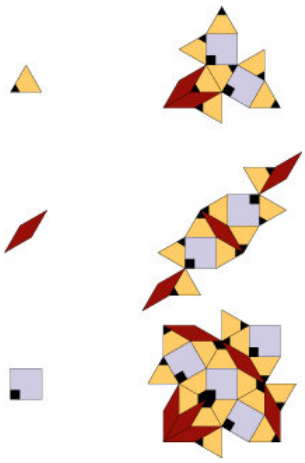
LITERATURE

Workflow: Watanabe Ito Soma 12-fold (variant)

by **tatasz** Watching

Published: Apr 10, 2016

Here, i'll be making the following tiling: **tilings.math.uni-bielefeld.de/...**.
The images mostly come from the Tiling Encyclopedia, with some notes from me.

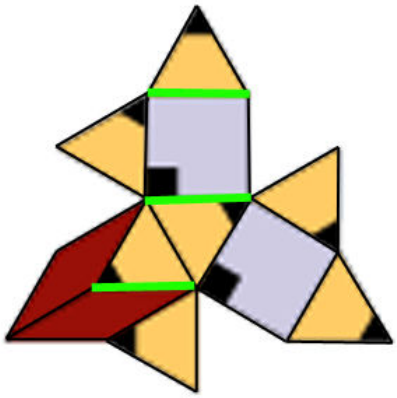


PS: this looks more complicated than it actually is

BASIC OVERVIEW

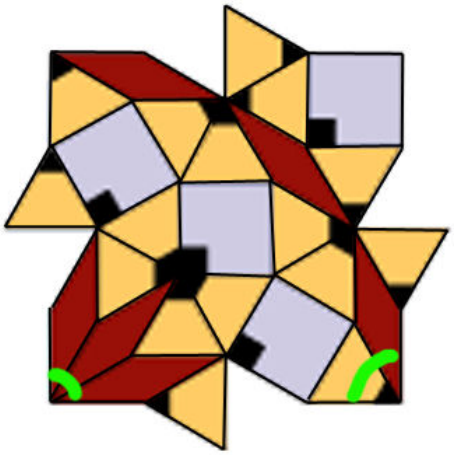
There are 3 types of tiles, triangle (T), rhomb (R) and square (S). They have specific orientations (so rotating a triangle 60 degrees will not result in the same triangle). Also, flipping the pieces either horizontally or vertically will break the pattern.

We also notice that the sides of all tiles are equal, because they share edges in the pattern. For example, below, look at the edges highlidghted in green:



Also, we can figure out the angles. It is pretty obvious in this case, but its always worth checking as what looks like 30 degrees on a picture may as well be some 26.56505
(yup, me and the kite-domino tile)

The rhomb: first, we know that the acutes angles are equal, and the obtuse angles are equal from definition of rhomb. Then, we also see that (below) 3 of the acute angles add up to 90, and an acute angle of the rhomb plus 60 degrees also adds up to 90. So we have that the acute angle is 30 degrees (90 / 3), and the obtuse angle is (it is a quadrilateral so all angles should addup to 360) equal to (360 - 30 * 2) / 2 = 150.



Last but not least we have 50 tile pieces:

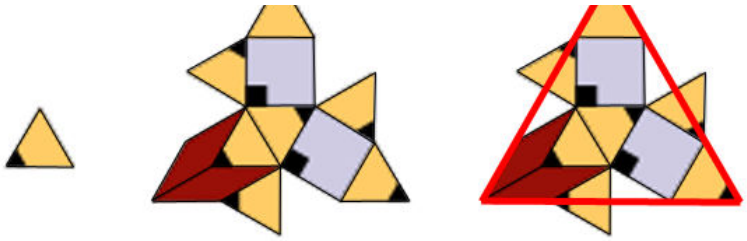
31 triangles
11 rhombs
8 squares

And each tile piece is made of:

Triangle: 7 triangles, 2 rhombs and 2 squares
Rhomb: 8 triangles, 3 rhombs and 2 squares
Square: 16 triangles, 6 rhombs and 4 squares

