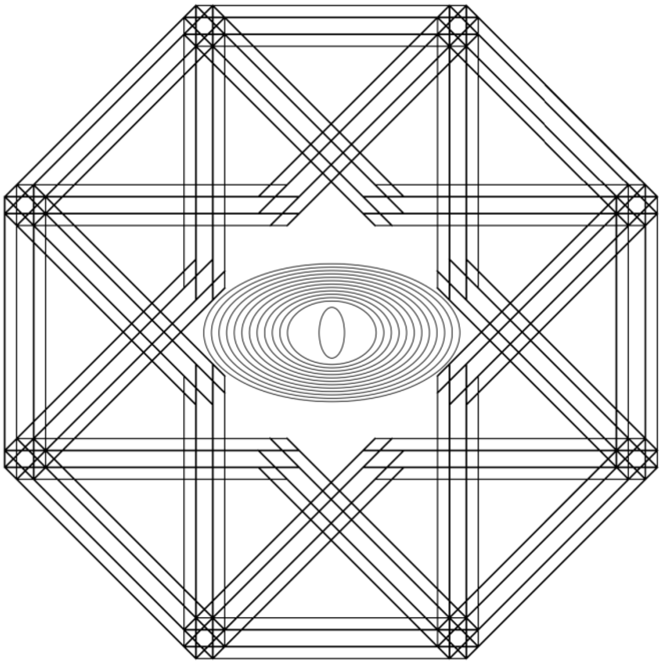
M2 — “Lasered”

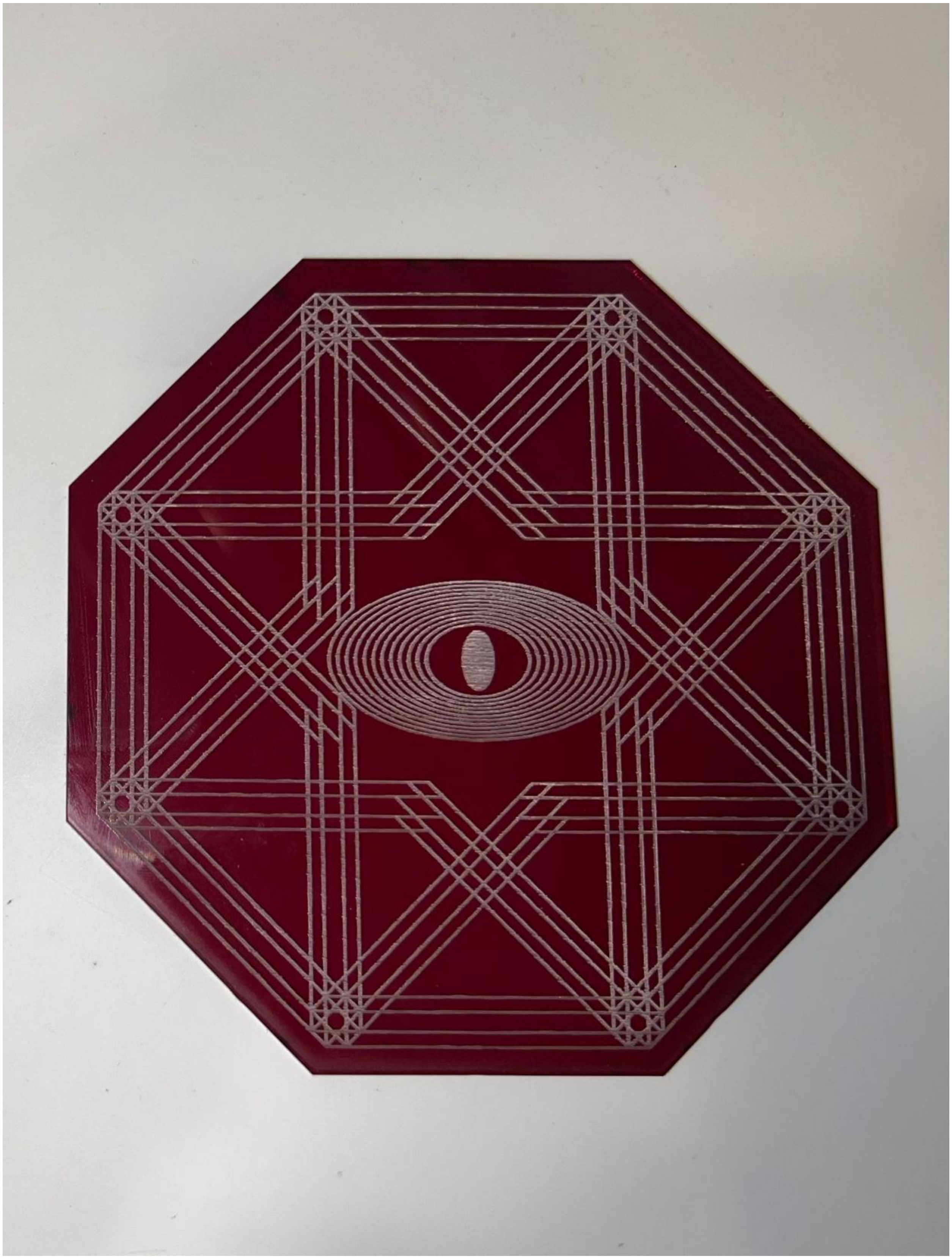
“Lasered” is centered around processing, laser cutting, and generated design. Concurrently, it is an exercise in fabrication, providing the opportunity to work with the laser cutters themselves, etching and removing generated designs. “Lasered” encourages practice with processing, making more detailed designs as our understanding of, and familiarity with, processing grows.

