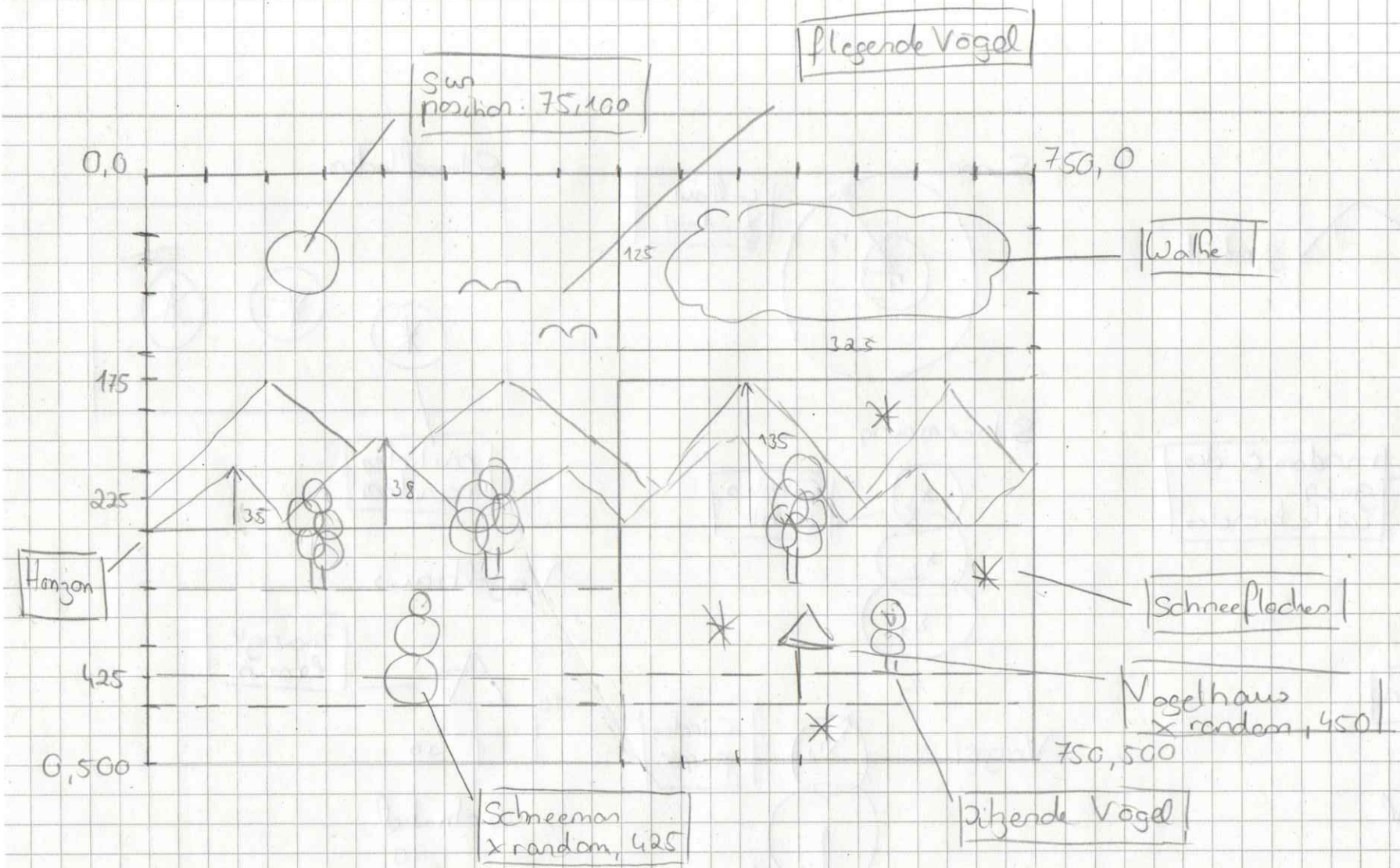
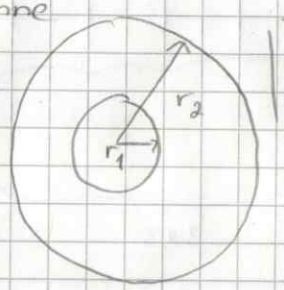
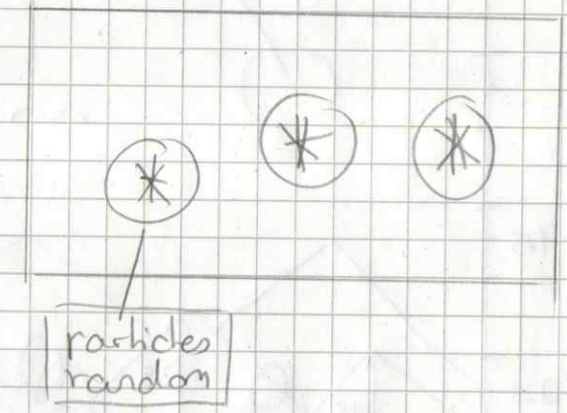

fliegende Vögel
Sun
position: 75,100
0,0
750,0
125
Wolke
325
175
225
135
38
35
Hangon
Schneeflocken
425
Vogelhaus
x random, 450
0,500
750,500
Schneeman
x random, 425
sitzende Vögel