SCALE

The first step is to calculate the scale. There is no info on that in the Tiling Encyclopedia, so we may need to figure this out also (after some tilings, you can usually guess just from looking). Take a look at the picture below:



The small orange triangle is the scaled down version of the big triangle (marked in red). So the side of the small triangle, s, is equal to the side of the big triangle, S, multiplied by the scale factor:

s = S * p (formula 1)

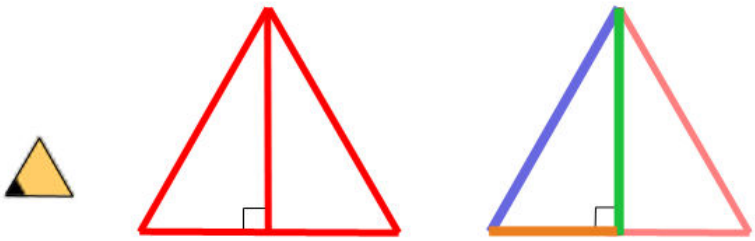
On the other hand, as the big triangle is tiled with smaller pieces, we can see that its side is equal to the sum of following elements:

side of the small triangle
twice the height of the small triangle
side of the rhomb (which is equal to the side of the small triangle

So:

S = s + 2 * (height) + s (formula 2)

Let's quickly find the height:



Lets use Pythagorean Theorem. The length of the blue line is s, as it is the side of the small triangle. The length of the orange line is half s. So we have:

(height)^2 + (0.5 * s)^2 = s^2
(height)^2 = s^2 - 0.25 * s^2
(height)^2 = 0.75 * s^2
height = s * sqrt(0.75)
height = s * sqrt(3) / 2 (formula 3)

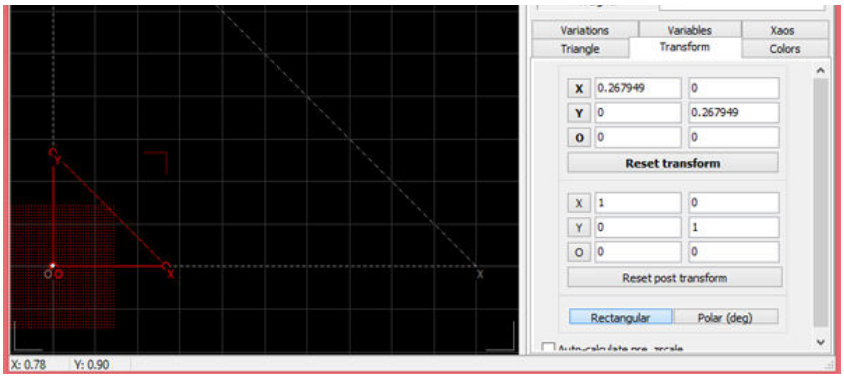
Putting formulas 1, 2 and 3 together, we have:

s = S * p (from formula 1)
s = (2 * s + 2 * height) * p (using formula 2 to remove the S value)
s = (2 * s + 2 * s * sqrt(3) / 2) * p (using formula 3 for height)
s = s * (2 + sqrt(3)) * p
p = 1 / (2 + sqrt(3))
p = 0.267949

So now we know that we need to scale all the transforms to 26.794919 % of their original size.

LET'S OPEN APO

Now we finally can do something in Apo.



And then duplicate x49 to get all the 50 transforms we need, one for each piece.

WEIGHTS

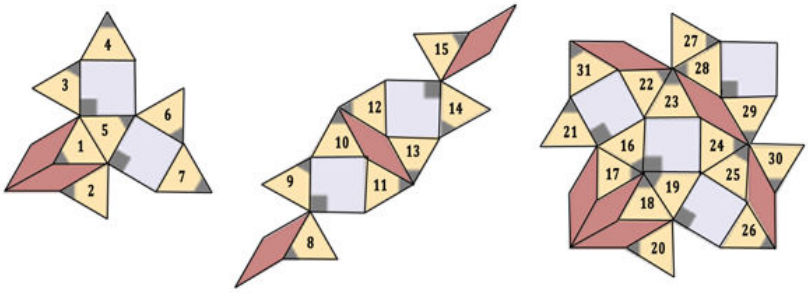
I usually set up the weights first. This helps to later position the tiles: as it is closer to the final result, you can more easily see if you positioned something wrong, or got a wrong rotation.