Materials Used

- Acrylic

Design #1





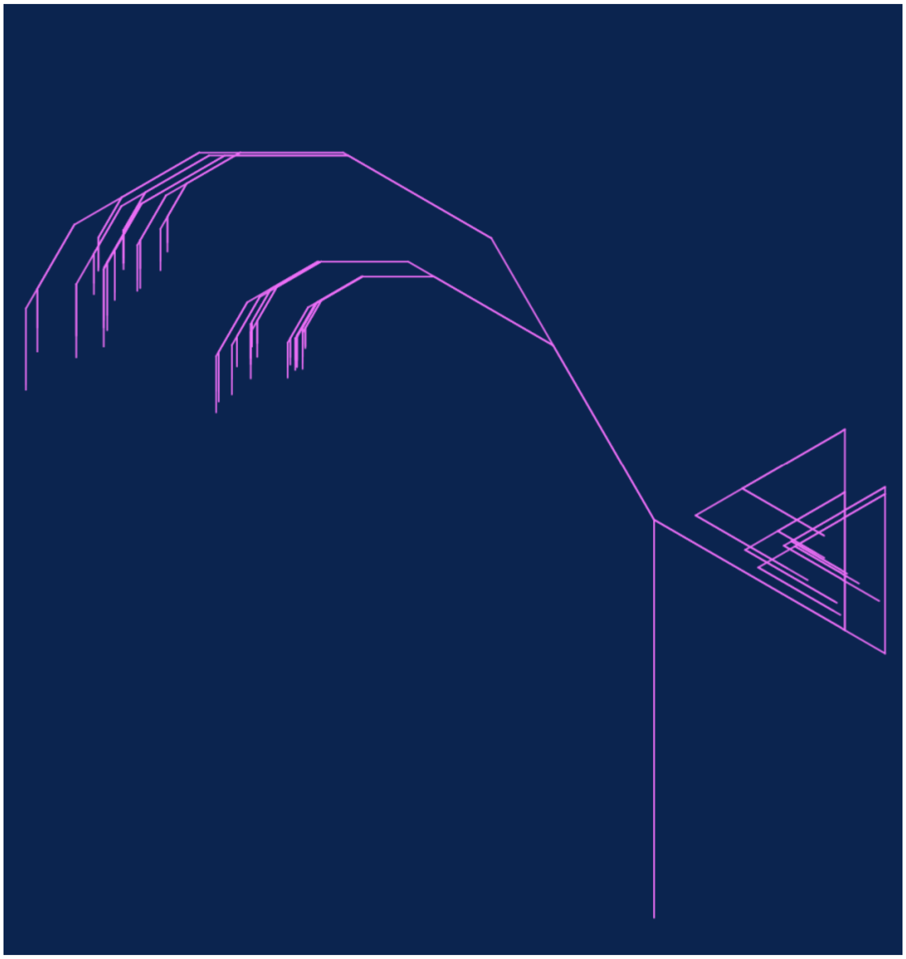
My first design utilized the turtle library as well as processing functions to create a star-like pattern house within an octagon using forward movement and angle changes. These movements were housed within for loops to create the repetition. I decided to follow this repetitive nature into the center of the octagon 'housing,' using ellipses to create something reminiscent of an eye in the center. On the laser cut piece, the center of the eye is filled to emphasize this.

