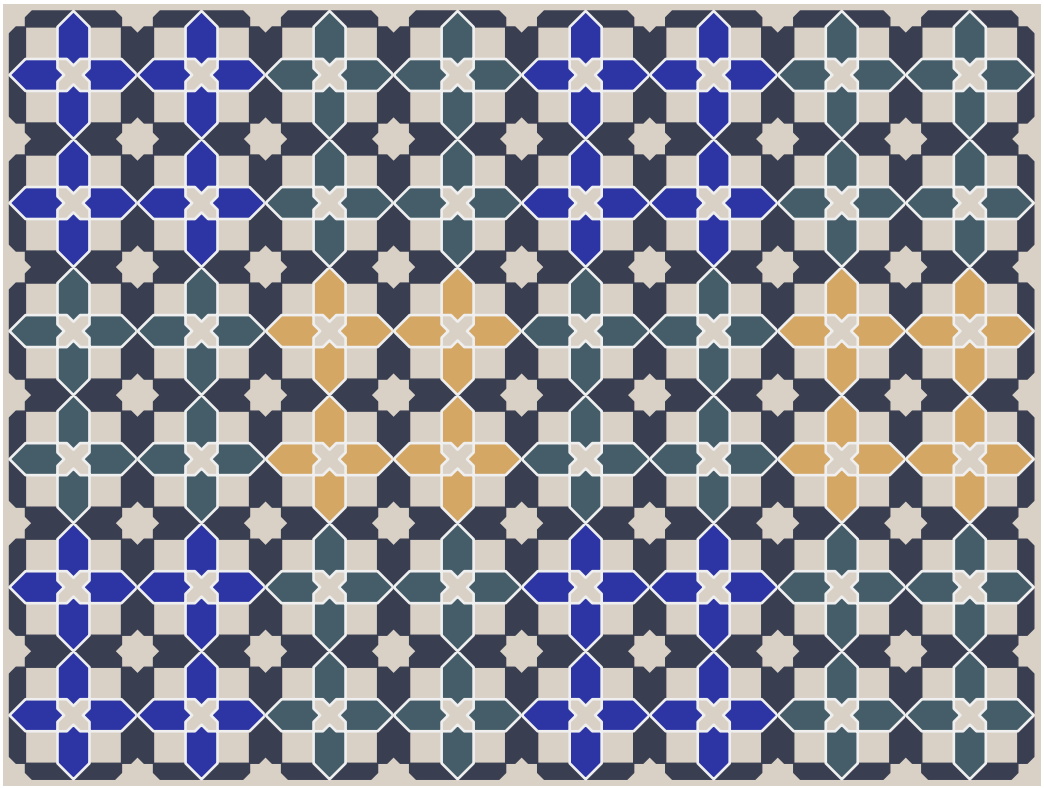
```
numeric u, r; u = 33; r = sqrt(1/2);
path s; s = unitsquare shifted -(1/2, 1/2) scaled u;
path o; o = for i=0 upto 7: point r of s rotated 45i -- endfor cycle;
path c; c = for i=0 upto 3:
  (subpath (0, 1-r) of o --
  point 0 of o rotatedabout(point 1 of s, -45) --
  subpath (r, 1) of o) rotated 90i --
endfor cycle;

vardef unit(expr fg, bg) = image(
  fill s withcolor bg;
  fill c withcolor fg;
  draw s withpen pencircle scaled 1 withcolor 31/32;
  draw c withpen pencircle scaled 1 withcolor 31/32;
) enddef;

for i=1 upto 13:
  for j=1 upto 8:
    draw if odd (i+j):
      unit(navy, ivory)
    else:
      unit(gold, forest)
    fi shifted (i*u, j*u);
  endfor
endfor
```