Remember that we have the following pieces:

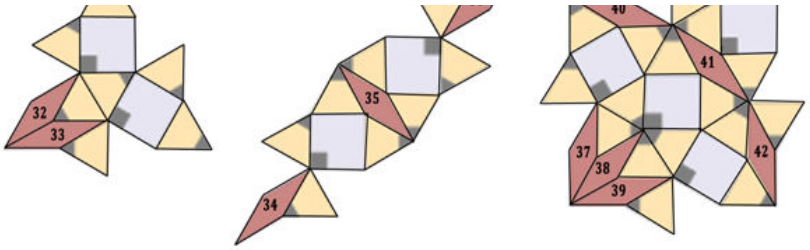
- 31 triangles
- 11 rhombs
- 8 squares

Below, I numbered them so we have a reference:

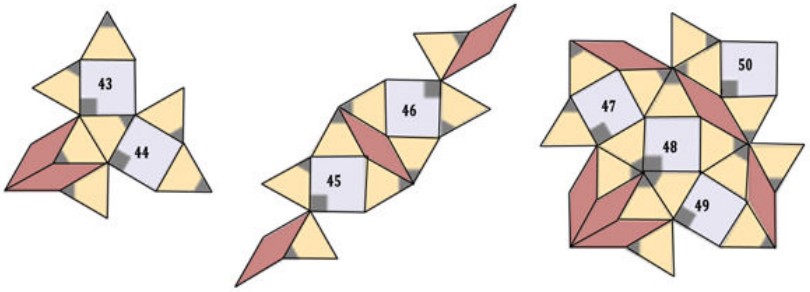
Triangles



Rhombs:



Squares



Now, lets adopt a following convention. If a piece is a triangle, I will write "from T", and then same for rhombs ("from R") and for squares ("from S"). Then, if a piece is part of a triangle, I will write "to T", same for rhombs ("to R") and for squares("to S"). If you write it down or just picture in your head, it shall be easy to set up all the xaos values.

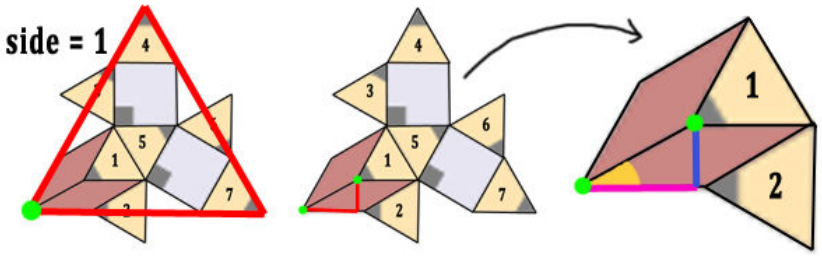
Transforms 1 - 7, 32, 33, 43, 44 (which are the pieces of the triangle) will have a "to T" on them, so they will only send points to transforms 1 - 31 (triangle shaped pieces)
Transforms 8 - 15, 34 - 36, 45, 46 will have a "to R" on them.
The other transforms will have a "to S" on them, sending points to square transforms only.

Now thats a lot of xaos values to set. I usually do it in notepad, as doing it in apo sucks big time.
The chaos = " ... " part of the code holds the "to" weights, so if a transform sends points to transforms 1, 2 and 3 but not to 4, 5 and 6, it will look like chaos = "1 1 1 0 0 0".
The way I ordered the trasnforms helps the hacking, as i have the "to" weights in big chucks, making it easy to type and check.

Also, as the tiles may overlap, we need to set some opacities to 0.
I will leave the big square visible, and turn off everything else: so set opacity for transforms 1 - 15, 32 - 36 and 43 - 46 should be 0.