Berge

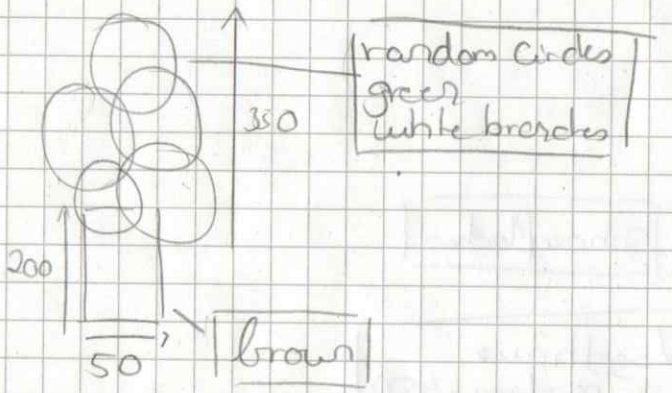
Sonne



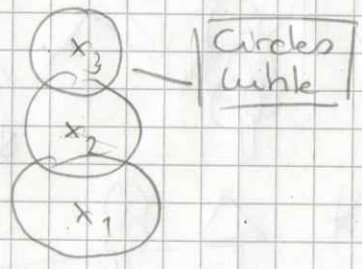
Schneeflocken



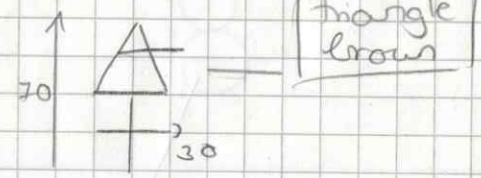
Baum



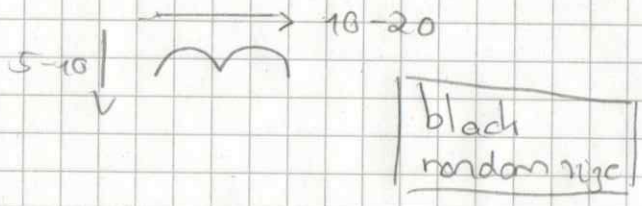
Schneemann



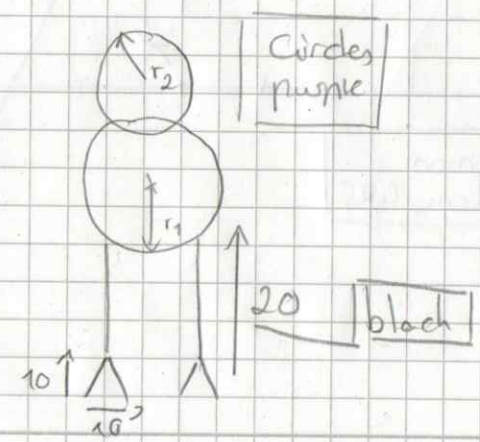