Rosette



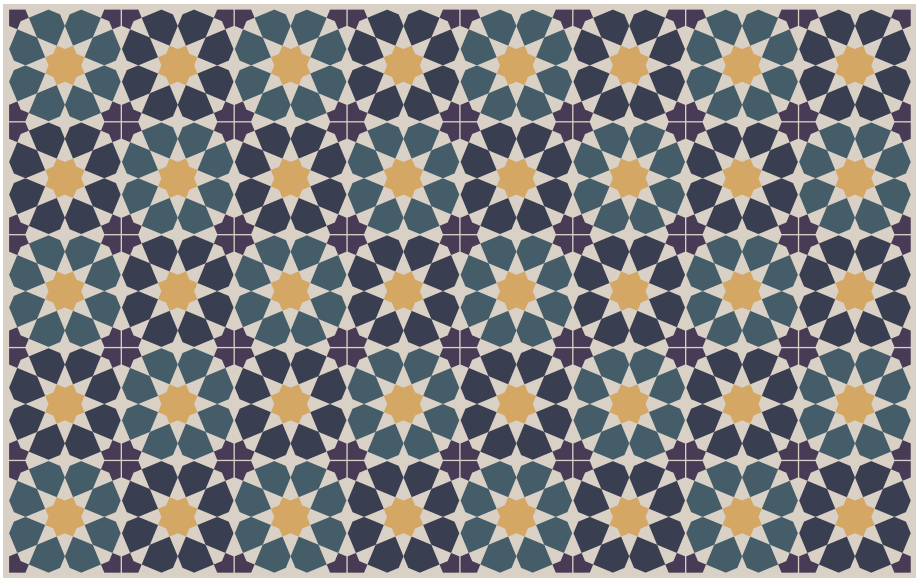
```
background := ivory;
path o; o = for i=0 upto 7: 48 dir (45i + 45/2) -- endfor cycle;
fill o withcolor navy;
numeric r, s; r = 3; s = 13; pair q; q = (s,s) scaled sqrt(1/2);
path t; t = (up -- origin -- right) scaled s;
path u; u = (down * s -- up * r -- right * r -- left * s);
path leaf; leaf = t shifted q --
                  u shifted (point 1/2 of o - 1/2(r,r)) -- cycle;
for i=0 upto 7:
  unfilldraw leaf rotated 45i withpen pencircle scaled 1/4;
endfor
path v; v = star(8,2,s); % common star macro
fill v withcolor gold; undraw v;
path w; w = (origin -- subpath(4,8) of v -- cycle)
            shifted (xpart point 0 of o, ypart point 1 of o) shifted q;
path p; p = point 0 of o -- point 5 of w shifted (q rotated 180) --
            point 5 of w -- point 0 of o shifted (q rotated -90) -- cycle;
for i=0 upto 3:
  filldraw w rotated 90i withcolor navy;
  filldraw p rotated 90i withcolor plum;
  filldraw p reflectedabout(left, right) rotated 90i withcolor plum;
endfor

picture unit; unit = currentpicture; clearit;
numeric u, v; (u,v) = urcorner unit - llcorner unit;
for i=1 upto 5: for j = 1 upto 3:
  draw unit shifted (i*u, j*v);
endfor endfor
picture P; P = currentpicture; clearit;
P := P scaled (396 / xpart (urcorner P - llcorner P));
unfill bbox P; draw P;
```