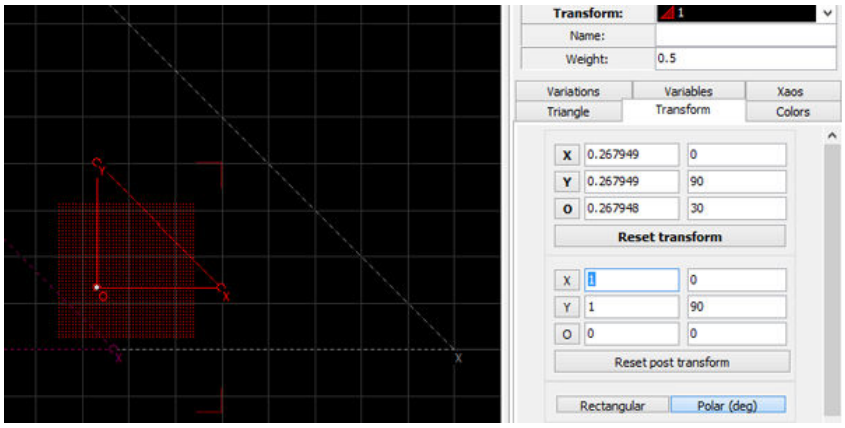
POSITIONS AND ROTATIONS

Lets now calculate some positions and rotations. First, i'll set the reference: the origin will be the left bottom corner of the big triangle, and its side will be equal to 1. So the sides of the small pieces are all equal to 1 / (2 + sqrt(3)) = 0.267949.



Tansform 1. Compare transform 1 with the original shape: the black thingy is in the bottom left corner for both, so there is no rotation necessary.
Now, we need to find the shift. We know that the acute angle (orange) is equal to 30 degrees, and that the sides of the rhomb are all equal to 0.267949, which means the distance between the two green dots on the middle and right pics is equal to 0.267949.
Now, we need to find the length of the purple (horizontal shift) and blue (vertical shift) lines.

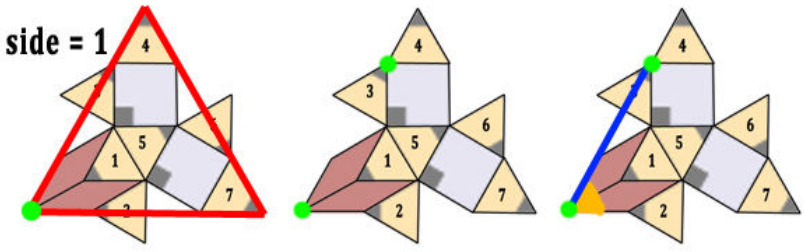
In this case, there is an Apo trick that makes it easier: Polar Coordinates mode. In this mode, the position is not represented like horizontal and vertical shifts, but as angle



Transform 2: transform 2 is just shifted horizontally – the shift is equal to the side of the rhomb, 0.267949. It is also rotated by 30 CCW (compare it with the original tile to see this).

Transform 3: once again, we will use the polar coordinates trick to position this one. Notice that the angle is 60 degrees (twice the acute angle of the rhomb), and the distance is equal to:

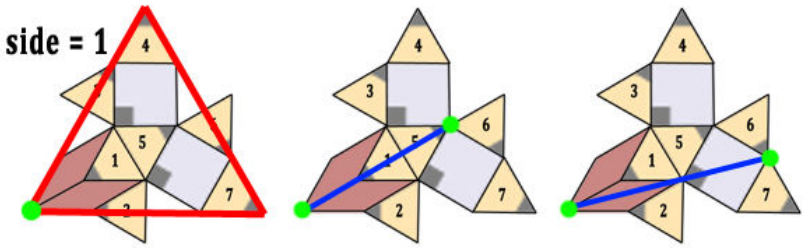
(side of the rhomb) + 2 * (height of the triangle)
0.267949 + 2 * 0.267949 * sqrt(3) / 2
(replacing values from previous calculations)
0.267949 * (1 + sqrt(3))
0.732051



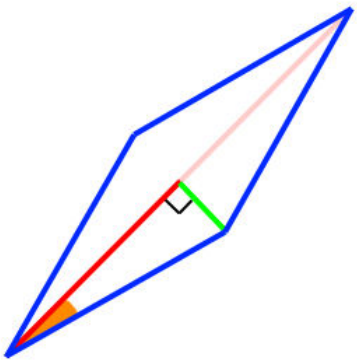
Now, lets look at the rotation of the tile: 150 degrees CW.