Code #1

```
import Turtle.*;
import processing.pdf.*;
Turtle t;


void setup(){
size(1000,1000);


t = new Turtle(this);


t.setX(100);
t.setY(600);


t.setHeading(90);


noFill();
noLoop();
}


void draw(){
 float y = 105;
 float z = 75;
```

```
 float k = 10;

 float j = 10;


for (int p = 0; p < 77; p += 1){

 for (int i = 0; i < 77; i += 1){

   t.forward(300);

   t.right(45);

   t.forward(300);

   t.left(90);

   t.forward(300);

   t.left(90);

 }


  t.forward(20);


}

   for (int a = 0; a < 12; a += 1){

    ellipse(453,475,y,z);

    y+=18;

    z+=8;

  }





  // square(433,455,40);

  fill(255);

  ellipse(453,475,30,60);

}


void keyPressed() {

 // press 's' to save a svg of your drawing

 if (key == 's') {
   // Make file name with the currrent date/time
   String folder = "output";
   String fileName = "drawing-" + getDateString() + ".pdf";
   beginRecord(PDF, folder + "/" + fileName);
   setup();
   draw();
   endRecord();
   println("Saved to file: " + fileName);
 }
}


// Generates a date string of the format year_month_day-hour_min_second
String getDateString() {
 String time = str(hour()) + "_" + str(minute()) + "_" + str(second());
 String date = str(year()) + "_" + str(month()) + "_" + str(day());
 return date + "-" + time;
}
```