Vogelhaus



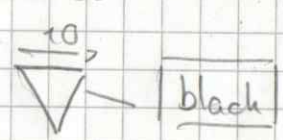
Fliegender Vogel



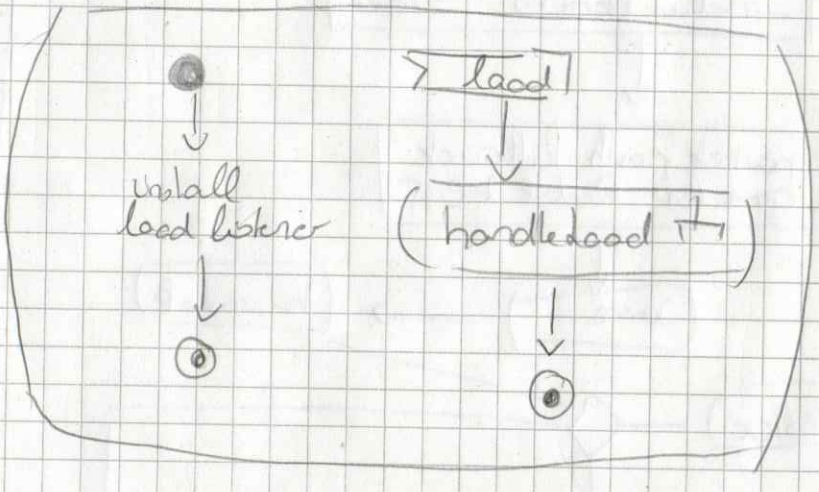
Vogel

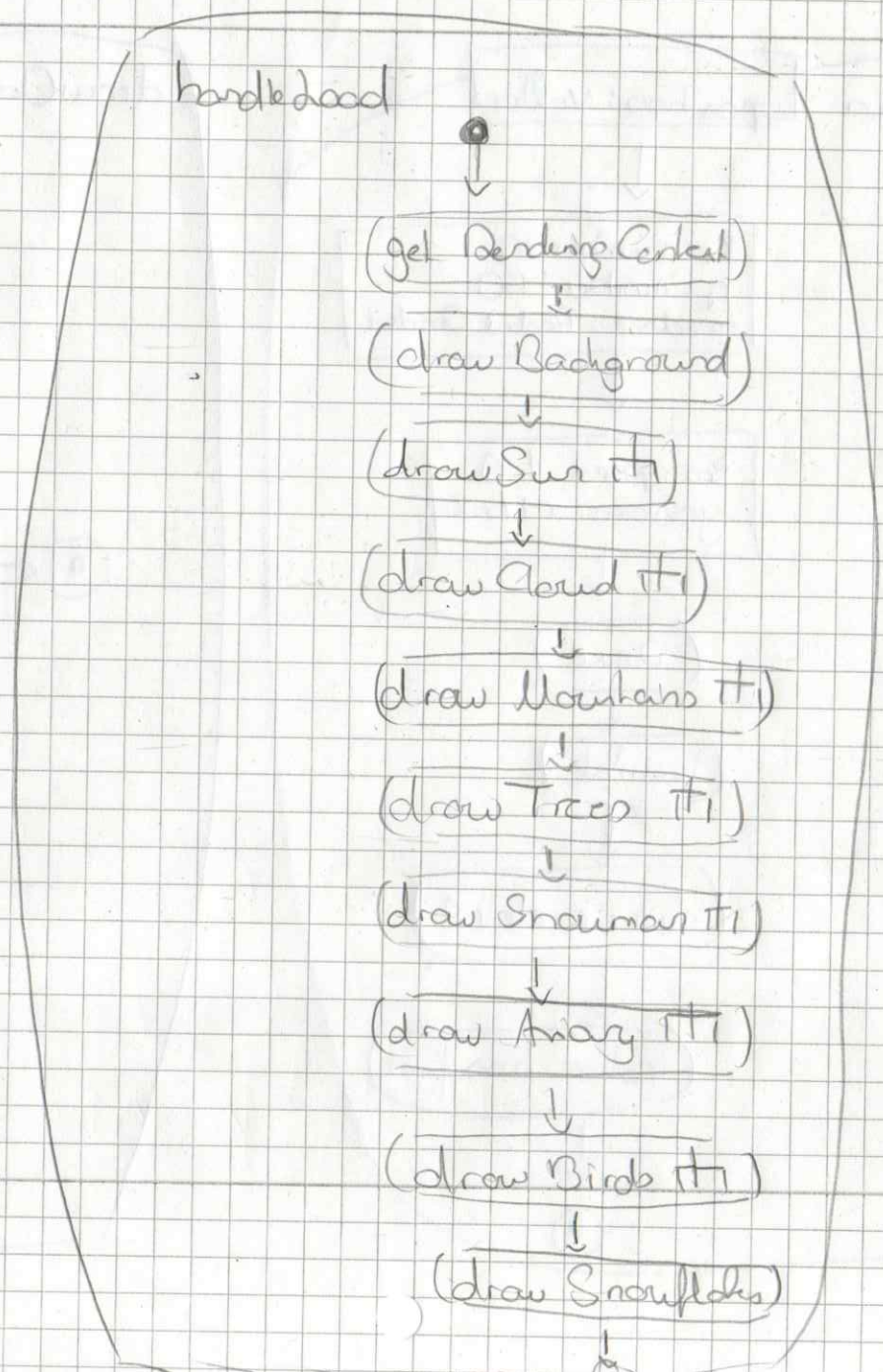


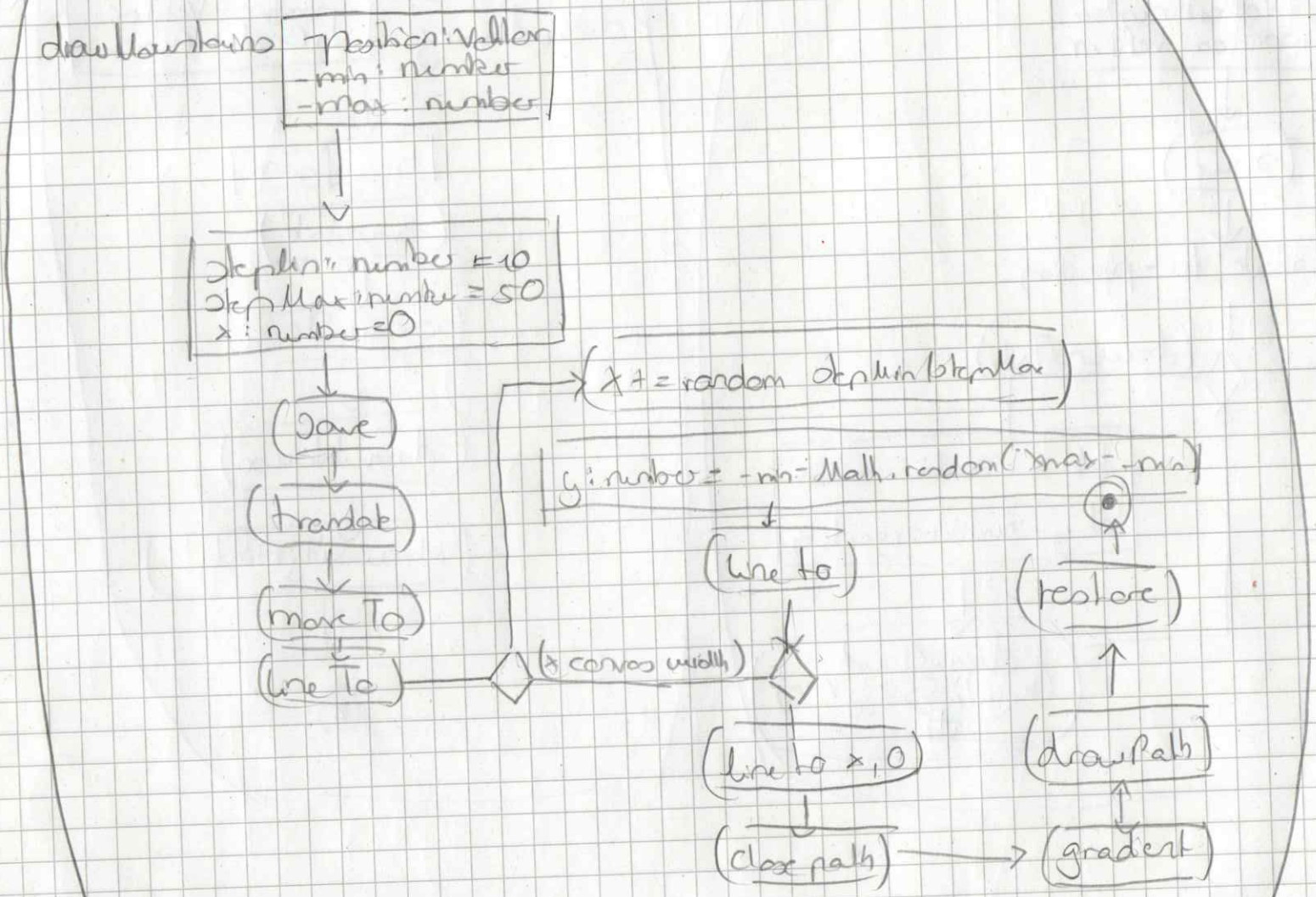
Schnabel

