puncture

```
/**
* project-lumbar puncture
*/
import processing.serial.*;

Serial myPort;        // The serial port

//initialize all variables
float inByte = 0; //current value of the first variable in the string
float lastByte = 0; //previous value of the first variable in the string
float inByte2 = 0; //current value of the second variable in the string
float lastByte2 = 0; //previous value of the second variable in the string
float myValue = 0;
float myValue1 = 0;

PImage img; // Declare variable "a" of type PImage
PImage img1;
float mx; // needle pos
float my;
float mx1; // uper block pos
float my1;
float mx2; // lower block pos
float my2;
float mxa; // x-rail pos
float mya;
float mxb; // y-rail pos
float myb;

void setup() {
 size(893, 609);
 noStroke(); // no outline
 img = loadImage("2.jpg"); // Load the image into the program
 img1 = loadImage("1.jpg");

 // List all the available serial ports
 println(Serial.list());

 // Check the listed serial ports in your machine
 // and use the correct index number in Serial.list()[] below.
 // Note that these are indexed from 0, and you are looking for the same port as your ardunio.
 myPort = new Serial(this, Serial.list()[0], 9600); //make sure baud rate matches Arduino

 // A serialEvent() is generated when a newline character is received :
 myPort.bufferUntil(&apos;\n&apos;);
}

void serialEvent (Serial myPort) {
 // get the ASCII string:
 //myValue != NaN Float.isNaN(myValue) myPort.readStringUntil(&apos;\n&apos;)
 String string = myPort.readStringUntil(&apos; &apos;);
 if (string != null){
   string = trim(string);
   myValue = float(string);
   println (string);
 }
 String string1 = myPort.readStringUntil(&apos;\n&apos;); //mass position
 if (string1 != null){
   string1 = trim(string1);
   myValue1 = float(string1);
   println (string1);
 }
}

void draw () {
 // background
 image(img, 0, 0);
 fill(255);
 rectMode(CENTER);
 rect(width/2+80, height/2+40, 200, 400);

 // resitrit
 fill(255);
 rectMode(CENTER);
 rect(width/3-20, height/2+30, 300, 300);
 float rx = width/3-20;
 float ry = height/2+30;
 float xdown = map(myValue, 0, 230, rx-100, rx+100);
 float xup = map(myValue1, 0, 9.83, ry-75, ry+75);

 // rectangle with two points (the top left corner and the bottom right corner)
 // rail width 50
 // upper rail
 fill(255);
 rectMode(CENTER);
 rect(rx, ry-150+25, 300, 50);

 // lower rail
 fill(255);
 rectMode(CENTER);
 rect(rx, ry+150-25, 300, 50);

 // x-rail
 mxa = constrain(xdown, rx-150+25, rx+150-25);
 mya = constrain(xup, ry, ry);
 fill(255);
 rectMode(CENTER);
 rect(mxa, mya, 50, 300);

 // upper block
 mx1 = constrain(xdown, rx-150+25, rx+150-25);
 my1 = constrain(xup, ry-150+25, ry-150+25);
 fill(255);
 rectMode(CENTER);
 rect(mx1, my1, 50,50);

 // lower block
 mx2 = constrain(xdown, rx-150+25, rx+150-25);
 my2 = constrain(xup, ry+150-25, ry+150-25);
 fill(255);
 rectMode(CENTER);
 rect(mx2, my2, 50,50);
```

```
// y-rail
mxb = constrain(xdown, rx, rx);
myb = constrain(xup, ry-150+75, ry+150-75);
fill(255);
rectMode(CENTER);
rect(mxb, myb, 300, 50);

//
imageMode(CORNER);
image(img1, rx, ry-100, 300, 300);

// needle
mx = constrain(xdown, rx-150+25, rx+150-25);
my = constrain(xup, ry-150+75, ry+150-75);
fill(200);
rectMode(CENTER);
rect(mx, my, 100, 50);
fill(200); // needle tip (right
rectMode(CENTER);
rect(mx+100, my, 120,5);
fill(200); // left of needle
rectMode(CENTER);
rect(mx-30, my, 80,15);
fill(200); // left of needle
rectMode(CENTER);
rect(mx-65, my, 15,80);

}
```