recTiles

Julia Schedler

2023-07-10

Table of contents

Tiles **3**

Julia's First Tiles **4**

Hand-drawn 4

First Tile 4

Second Tile 5

Third Tile 7

Computer-generated 9

Fourth Tile 9

Fifth Tile 10

Tiles

This is Julia's transcription of what Alex has been showing her so far about his tile stuff.

Julia's First Tiles

Hand-drawn

First Tile

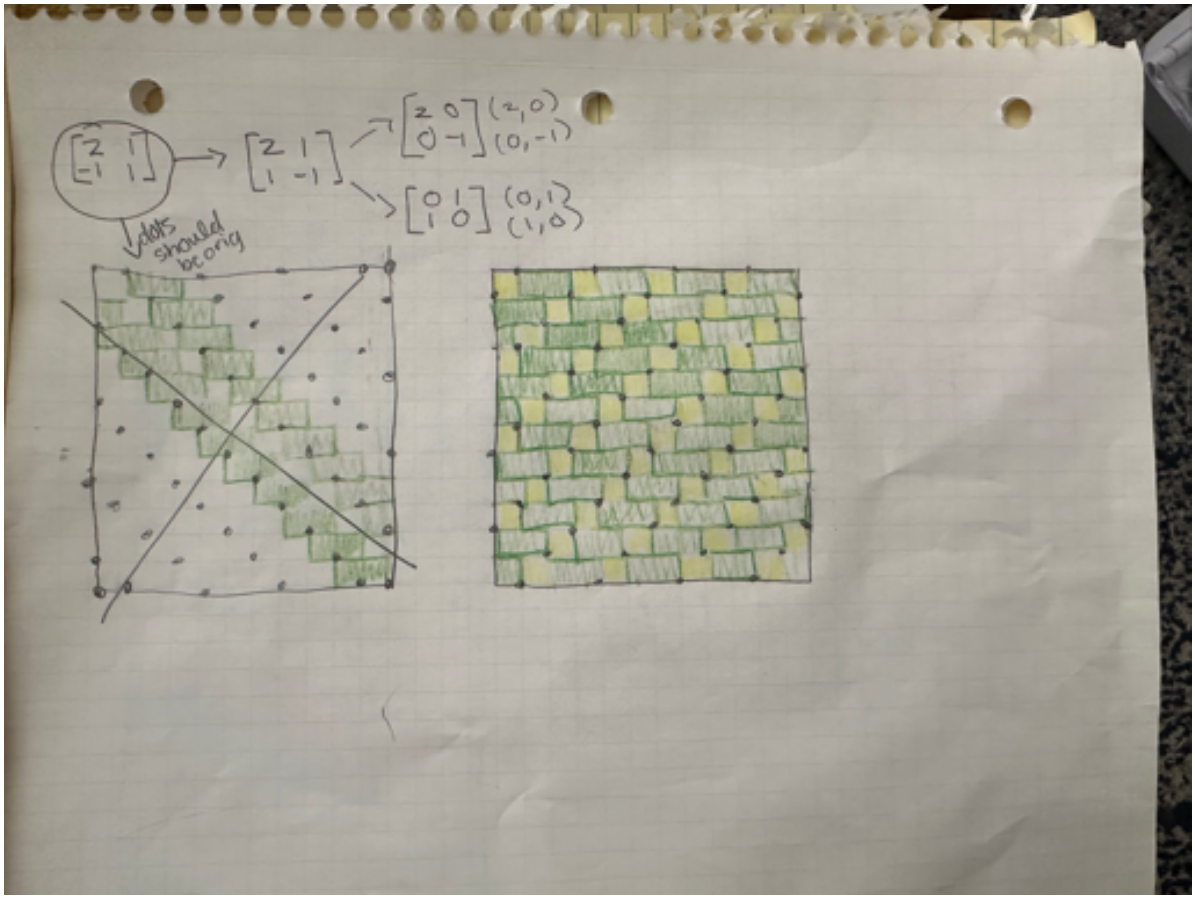
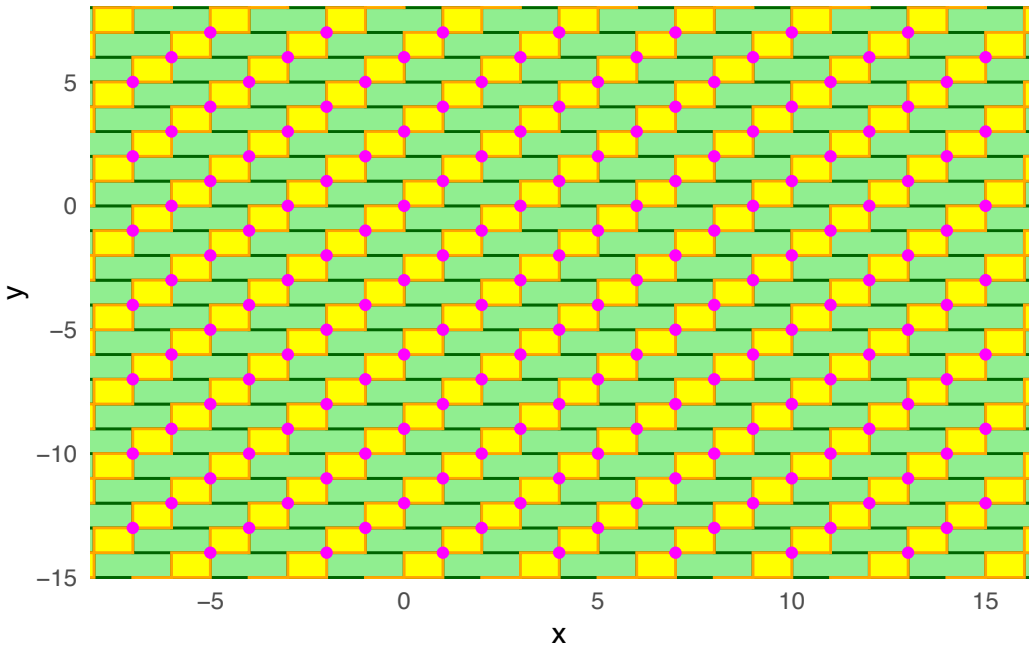