Scotch



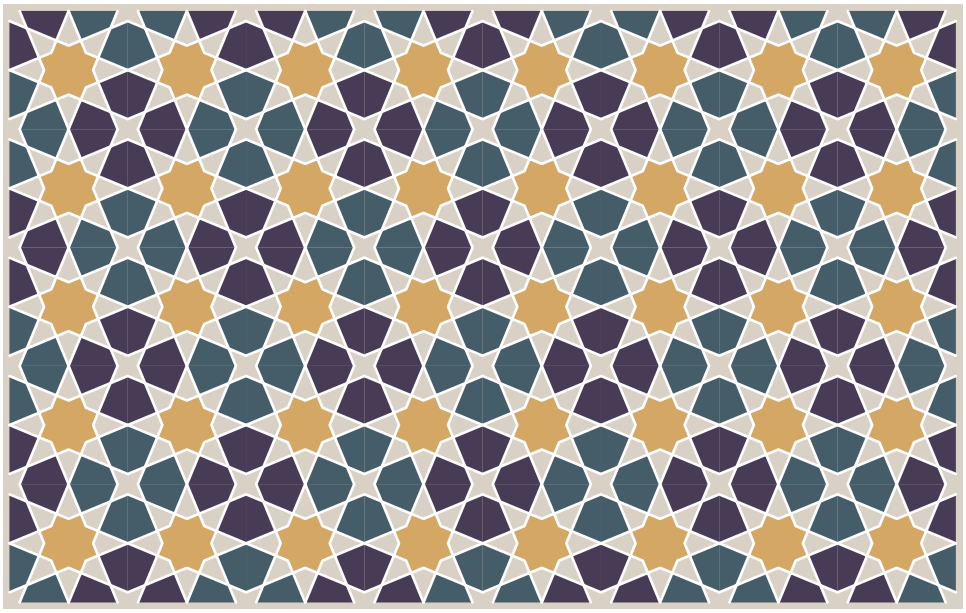
```
background := ivory;
numeric r; r = sqrt(2)-1;
path s; s = (0, r-1) -- (r, -1) -- (1, -1) -- (1, -3) -- (0, -4);
path t; t = (0, -4) -- (1, -3) -- (3, -3) -- (3, r-4) -- (3-r, -4);
s := s -- subpath (3,1) of s reflectedabout(up, down) -- cycle;
t := t -- subpath (4,1) of t reflectedabout(up, down) -- cycle;

numeric u; 4u = 24;
s := s scaled u;
t := t scaled u;

vardef unit(expr shade) = image(for i=0 upto 3:
  filldraw t rotated 90i withcolor navy;
  fill s rotated 90i withcolor shade;
  draw s rotated 90i withpen pencircle scaled 1 withcolor 15/16;
endfor) enddef;

for i=2 upto 9:
  for j=2 upto 7:
    color shade; shade =
    if odd floor(i/2): if odd floor(j/2): lapiz else: forest fi
                  else: if odd floor(j/2): forest else: gold fi fi;
    draw unit(shade) shifted ((i, j) scaled 8u);
  endfor
endfor

picture P; P = currentpicture;clearit; unfill bbox P; draw P;
```

Star-flower



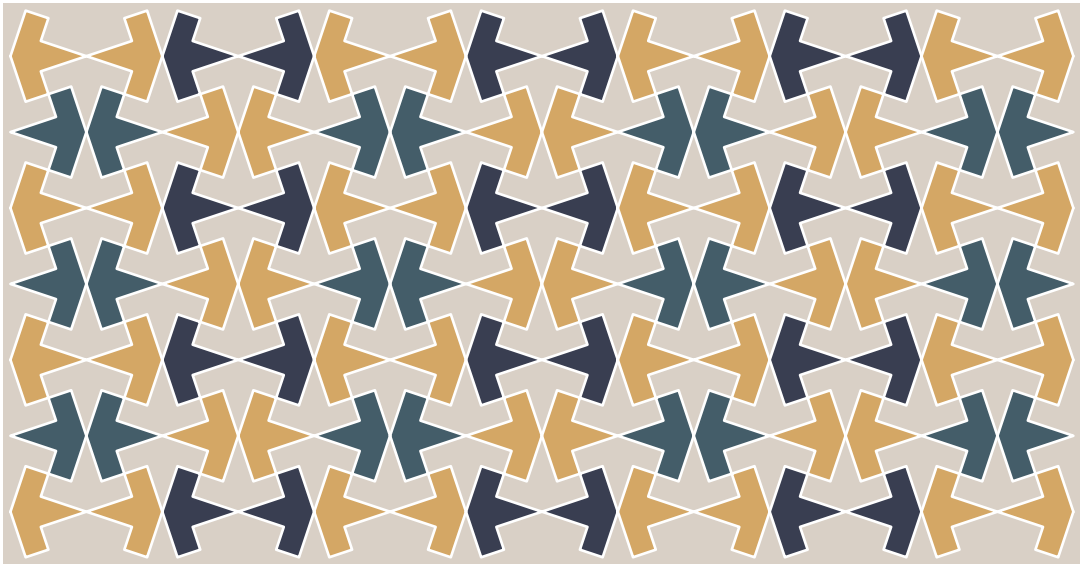
```
background := ivory;
path s, t, a, b; pair p, q;
s = star(8, 2, 8) rotated -45/2;  % the common star macro
p = whatever[point  4 of s, point  0 of s]
  = whatever[point 14 of s, point 10 of s];
t = (origin -- subpath (-3, 1) of s -- cycle) rotatedabout(p, 180);
q = whatever[point 2 of s, point 6 of s]
  = whatever[point 8 of s rotatedabout(p, 180),
                                    point 12 of s rotatedabout(p, 180)];
a = point 0 of s -- p -- point 2 of t -- p rotatedabout(point 0 of t, -45)
    -- q -- p rotated 45 -- cycle;
b = a reflectedabout(up, down);

picture tile[]; numeric n; n = -1;
forsuffixes $=forest, navy:
  tile[incr n] = image(
    fill s withcolor gold;
    for i=0 upto 3:
      fill t rotated 90i withcolor plum;
      fill a rotated 90i withcolor $;
      fill b rotated 90i withcolor $;
    endfor);
endfor

numeric u, v, grouting; grouting = 1/2;  % also try negative...
(u-grouting, v-grouting) = urcorner tile0 - llcorner tile0;
for i=1 upto 8:
  for j= 1 upto 5:
    draw tile[(i+j) mod 2] shifted (i*u, j*v);
  endfor
endfor
picture P; P = currentpicture; unfill bbox P; draw P;
```

