Assignment 2

Emre Hancı, 21604552
Department of Computer Engineering
Hacettepe University
Ankara, Turkey
b21604552@cs.hacettepe.edu.tr

November 20, 2021

1 Introduction

In this assignment, there are two parts. In the first part there was a red square which rotate continuously with static speed. We need to provide 4 button which is "Toggle", "Slow Down", "Speed Up", "Random Color" and transform square to triangle with only border which has three different color on the corner points. Toggle button supposed to stop and start rotate animation. Slow Down button should decrease triangle rotate speed. Speed Up button should increase triangle rotate speed. Random Color should assign three different color to each corner point. In the second part the expectation from us was adding spin, scale and spiral animation to our emoji. Spin and spiral animation supposed to getting faster or slower according to user input.

2 Experiment

2.1 Part 1

In the Part 1 there was a project given by teaching assistant. According to the expectation, I made some changes in the rotatingSquare1.html file, four button added, which are toggle, slow down, speed up, random color. In the fragment shader there was oColor variable, I changed it to the take color from vertex shader. In the vertex shader I defined variable to take from javascript part. In the rotatingSquare1.js draw array changed from Triangle Strip to Line Loop, and vertices changed according to the points, and random color generator function add. With those change output comes the expectation.

2.2 Part 2

In the Part 2, According to the expectation, 6 button and 2 input number added. I have already write a function which scale emoji to given input in the

last assignment, and in the this assignment, I use that function for animation for scaling, and remove the limitation with 45 degree and -45 degree for the spinning animation. For the last step which is spiral animation, I couldn't provide any function, spiral animation is missing in the assignment. While rendering page, if spin animation is enable, rotate angle calculated by increasing with spinSpeed which is 1 at the default, and can be changed by the user. if scale animation is enable, points are reset to the default and according to the scaling direction, scale variable increase or decrease with angleStep which is multiplied by 0.01. After scale calculated, emoji redraw by calculated scale.

Table 1: Classes

Class Name	Attributes	Methods
Render	self, msLastFrame, callBack, run	constructor, start
ShaderUtil	No Attribute	start, initGL, domShaderSrc, createShader, createProgram, getLocations, initProgram, buffer, drawShape, clear, getConcatData, createYellowCircleArray, createBlackCircleArray, createColorArray, quadraticBezier, getCurvePositions, drawFace, getYellowColor, drawEyes, drawMask, drawMaskCorners, drawCurves, scaleDraw, scaleArray, scaleCurves, drawScene, onRender, getAngle, resetAngle, getTransformationMatrix, addEventListeners, parsePressedKey

Table 2: Methods in Render Class

Method Name	Input(s)	Output(s)	Info
constructor	callback		creates render object
start			runs the animation frame

3 Conclusion

As a conclusion part 2 in the assignment was so challenging, I had to search and learn how can I implement uniform animation and make rotation according to assignment paper. I learn those information and had an idea how its work, and also how create animation scenes implements one of the I learn in this assignment.

Table 3: Methods in Shader Class

Method Name	Input(s)	Output(s)	Info
start	canvasID, newScale		calls initGL, initProgram and starts the render
initGL	canvasID, newScale		gets gl and set scale and clear canvas
domShaderSrc	elmID		gets shader
createShader	src, type		compile shader
createProgram			attach shaders and link program
getLocations	program		gets uniform and attirib locations
initProgram	program		adds event listener and calls gl.useProgram
buffer	shape, shape_color		bind buffer and set buffer data and sub data
drawShape	shape, shape_buffer, shape_mode, vertex_number		draws the shape
clear			clear the canvas
getConcatData	radius, centers, j, i=0		gets circular data
createYellowCircleArray			creates yellow circle
createBlackCircleArray			creates black circle
createColorArray	color, loop		creates color array
quadraticBezier	p0, p1, p2, t		calculates quadratic bezier
getCurvePositions	p0, p1, p2		gets curve points
drawFace			draws face
getYellowColor			gets yellow color array according to angle on the animation
drawEyes			draws eyes
drawMask			draws mask
drawMaskCorners	buffer, loop		draws mask corners
drawCurves			draws mask curves
scaleDraw			scale the emoji elements
scalingAnimation			changes the draw according to the current angleStep
resetScale			makes emoji elements to default scale
scaleArray	array		calculates points by scale
scaleCurves			scales curves points
drawScene			draws uniforms and shapes
onRender	dt		cals getAngle if animation true
getAngleWithLimitation	dt		gets rotate angle according to delta time with degree limitation
getRotateAngle	dt		gets rotate angle according to delta time
getAngle	dt		gets angle according to delta time
resetAngle			reset angle to initial state
getTransformationMatrix			calculates