

**MINISTERUL EDUCAȚIEI, CULTURII ȘI CERCETĂRII AL REPUBLICII MOLDOVA**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică şi Microelectronică Departamentul Inginerie Software și Automatică**

**Gusev Roman FAF-222**

Report

*Laboratory work n.4*

*Point A*

***of Computer Graphics***

Checked by:

**Olga Grosu,** *university assistant*

DISA, FCIM, UTM

**Chișinău – 2023**

**1. Purpose of the Laboratory Work**

In point A of laboratory Work nr. 4, I had to draw a sketch and use different random functions, such as randomGaussian, PerlinNoise etc. Main purpose of this point is to make me familiar with random theory, Probability and random functions in Processing, and how they may be used to simulate different kind of things.

**2. Condition of the Laboratory Work**

A: Do the sketch using the function:

• randomGaussian()

• randomSeed()

• random()

• noiseDetail()

• noiseSeed()

• noise()

• map()

**3. The program code**

// Laboratory Work 4

// Point A

// Student: Gusev Roman

// Group: FAF-222

// setting the sizes for both parts of the canvas

int width\_window = 1000;

int height\_window = 500;

// setting the random seed;

int seed = round(random(0, 1));

// setting values for randomGaussian

int sd = 160;

int mean = 0;

// setting values for Perlin Noise Generator

float noiseVal;

float noiseScale=0.05;

void setup() {

size(1000, 1000);

randomSeed(seed);

fill(10);

rect(0, 0, width\_window, height\_window);

frameRate(200);

}

void draw() {

stroke(lerpColor(color(255), color(255, 166, 0), random(0, 1)));

int randomWeight = round(random(1, 3));

strokeWeight(randomWeight);

// left side of upper half randomGaussian sketch

float x\_loc = randomGaussian();

x\_loc = abs(x\_loc \* sd) + mean;

x\_loc = constrain(x\_loc, 0, width\_window/2);

float shift = x\_loc \* 0.32;

float y\_loc = random(0 + shift, 500 - shift);

point(width\_window / 2 - x\_loc, y\_loc);

// right side of upper half randomGaussian sketch;

x\_loc = randomGaussian();

x\_loc = abs(x\_loc \* sd) + mean;

x\_loc = constrain(x\_loc, 0, width\_window/2);

shift = x\_loc \* 0.32;

y\_loc = random(0 + shift, 500 - shift);

point(width\_window / 2 + x\_loc, y\_loc);

fill(0);

stroke(0);

rectMode(CENTER);

rect(width\_window/2 + 6, height\_window + 25, 290, 30);

textSize(20);

fill(255);

text("Press any key to draw Perlin Noise", width\_window/2 - 135, height\_window + 30);

// lower half designed for Perlin Noise

if (keyPressed) {

drawPerlinNoise();

}

}

void drawPerlinNoise() {

noiseSeed(1); // You can change the seed value

noiseDetail(round(random(1, 5)), 0.5); // Adjust these parameters to control the noise

for (int x = 0; x < width; x++) {

for (int y = height\_window; y < height; y++) {

float noiseVal = noise(x \* noiseScale, y \* noiseScale);

float bright = map(noiseVal, 0, 1, 0, 255);

stroke(bright, 100);

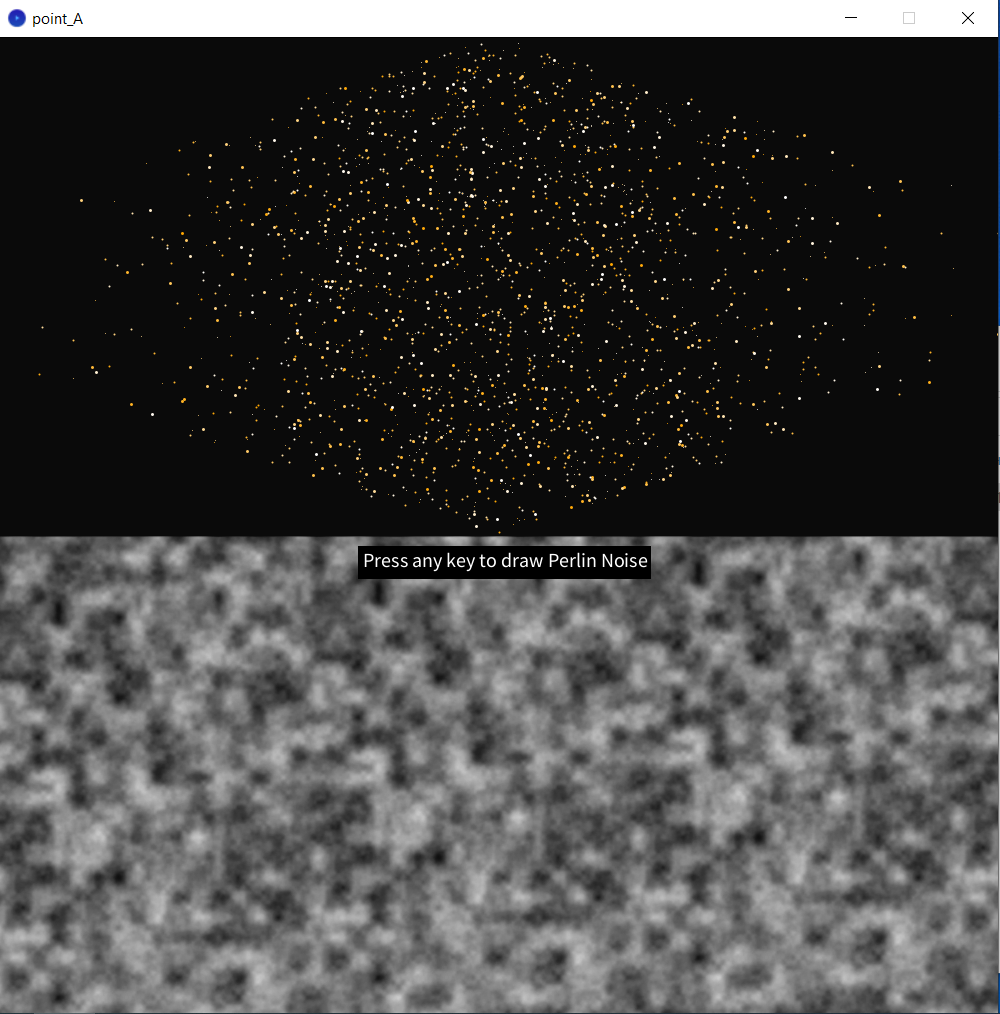
point(x, y);

}

}

}

**4. Screen printing of program execution**

****

**5. Conclusion**

By the end of this Laboratory Work nr. 4, I accomplished given task exactly with the requirements. I draw on canvas two halves: upper half – I used randomGaussian function and several other mentioned in the condition of the laboratory, such as map, randomSeed and random. I sketched two trapezes that has a higher density of stars in the center, between them. I have encountered some troubles doing and figuring out how to draw correctly the trapezes and reverse their gaussian density, and lower half of the canvas – I sketched basic Perlin Noise Generation. User has to press a key to draw a Perlin Noise map. I also used several random functions, noiseSeed function, noiseDetail and others. When any key is pressed, the map is changed with a random noiseDetail value. I have not encountered big problems here. As a sum of the above, I can denote that this point I accomplished with success.