Figure 1: Actually I did this one second, but this is the first matrix I chose. Alex suggested I change to the one visualized next.

$$M = \begin{bmatrix} 2 & 1 \\ -1 & 1 \end{bmatrix}$$



Second Tile

$$M = \begin{bmatrix} 2 & 5 \\ -1 & 3 \end{bmatrix}$$

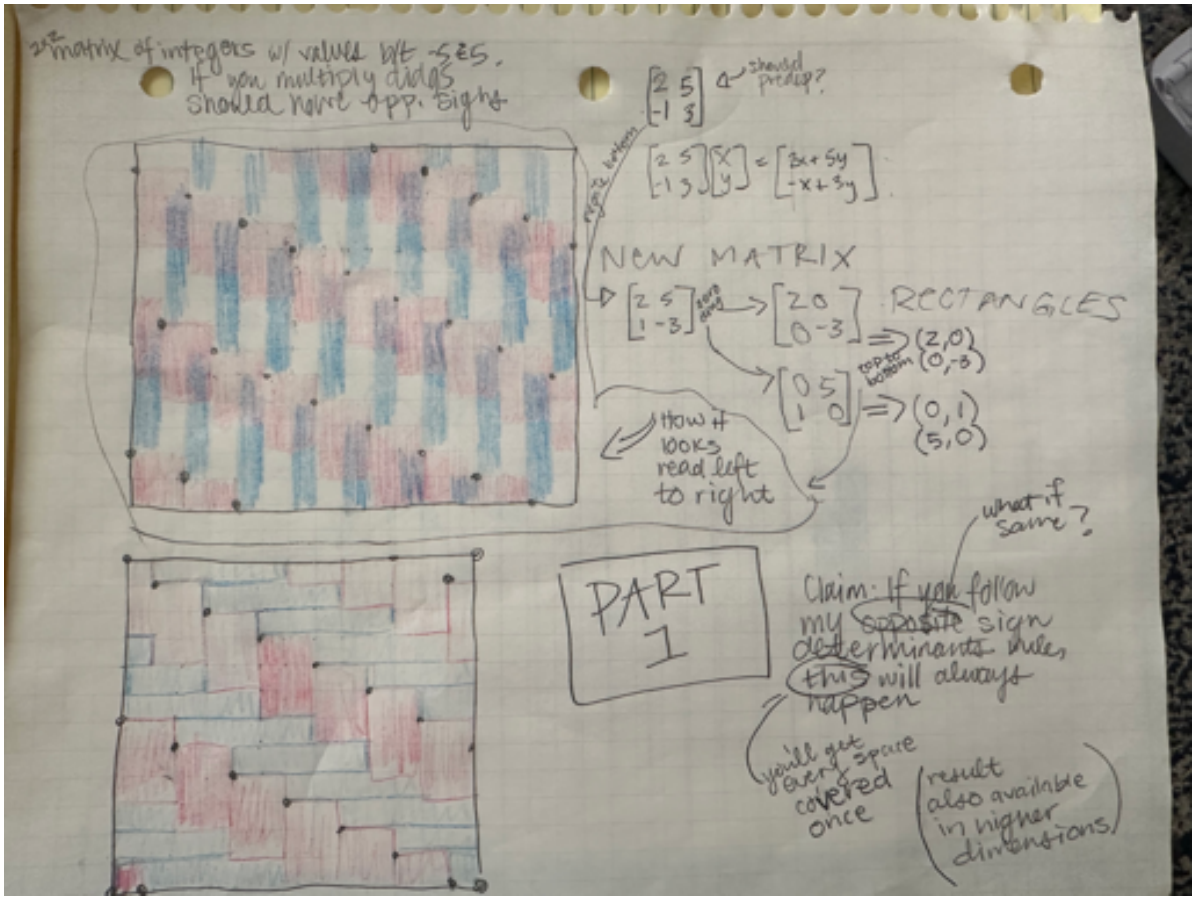
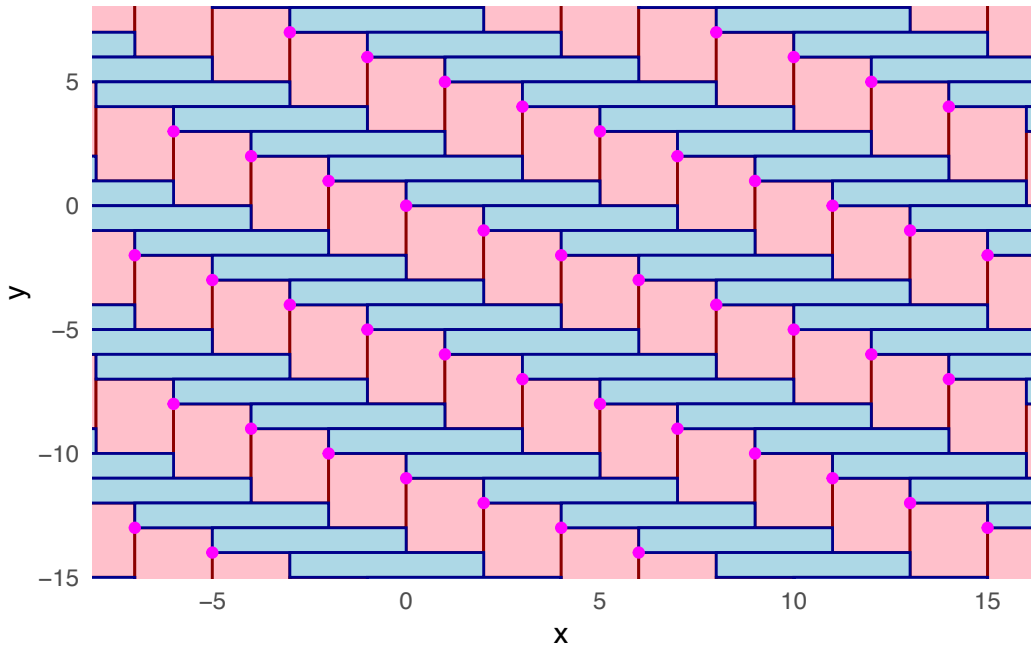
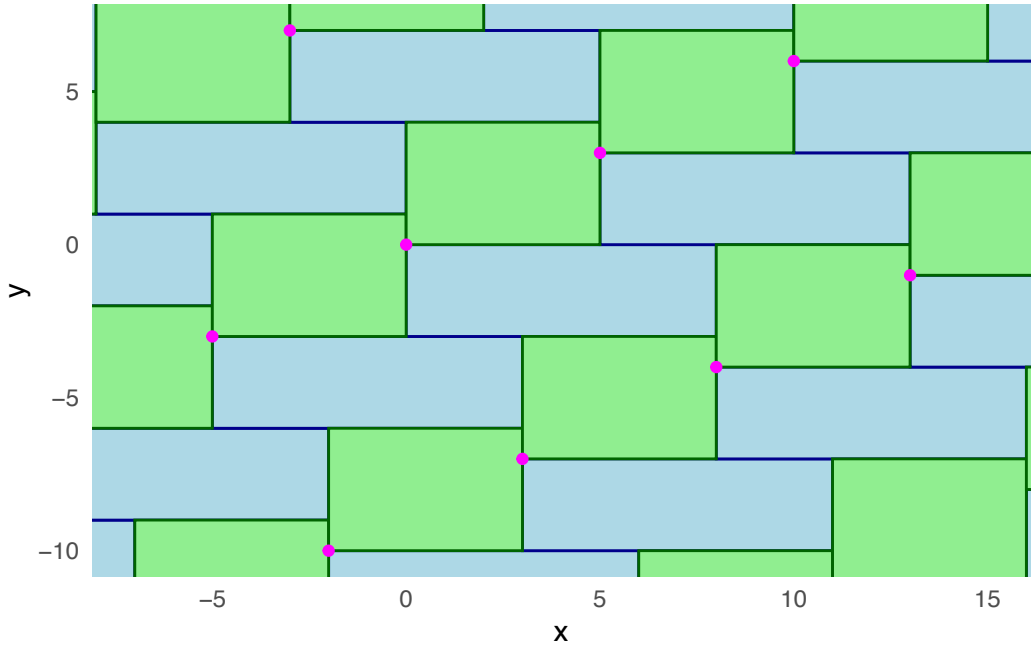