Transform 4: similarly to transform 3, you can use polar coordinates to position it. The angle is also 60 degrees, and the distance is 1 unit.
The rotation is 120 degrees CW. Can you see why?

Transform 5: notice that its position is similar to one of the transform 3, only the angle is different – now 30 degrees. The rotation is 180 degrees CW.



Transform 6: this one is a bit trickier. You can see that the angle for polar coords is 15 degrees (half of the acute angle of the rhomb). The distance is the sum of two diagonals, one of the rhomb, and one of the square.



On the picture above, the line in red is equal to half of the diagonal of the rhomb. The side of the rhomb, as we calculated before, is equal to 0.267949, and the orange angle is 30 / 2 = 15 degrees. Lets calculate the length of the red line:

cos(15) = (red) / 0.267949
(red) = 0.267949 * cos(15)
(red) = 0.258819

d^2 = 0.258819^2 + 0.258819^2
d^2 = 2 * 0.258819^2
d = sqrt(2) * 0.258819
d = 0.378937

So, we can calculate the distance for transform 6:

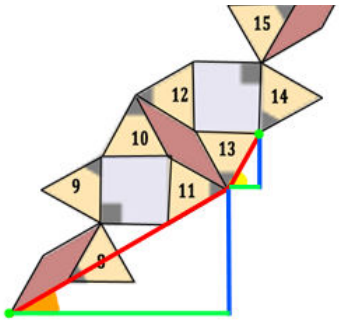
(rhomb diagonal) + (square diagonal) =
2 * 0.258819 + 0.378937=
0.896575

So we have: angle equal to 15 degrees and distance equal to 0.896575 for the polar coords, and the transform rotation is 60 degrees CCW.

Lets skip a few transform which are pretty straightforward, and go to:

Transform 14: As you can see below, the horizontal shift of this transform is given by the sum of lengths of the green lines, and the vertical, by the sum of length of the blue lines. Also, the orange angle is equal to 30, and the yellow, to 60 degrees.
From definition of cosine and sine, we know that:

cos(angle) = (green) / hypotenuse
sin(angle) = (blue) / hypotenuse



Last but not least, notice that the length of the big red line is 1 (as it is the side of the big rhomb), and the length of the short red line is 0.267949, as it is the side of a tile piece.

So lets calculate the lengths:

(green) = cos(30) * 1 + cos(60) * 0.267949
(green) = 0.866025 + 0.1339745
(green) = 1
(blue) = sin(30) * 1 + sin(60) * 0.267949
(blue) = 0.5 + 0.232051
(blue) = 0.732051

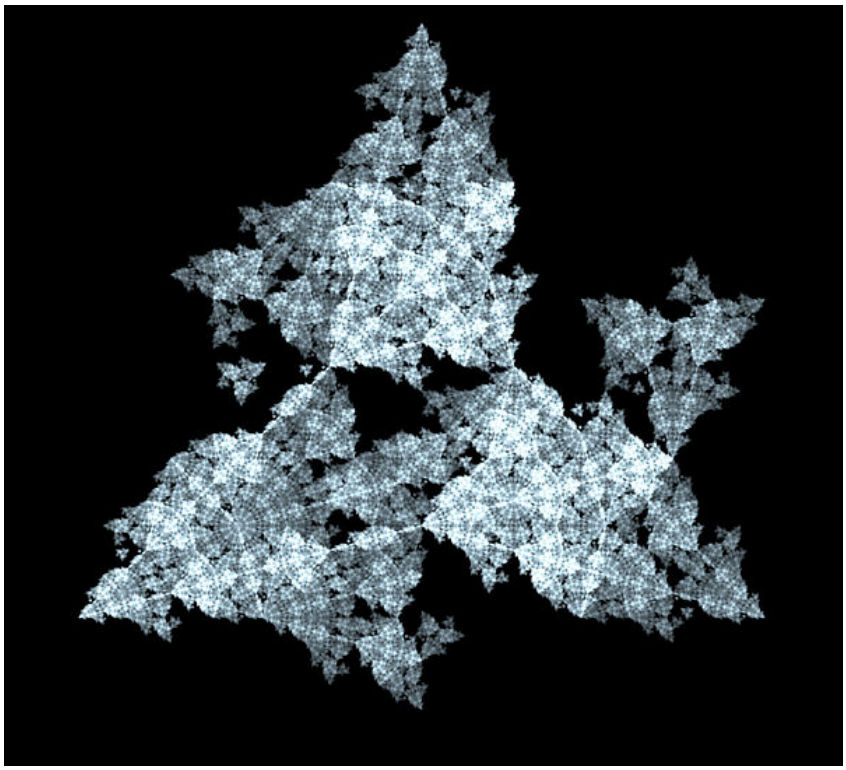
And just go on... Honestly, placing stuff is faster than typing it out