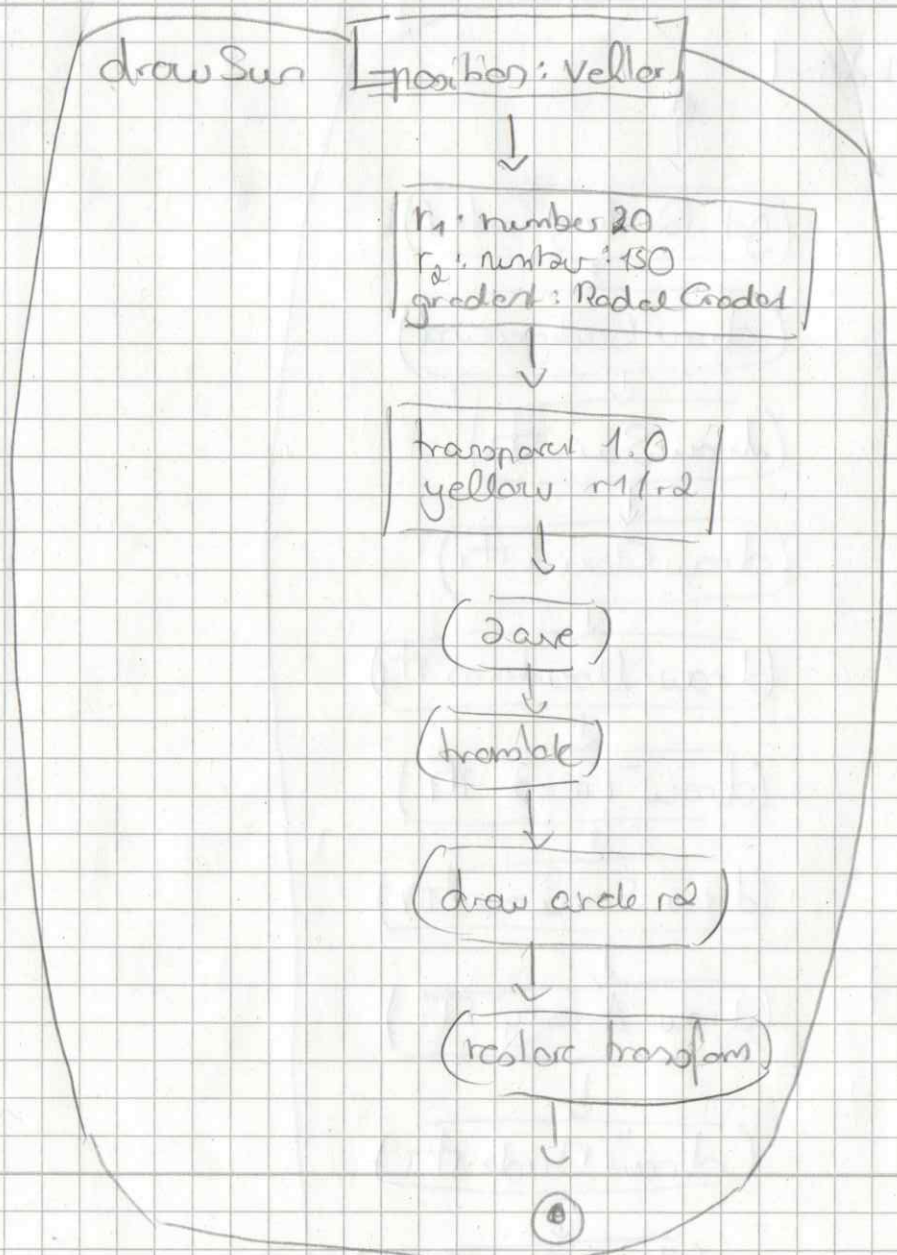


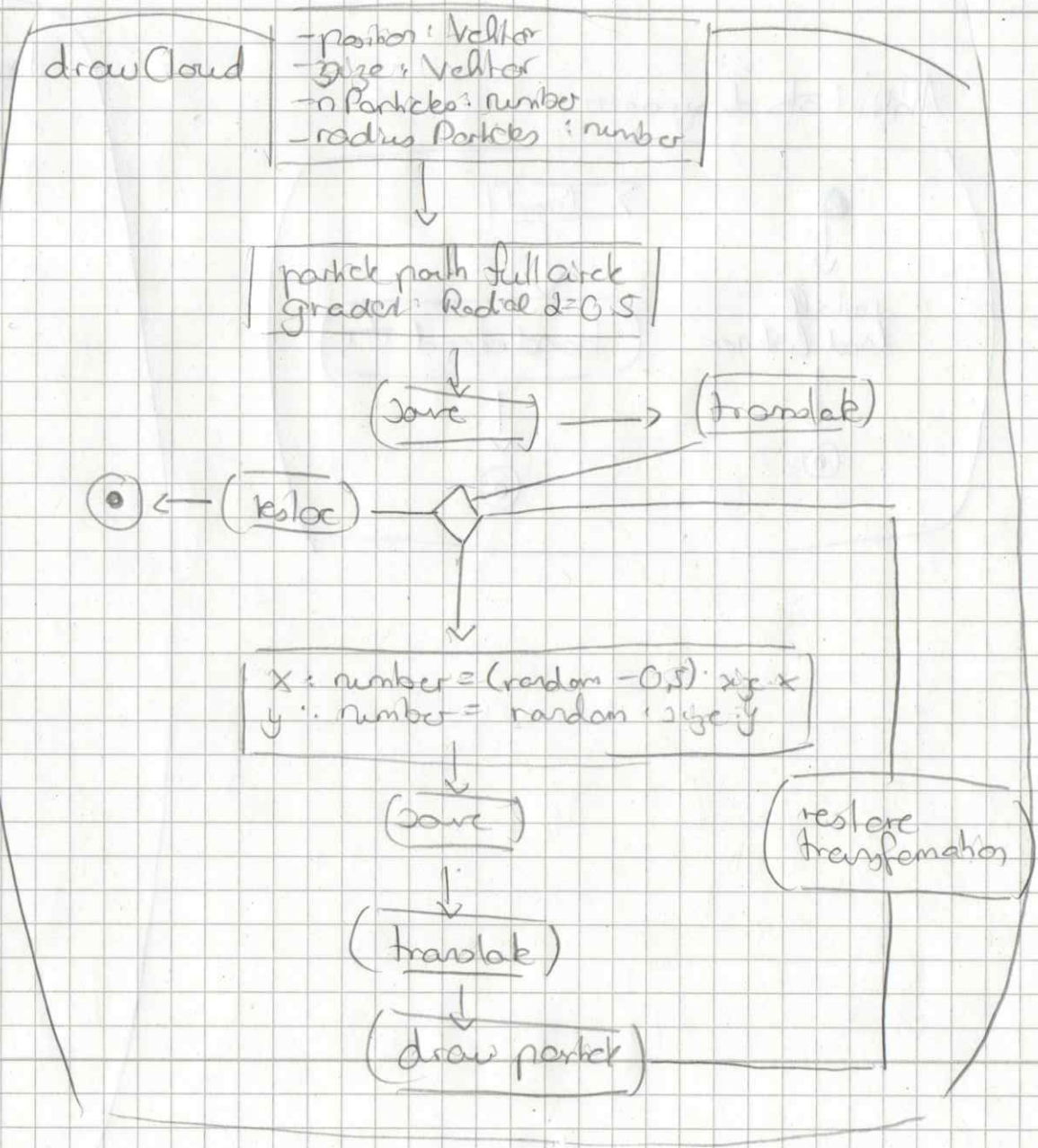
Aktivitetsdiagram

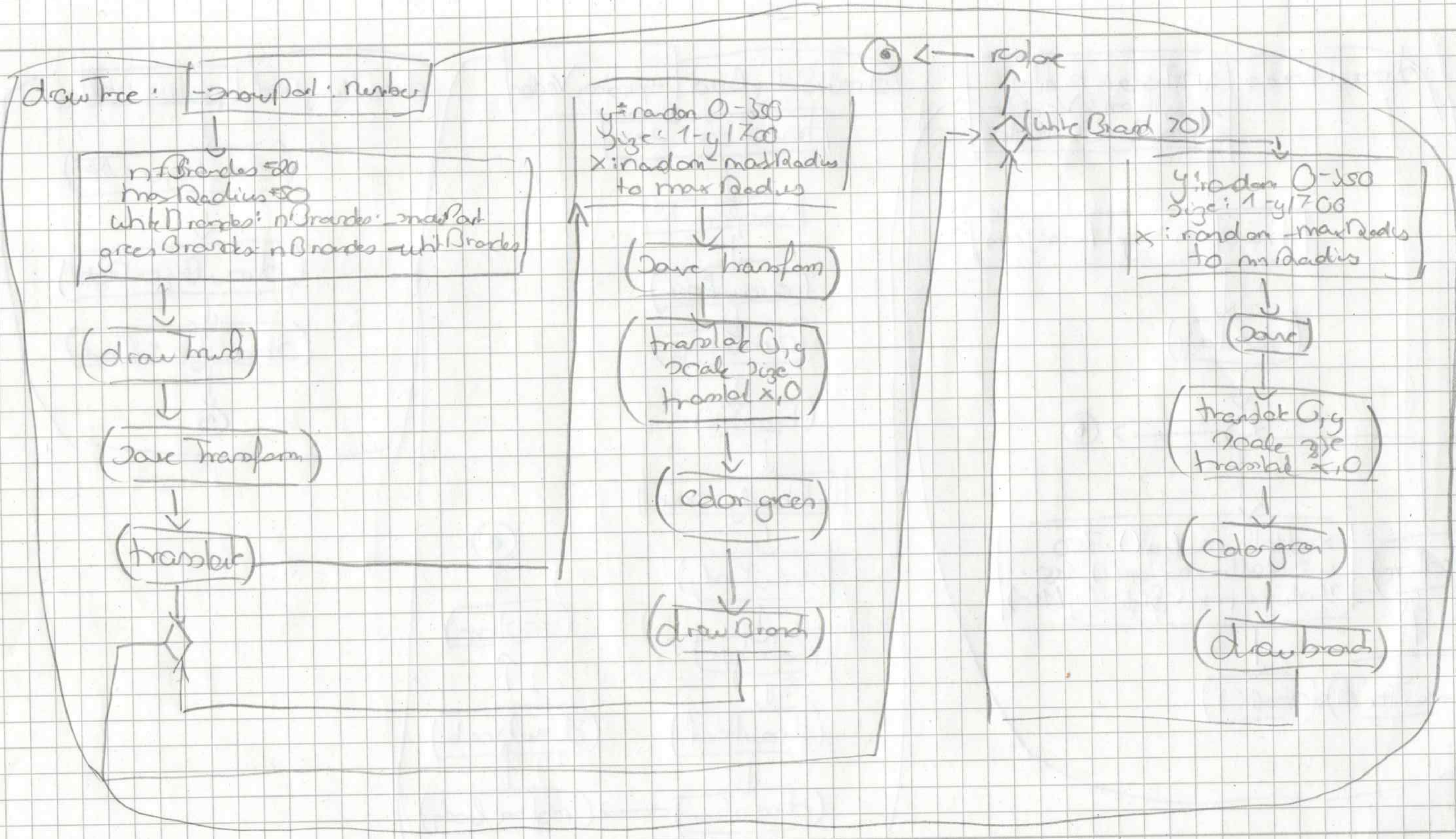


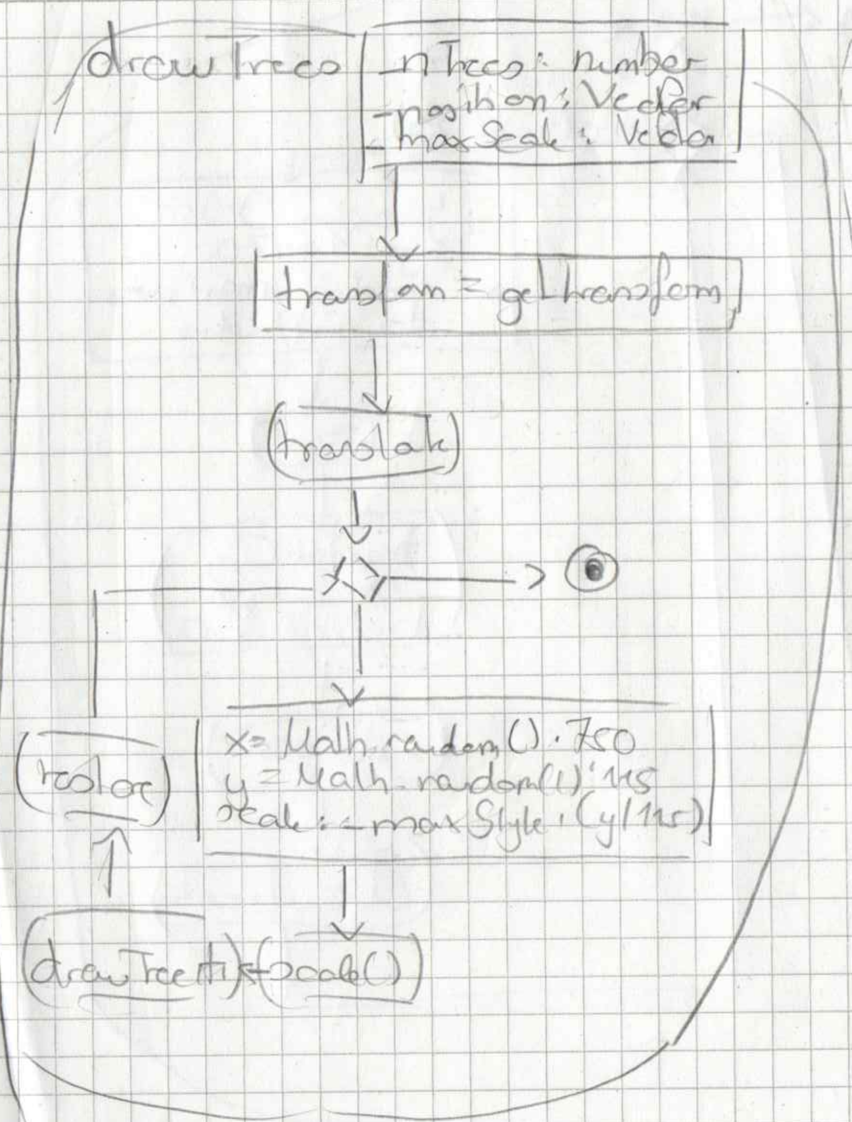


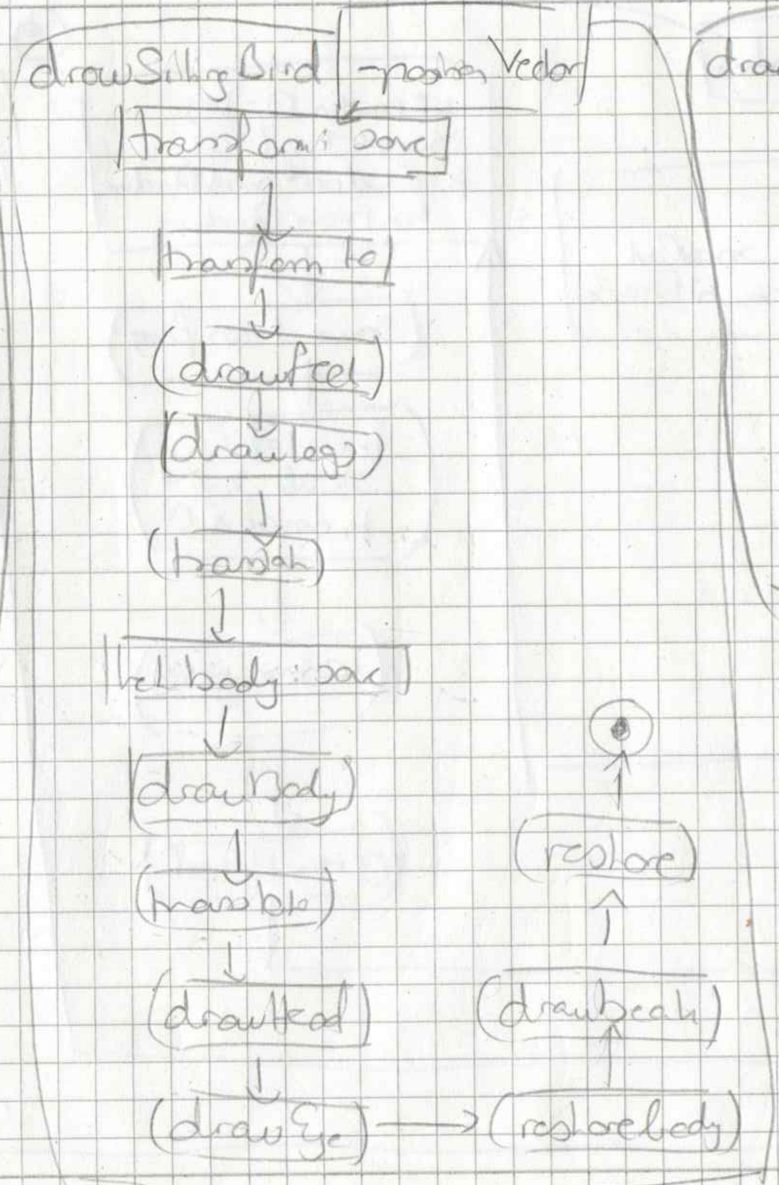

drawMountains