Sun



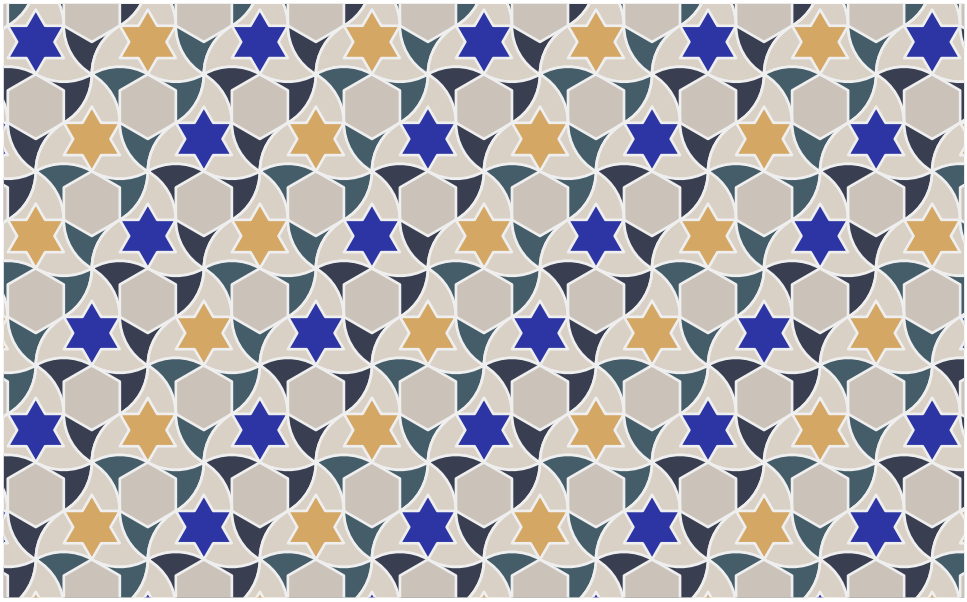
```
background := ivory;
path S, W, M; numeric a,b,c,d,e;
S = star(8, 2, 12) rotated 45/2;  % the common star macro
z0 = a[point 2 of S, point 14 of S] = b[point 4 of S, point 8 of S];
z1 = c[point 6 of S, point 10 of S] = d[point 4 of S, point 0 of S];
z2 = e[point 6 of S, point 10 of S]; x2=x0;
W = z2 -- z1 -- point 4 of S -- z0 -- cycle;
M = W reflectedabout(origin, dir 45);
picture tile; tile = image(
  fill S withcolor gold;
  for i=0 upto 1:
    fill W rotated 180i withcolor forest;
    fill M rotated 180i withcolor forest;
    fill W rotated (90+180i) withcolor plum;
    fill M rotated (90+180i) withcolor plum;
  endfor
  drawoptions(withpen pencircle scaled 1 withcolor white);
    draw S;
    for i=0 upto 3:
      draw subpath (0, 3) of W rotated 90i;
      draw subpath (0, 3) of M rotated 90i;
    endfor
  drawoptions();
);
numeric u, v, grouting; grouting = -1;
(u - grouting, v - grouting) = urcorner tile - llcorner tile;
for i = 1 upto 8:
  for j = 1 upto 5:
    draw tile if odd (i+j): rotated 90 fi shifted (i*u, j*v);
  endfor
endfor
picture p; p = currentpicture; unfill bbox p; draw p;
```