LAST CHECKS

Then, as always, I have some messed up tiles. To find those, i do the following:

1. Set the opacity of all transforms to 0
2. Set the opacity of transforms that make a certain big element (for example, all transforms that make the big triangle)
3. Compare the result with the original tiling to figure out where the mistake is
4. Fix the bugs

First, i look at the triangle:

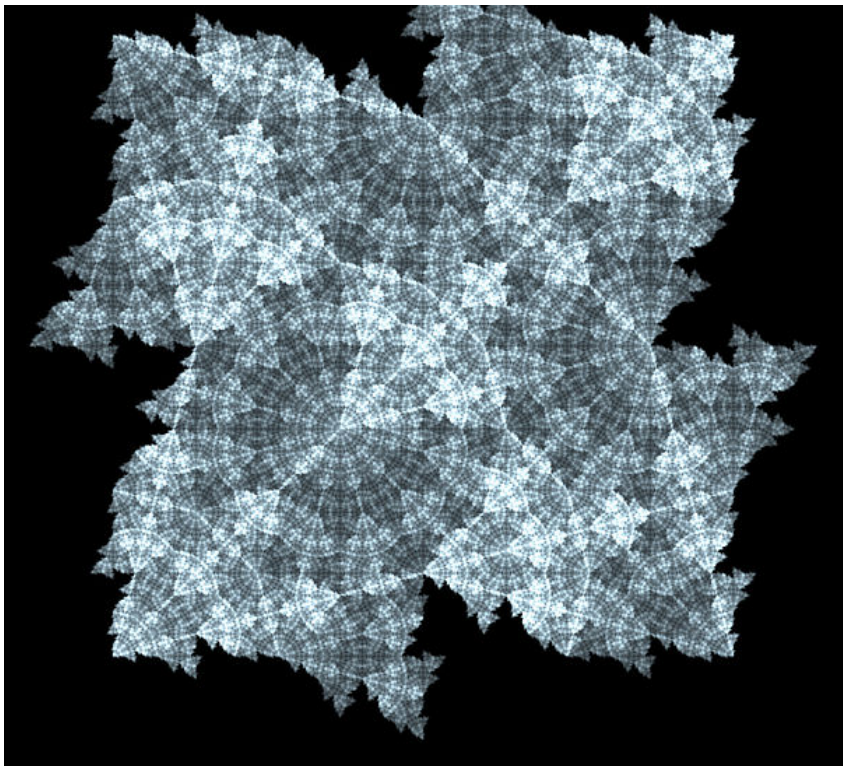


Appearently, i got the rotations of transforms 5 and 6 wrong. Compare with the original pic to see how i figured that out.