position: Vector
-min: number
-max: number
stepMin: number = 10
stepMax: number = 50
x: number = 0
save
translate
moveTo
lineTo
x += random stepMin stepMax
y: number = -min - Math.random() * (max - min)
lineTo
x < canvas.width
lineTo x, 0
closePath
gradient
drawPath
restore

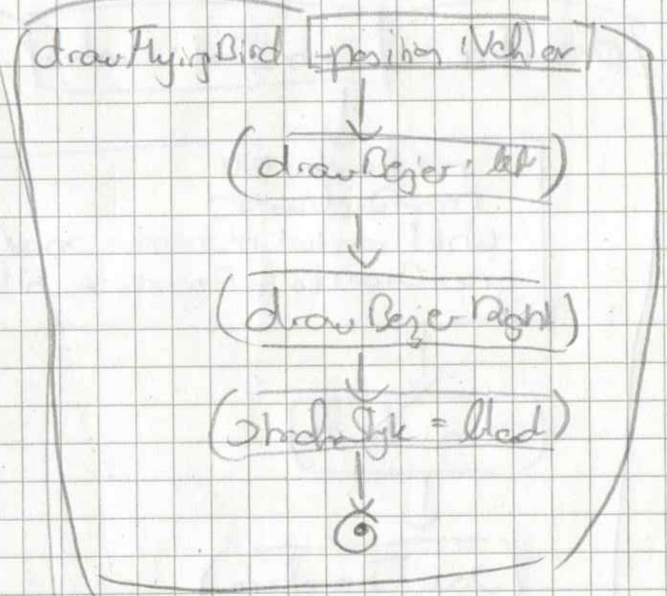

drawSun
-position : Vector
r1 : number 20
r2 : number 150
gradient : RadialGradient
transparent 1.0
yellow rgb
save
translate
drawGradCircle
restore transform