Tack



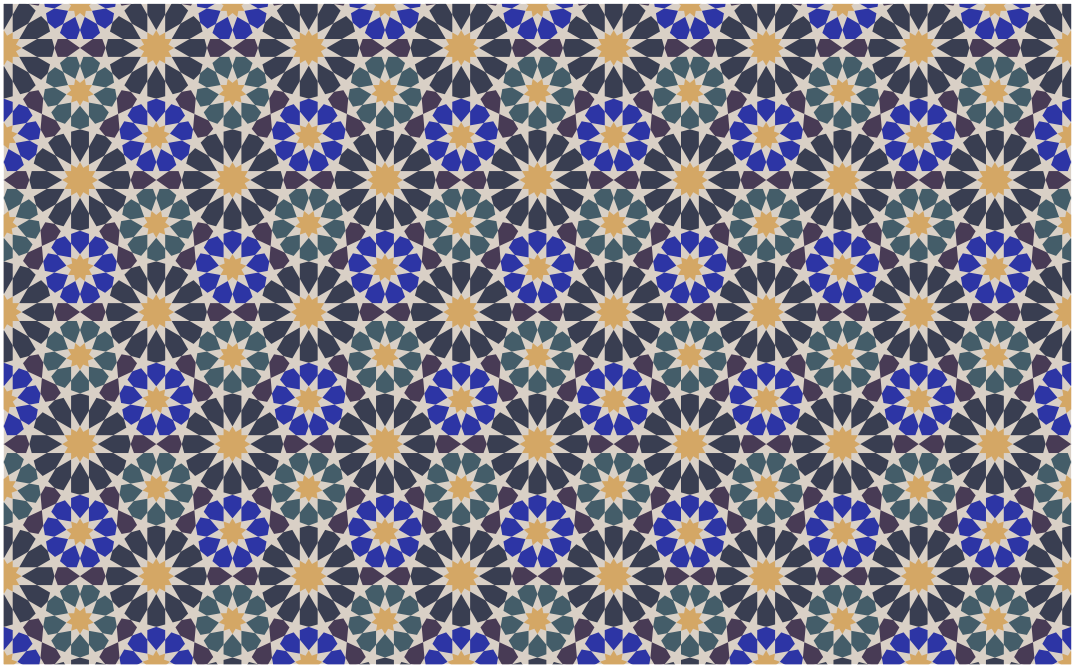
```
path tack;
tack = (0,0) -- (0,2) -- (1,2) -- (1,1) -- (3,1);
tack := tack rotated - angle point 4 of tack;
tack := tack shifted -1/2 point 4 of tack;
tack := tack -- subpath (3,0) of tack reflectedabout(left, right) -- cycle;
tack := tack scaled 9;

pair u, v; u = point 4 of tack - point 0 of tack; v = u rotated 90;

background := ivory;
for i=2 upto 15:
  for j=0 upto 6:
    path t; t = tack rotated (180 ((i+j) mod 2));
    pair w; w = i*u + j*v;
    fill t shifted w withcolor
      if odd floor(i/2): if odd j: forest else: gold fi
                    else: if odd j: gold   else: navy fi fi;
    draw t shifted w withpen pencircle scaled 1 withcolor white;
  endfor
endfor
picture P; P = currentpicture; clearit; unfill bbox P; draw P;
```