Figure 2: The second tiling, actually the very first that I drew (well, the first I drew correctly--see my mistake?)



Third Tile

$$M = \begin{bmatrix} 8 & 5 \\ -4 & 3 \end{bmatrix}$$



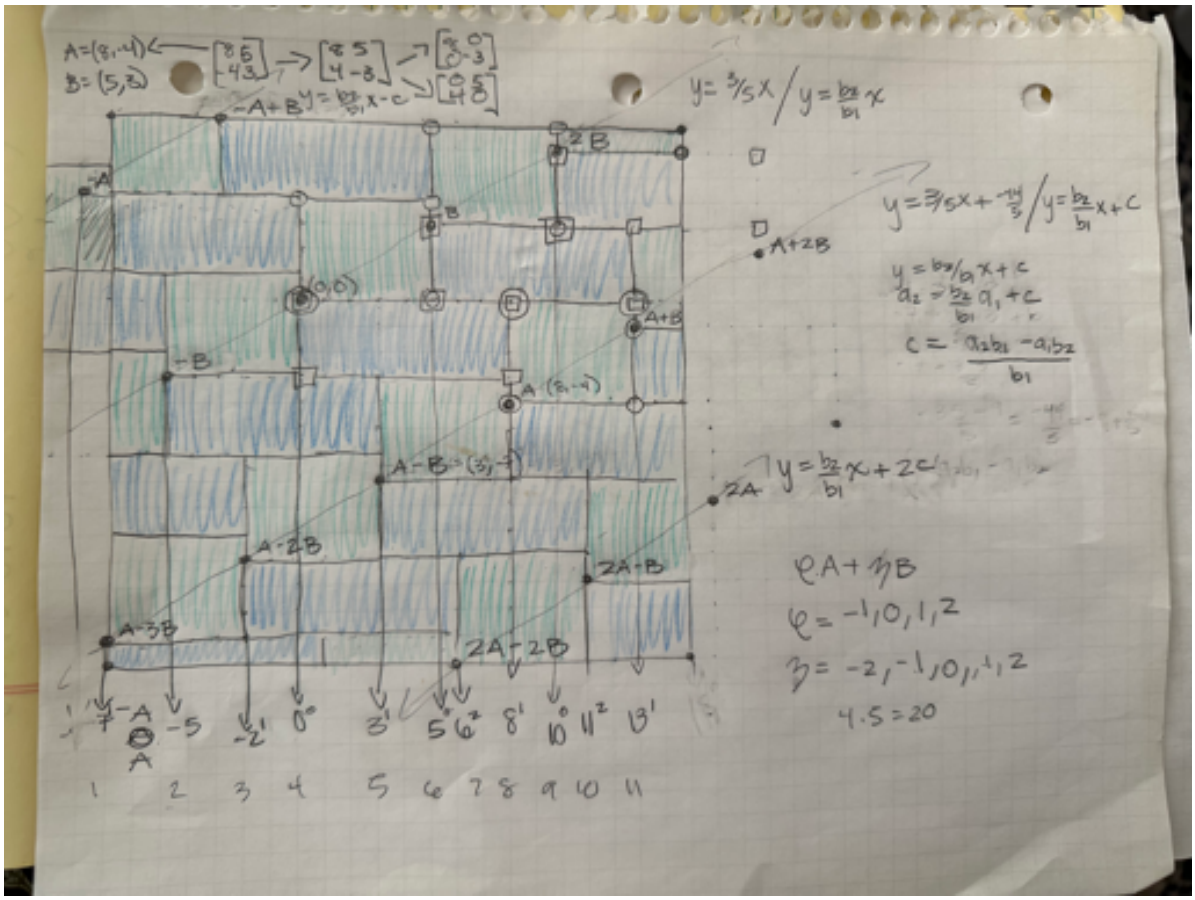


Figure 3: The drawing for the third tiling. I did this one for fun by myself! I chose big numbers so I could have space to write out coordinates and work out some math for the purposes of coding it up.