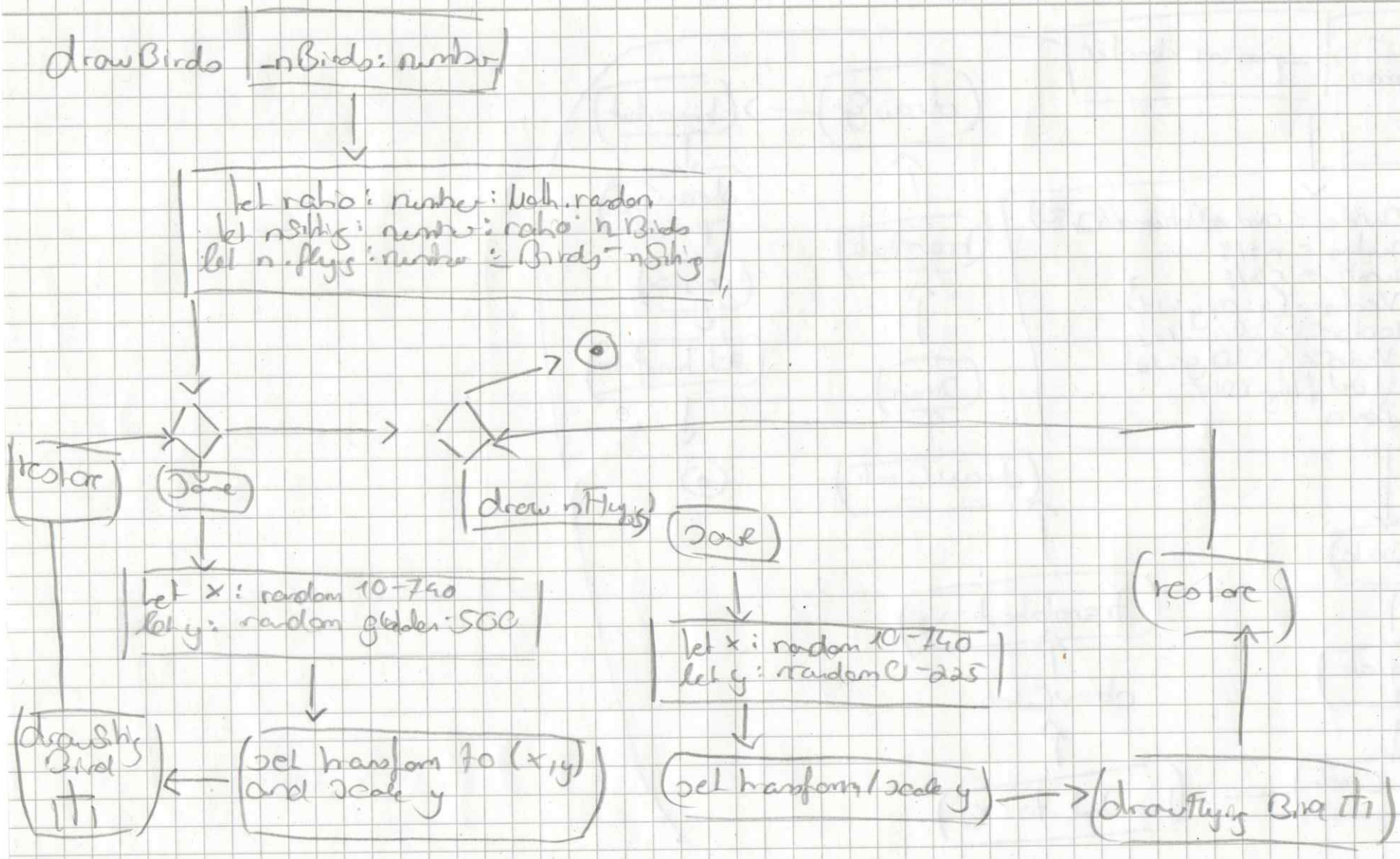

drawCloud
-position : Vector
-size : Vector
-nParticles : number
-radiusParticles : number
particle path fullcircle
gradient : Radial d=0.5
save
translate
restore
x : number = (random -0.5) * size.x + x
y : number = random * size.y
save
translate
drawParticle
restore transformation

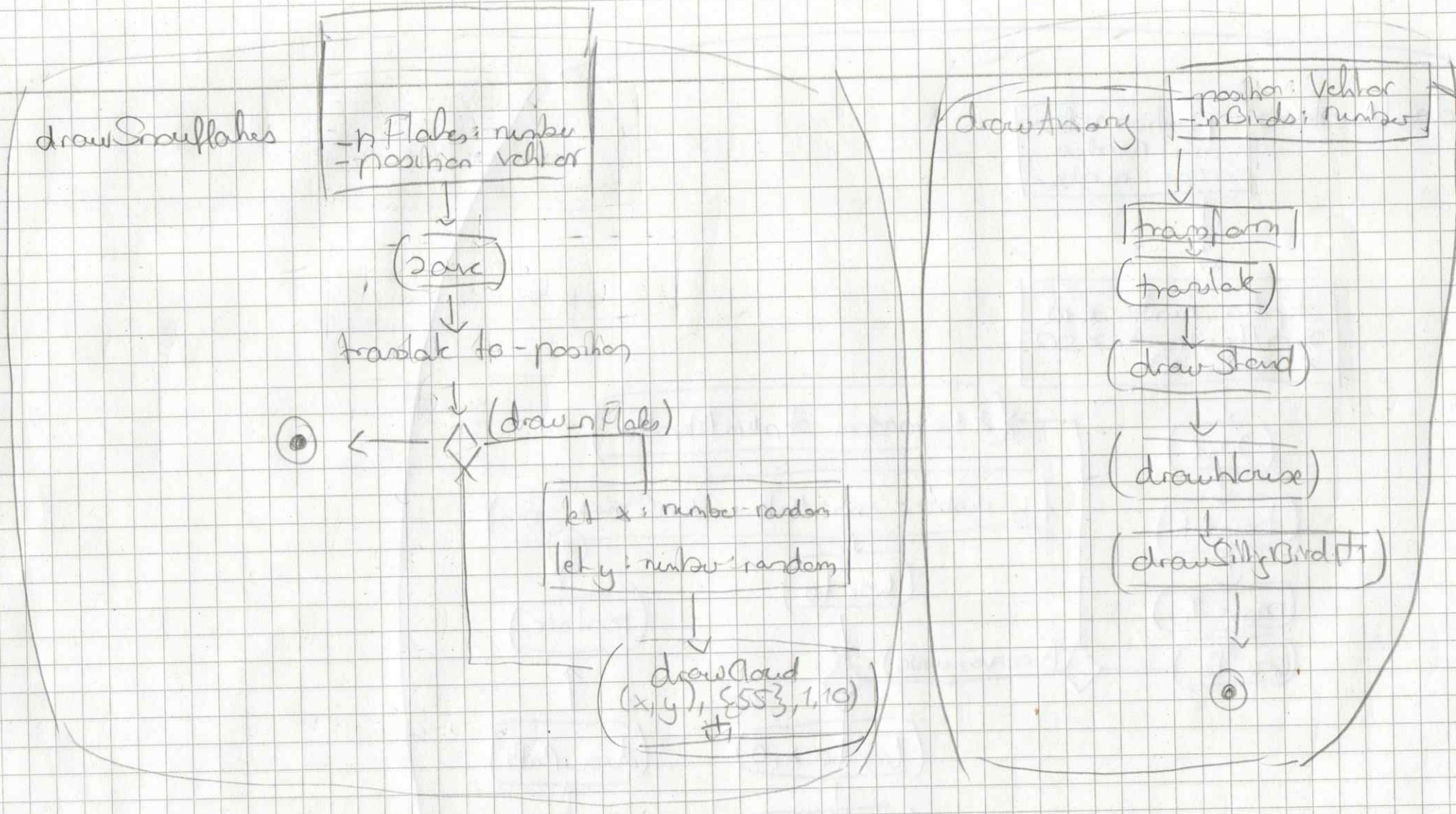

drawTree :
-snowPart : number
whiteBranches: nBranches · snowPart
greenBranches: nBranches - whiteBranches
drawTrunk
save Transform
translate
y: random 0-350
size: 1-y/700
x: random maxRadius to maxRadius
save transform
translate 0,y
scale size
translate x,0
color green
drawBranch
(whiteBranch 70)
restore
y: random 0-350
size: 1-y/700
x: random -maxRadius to maxRadius
save
translate 0,y
scale size
translate x,0
Color green
drawbranch