Hex-star



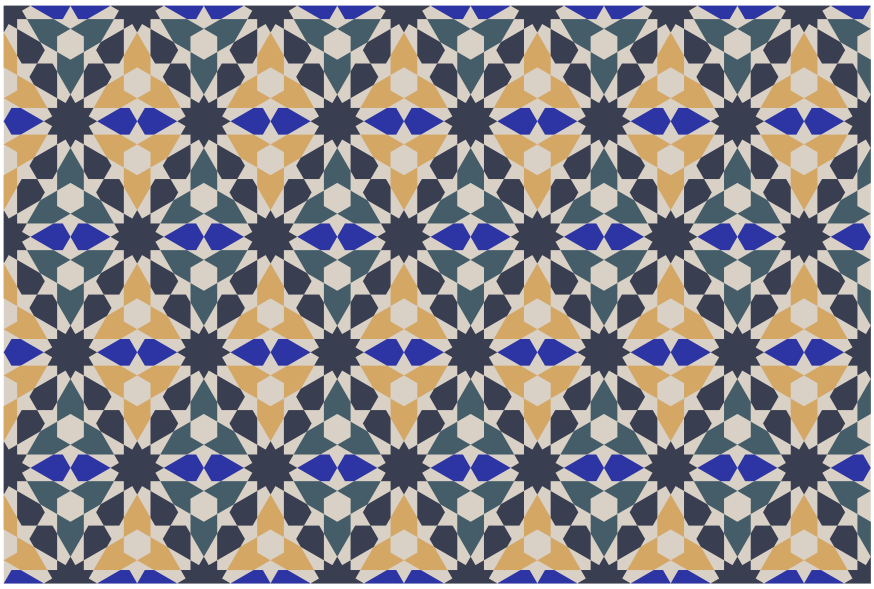
```
numeric r; r = sqrt(1/3);
path t; t = (r * dir   0 {dir  60} .. {dir 120} 2r * dir  60 {dir 240}
          .. r * dir 120 {dir 180} .. {dir 240} 2r * dir 180 {dir 360}
          .. r * dir 240 {dir 300} .. {dir 360} 2r * dir 300 {dir 120}
          .. cycle) rotated -30 scaled 21;
numeric a; (a, whatever) = t intersectiontimes (origin -- infinity * up);
path h; h = for i=0 upto 5: point a of t rotated 60i -- endfor cycle;
path t'; t' = t rotatedabout(point 1 of t, 60);
path h'; h' = h rotatedabout(point 1 of t, 60);
path s; s = for i=0 upto 5:
  point i of h' -- 1/3[point i of h', point i+2 of h'] --
endfor cycle;
vardef unit(expr a, b) = image(
  fill t withcolor a; fill h withcolor 15/16 ivory;
  fill t' withcolor ivory; fill s withcolor b;
  drawoptions(withpen pencircle scaled 1 withcolor 15/16);
    draw t; draw t'; draw h; draw s;
  drawoptions();
) enddef;
pair u, v; u = origin rotatedabout(point 1 of t, 120); v = u rotated -60;
numeric n; n = 6;
for i = -n upto n:
  for j = -n upto n:
    draw unit(if odd (i+j): navy else: forest fi,
      if odd (i+j): gold else: lapiz fi)
      shifted (i*u + j*v);
  endfor
endfor
clip currentpicture to unitsquare shifted -(1/2,1/2) scaled 5in yscaled 0.618;
```