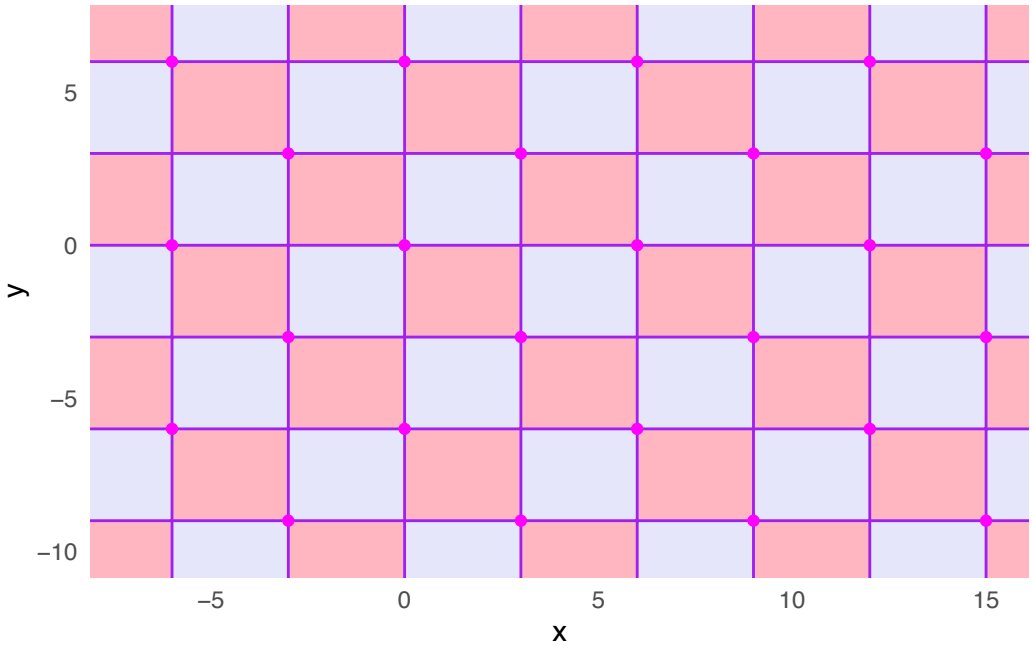
Computer-generated

Fourth Tile

$$M = \begin{bmatrix} 3 & 3 \\ -3 & 3 \end{bmatrix}$$

```
library(ggplot2)
## original coordinates
A <- matrix(c(3, -3), ncol = 1)
B <- matrix(c(3, 3), ncol = 1)

## coordinates for 4 copies/combos
copies <- -10:10
coefs <- expand.grid(copies, copies)

x.coords <- coefs$Var1*A[1] + coefs$Var2*B[1]
y.coords <- coefs$Var1*A[2] + coefs$Var2*B[2]

plot.dat <- data.frame(x = x.coords, y = y.coords)
p <- ggplot(data = plot.dat, aes(x = x, y = y)) +
    geom_point() +
    ylim(c(-14, 7)) +
    xlim(c(-7, 15)) +
    geom_rect(xmin = -7, xmax = 15, ymin = -14, ymax = 7,
                        fill = "#FFFFFF00", col = "black") +
    theme_minimal()


## could fix the grid to make these for printing!

## create on-diag rectangles
p1 <- p + geom_rect(xmin = plot.dat$x, xmax = plot.dat$x + A[1],
                            ymin = plot.dat$y, ymax = plot.dat$y - B[2],
                            fill = "lightpink", col = "pink") +
    ylim(c(-10, 7)) +
    xlim(c(-7, 15))
## create off-diag rectangles
p2 <- p1 + geom_rect(xmin = plot.dat$x, xmax = plot.dat$x + B[1],
                            ymin = plot.dat$y, ymax = plot.dat$y - A[2],
                            fill = "lavender", col = "purple")
```

```
p2 + geom_point(col = "magenta")
```



Fifth Tile

$$M = \begin{bmatrix} 12 & 6 \\ -3 & 1 \end{bmatrix}$$

```
library(ggplot2)
## original coordinates
A <- matrix(c(12, -3), ncol = 1)
B <- matrix(c(6, 1), ncol = 1)

## coordinates for 4 copies/combos
copies <- -10:10
coefs <- expand.grid(copies, copies)

x.coords <- coefs$Var1*A[1] + coefs$Var2*B[1]
y.coords <- coefs$Var1*A[2] + coefs$Var2*B[2]

plot.dat <- data.frame(x = x.coords, y = y.coords)
p <- ggplot(data = plot.dat, aes(x = x, y = y)) +
```

```
    geom_point() +
    ylim(c(-14, 7)) +
    xlim(c(-7, 15)) +
    geom_rect(xmin = -7, xmax = 15, ymin = -14, ymax = 7,
                        fill = "#FFFFFF00", col = "black") +
    theme_minimal()


## could fix the grid to make these for printing!

## create on-diag rectangles
p1 <- p + geom_rect(xmin = plot.dat$x, xmax = plot.dat$x + A[1],
                            ymin = plot.dat$y, ymax = plot.dat$y - B[2],
                            fill = "lightpink", col = "darkred") +
    ylim(c(-10, 7)) +
    xlim(c(-7, 15))
## create off-diag rectangles
p2 <- p1 + geom_rect(xmin = plot.dat$x, xmax = plot.dat$x + B[1],
                            ymin = plot.dat$y, ymax = plot.dat$y - A[2],
                            fill = "#E0FFFF", col = "#A2FFFF")
p2 + geom_point(col = "magenta")
```