Design #2

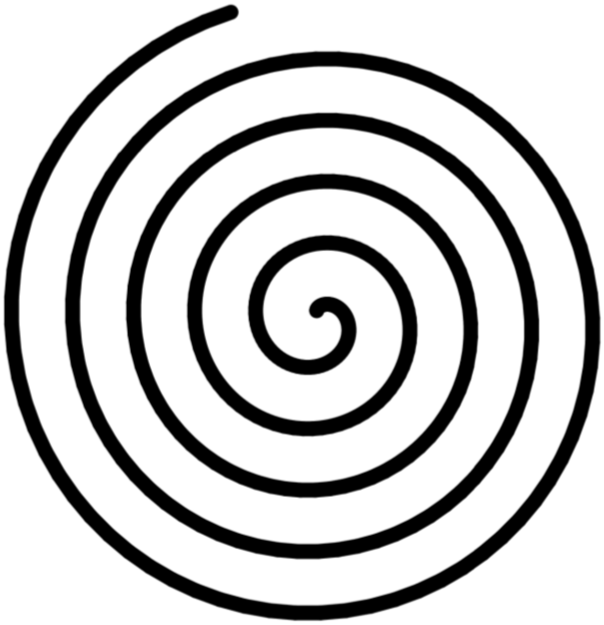


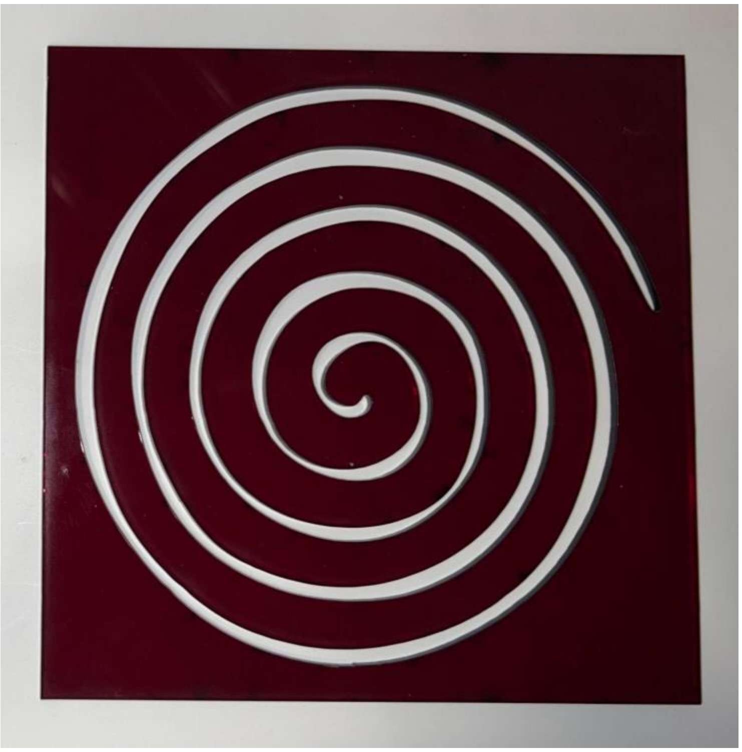
This second design was inspired by the branching function available with the turtle library. I found the natural branch patterns created to be quite satisfying. And so, I played around with angles, branch number, and even random generation, creating something that appears to me as a palm tree, almost, with the addition of some branched triangles creating the other half of the tree. To accomplish this, I followed the general parameters given by the turtle branch function, altering branch direction, length, and angle. Similarly, I took the opportunity to start playing around with colors in processing, settling on a pink branching and blue background.