Twelve-and-nines



```
numeric r[]; r0 = 19; r1 = 14; z0 = origin; z1 = (r0 + r1) * up;

path t, n, m; % twelve-gon, and nine-gons
t = (for i=0 upto 11: up rotated 30i -- endfor cycle) scaled r0 rotated -30;
n = (for i=0 upto 8: down rotated 40i -- endfor cycle) scaled r1 shifted z1;
m := n reflectedabout(origin, point 0 of t);
z2 = z1 reflectedabout(origin, point 0 of t);

path p; pair a, b, c, d, e;
a = whatever[point 1 of n, point 2 of n] = whatever[point 0 of t, point 1 of t];
b = whatever[point 1 of m, point 2 of m] = whatever[point 0 of t, point -1 of t];
c = whatever[a, b rotated 180] = whatever[b rotated 30, a rotated 210];
d = whatever[b, a rotated 180] = whatever[a rotated -30, b rotated 150];
e = whatever[b rotated 30, a rotated 210]
  = whatever[a rotated -30, b rotated 150];
p = (point 0 of t -- a -- c -- e -- d -- b -- cycle) rotated 30;
for i=1 upto 12:
  fill p rotated 30i withcolor navy;
endfor
fill star(12, 4, abs point 3 of p) withcolor gold;

path q, s; pair a, b, c, d, e;
a = whatever[point 0 of n, point 1 of n] = whatever[point -1 of p, point -2 of p];
b = whatever[point 0 of n, point -1 of n] = whatever[point 1 of p, point 2 of p];
c = whatever[a, b rotatedabout(z1, 180)]
  = whatever[b rotatedabout(z1, 40), a rotatedabout(z1, 220)];
d = whatever[b, a rotatedabout(z1, 180)]
  = whatever[a rotatedabout(z1, -40), b rotatedabout(z1, 140)];
e = whatever[b rotatedabout(z1, 40), a rotatedabout(z1, 220)]
  = whatever[a rotatedabout(z1, -40), b rotatedabout(z1, 140)];
q = point 0 of n -- a -- c -- e -- d -- b -- cycle;
s = q rotated -60;

for i=1 upto 9:
  fill q rotatedabout(z1, 40i) withcolor forest;
  fill s rotatedabout(z2, 40i) withcolor lapiz;
endfor;
fill star(9, 3, abs (point 3 of q - z1)) rotated 20 shifted z1 withcolor gold;
fill star(9, 3, abs (point 3 of q - z1)) shifted z2 withcolor gold;

path g; pair a;
a = whatever[point 1 of n, point 2 of m] = whatever[point 1 of m, point 2 of n];
g = subpath (-2, 2) of q rotated -30 -- a -- cycle;
for i=-1 upto 1: for r=0, 180:
  fill g rotatedabout(a, r) rotated 60i withcolor plum;
endfor endfor

picture unit; unit = currentpicture; clearit;
pair u, v; u = point 6 of m - point 3 of t; v = u rotated 60;
for i=0 upto 7: for j=0 upto 5:
  draw unit shifted (i * u + j * v - floor(j/2) * u) ;
endfor endfor

background := ivory; picture P; P = currentpicture; unfill bbox P; draw P;
clip currentpicture to unitsquare xscaled 7 xpart u yscaled 5 ypart v shifted z2;
```

Twelve-pointers



```
numeric u; u = 10;
path s; s = star(12, 4, u);
z0 = whatever[point 0 of s, point -8 of s]
   = whatever[point 2 of s, point 10 of s];
path t; t = star(12, 5, abs z0) rotated 15;
path h; h = star(6, 1, 2.5u) rotated 30;
numeric a, b; (a, b) = h intersectiontimes
                  subpath(0,-1) of t rotatedabout(point 0 of t, 180);
path p; p = subpath(0,2) of t -- subpath (6-a, a-6) of h -- cycle;
path r; r = (subpath(6-a, a-6) of h)
            reflectedabout(point 2-a/3 of h, point a/3-2 of h) rotated 30;
path q; q = subpath(2,4) of t -- r -- cycle;

vardef paint_tile(expr a, b) = image(
  fill s withcolor navy;
  for i=0 upto 5:
    fill p rotated 60i withcolor if i mod 3 = 1: lapiz else: navy fi;
    fill q rotated 60i withcolor if i mod 3 = 2: a else: b fi;
  endfor
) enddef;
picture rosette[];
rosette0 = paint_tile(forest, gold);
rosette1 = paint_tile(gold, forest);

pair u, v; u = lrcorner rosette1 - llcorner rosette1; v = u rotated 60;
for i=0 upto 7:
  for j=0 upto 5:
    draw rosette[j mod 2] shifted (i*u + j*v - floor(j/2)*u);
  endfor
endfor
picture P; P = currentpicture; fill bbox P withcolor ivory; draw P;
clip currentpicture to
  unitsquare xscaled 13/2 xpart u yscaled 5 ypart v shifted 1/2 u;
```