Now, lets look at the rhomb:

The position of transform 15 seems messed up. In fact, i got wrong the angle and it ended up in a wrong place

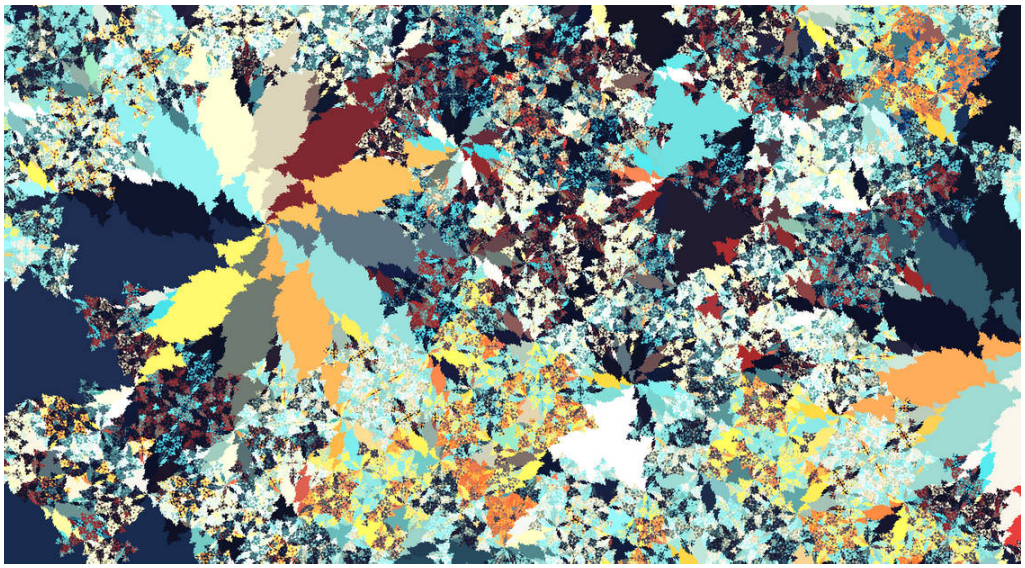
Finally, lets check the square:



Perfect

WEIGHTS

As i am lazy to properly calculate the correct weight fixes, i just set each transform weight igual to the number of transforms it sends point to. So if a transform has "to" weights values 1 to 31 transforms, I set its weight to 31 and so on.



linear substitution tiling workflow apophysis

Add to Favourites | Comment

COMMENTS 13 **OLDEST**

Add a new comment...

piethein21 Apr 10, 2016

I never ever did anything like this definitely will try

Reply

tatasz Apr 10, 2016

it is fun, and its a lot easier than it looks like

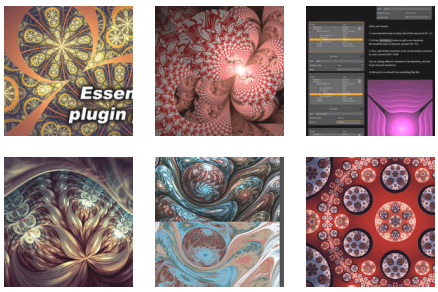
Reply

teundenouden Apr 10, 2016

I've made this one myself a long time ago but I didn't like it because it looks like someone took some bites out of the prototiles. It was quite a pity because this tiling obviously was a lot of work :S
The overlapping robinson tiling also gives weird effects.

Reply

More by tatasz Watching



colour

Reply

Lupsiberg Apr 12, 2016

Very interesting! Thank you very much for sharing 

Reply

Esherymack CORE Apr 23, 2016

holy *crap* this is cool
I'm uh, struggling through it to see if I'm even capable of it, but seeing as it's 1:40 AM and I'm stupidly sick, I'm putting it off until I can get a notebook out, but I'm up to the point of positioning things, so we'll see what I can do, I guess

Reply

dark-beam Jun 8, 2016

Too complicated

Reply

pillemaster CORE Feb 21, 2017

hint: try to tile the square with post transforms^^

Reply

pillemaster CORE Feb 21, 2017

...or even rhomb or triangle^^

Reply

tatasz Mar 17, 2017

Nah, i can do it with camera transforms now
Which i shall do for sure

Reply

pillemaster CORE Mar 18, 2017

that was what i meant with "post transforms"

Reply

tatasz Mar 21, 2017

Now to tile the triangle or the rhomb, ill need a few extra tools.
hmmmm so like, rotate by angle and duplicate.

Reply

pillemaster CORE Mar 21, 2017

i see, but the rhomb should fit in a rectangular pattern

Reply