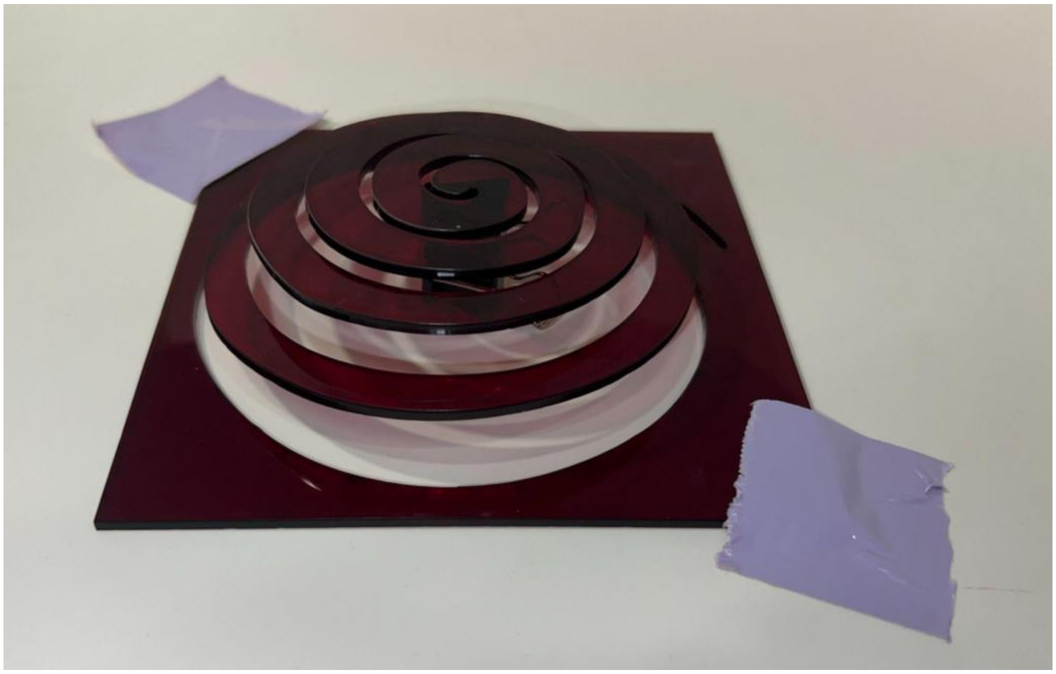
Code #2

```
import Turtle.*;
import processing.pdf.*;
Turtle t;


void setup() {
 size(1000,1000);
 background(9554);
 stroke(645,101,1799);
 t = new Turtle(this);
 noLoop();
}


void draw () {
 t.penUp();
 t.back(200);
 t.penDown();
 branch(7,300,30);
 bran(5,300,120);
}


//a recursive tree-drawing procedure
void branch(int iteration, float branchLength, int angle)
{
 if (iteration == 0)
   return;
 t.forward(branchLength);
 t.left(angle);
 branch(iteration-1, branchLength/random(1,2) + 1, angle);
 t.right(angle);
 t.left(angle);
 branch(iteration-1, branchLength/random(1,2) + 1, angle);
 t.right(angle);
 t.back(branchLength);
}


void bran(int iterat, float bLength, int ang)
{
 if (iterat == 0)
   return;
 t.forward(bLength);
 t.right(ang);
 branch(iterat-1, bLength/random(1,2) + 1, ang);
 t.left(ang);
 t.right(ang);
 branch(iterat-1, bLength/random(1,2) + 1, ang);
 t.left(ang);
 t.back(bLength);
}


void keyPressed() {
 // press 's' to save a svg of your drawing
 if (key == 's') {
   // Make file name with the currrent date/time
   String folder = "output";
   String fileName = "drawing-" + getDateString() + ".pdf";
   beginRecord(PDF, folder + "/" + fileName);
   setup();
   draw();
   endRecord();
   println("Saved to file: " + fileName);
 }
}


// Generates a date string of the format year_month_day-hour_min_second
String getDateString() {
 String time = str(hour()) + "_" + str(minute()) + "_" + str(second());
 String date = str(year()) + "_" + str(month()) + "_" + str(day());
 return date + "-" + time;
}
```

Design #3







For my third and final design, I went with a tight spiral using a simple for loop in conjunction with basic turtle movements. My idea here was to play with stroke weight in order to completely cut out the spiral, allowing for the shape traced to create a sort of pop-up cut out. Although this design was a bit less intricate than my others, I was most excited to fabricate this shape, and to see if my idea would bear fruit.

Code #3

```
import Turtle.*;
import processing.pdf.*;


Turtle t;


void setup(){
size(900,900);
strokeWeight(20);
noLoop();


t = new Turtle(this);
}


void draw(){
 float dist = 0.1;


 for (int i = 0; i < 375; i += 1){
   t.forward(dist);
   // t.drawTurtle();
   t.right(5);
   dist += 0.1;
 }
}


void keyPressed() {
 // press 's' to save a svg of your drawing
 if (key == 's') {
   // Make file name with the currrent date/time
   String folder = "output";
   String fileName = "drawling-" + getDateString() + ".pdf";
   beginRecord(PDF, folder + "/" + fileName);
   setup();
   draw();
   endRecord();
   println("Saved to file: " + fileName);
 }
}


// Generates a date string of the format year_month_day-hour_min_second
String getDateString() {
 String time = str(hour()) + "_" + str(minute()) + "_" + str(second());
 String date = str(year()) + "_" + str(month()) + "_" + str(day());
 return date + "-" + time;
}
```

Problems/Conclusion

To be completely frank, I did not face too many problems over the course of this project. Aside from the occasional missed semicolon, incorrectly declared for loop, or other typical coding problem, it went quite smoothly. The only challenge I truly had to overcome was dealing with rhino after I had encountered a layering issue due to my looping. Overall, I have been enjoying processing quite a bit, even playing with it and the turtle library in my free time. I similarly enjoy using the laser cutters and fabrication as a whole, so I was quite eager to transfer the designs onto pieces of plywood.