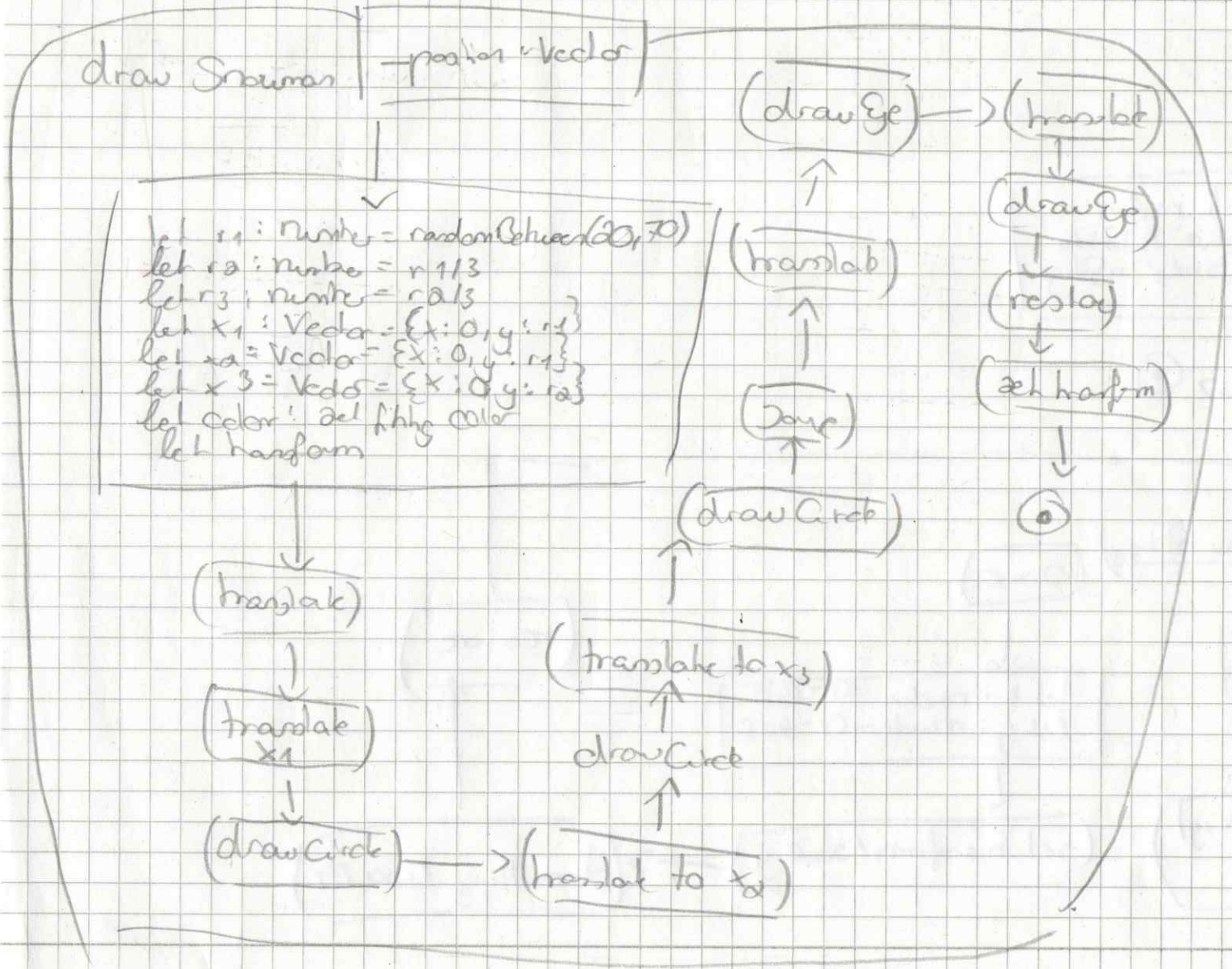

drawTrees
-nTrees: number
-position: Vector
-maxScale: Vector
transform = getTransform
translate
restore
x = Math.random() * 750
y = Math.random() * 145
scale := maxStyle * (y/145)
drawTree(i)
scale()


drawSittingBird
-position: Vector
transform: save
transform to
drawFeet
drawLegs
translate
let body: save
drawBody
translate
drawHead
drawEye
restorebody
drawbeak
restore


drawFlyingBird
-position: Vector
drawBezierLeft
drawBezierRight


drawBirds(nBirds: number)
let rabio: number: Math.random
let nSitting: number: rabio nBirds
let nFlying: number = Birds - nSitting
recolor
done
let x: random 10-740
let y: random ground-500
drawSitting Bird (T1)
set transform to (x,y) and scale y
drawFlying Bird (T1)
set transform/scale y
let x: random 10-740
let y: random 0-225
done
recolor


drawSnowflakes
-nFlakes: number
-position: vector
push
translate to -position
drawnFlakes
let x: number random
let y: number random
drawCloud
(x, y), {55}, 1, 10)
drawFlock
-position: Vector
-nBirds: number
transform
translate
drawStand
drawHouse
drawSillyBird(T)


drawSnowman
position: Vector
let r1: number = randomBetween(20,70)
let r2: number = r1/3
let r3: number = r2/3
let x1: Vector = {x: 0, y: r1}
let x2 = Vector = {x: 0, y: r1}
let x3 = Vector = {x: 0, y: r2}
let color: all thing color
let transform
translate
translate x1
drawCircle
translate to x2
drawCircle
translate to x3
drawCircle
translate
drawEye
translate
drawEye
rotate
set transform