6주차 결과보고서

전공: 컴퓨터공학과 학년: 4학년 학번: 20192135 이름: 윤영인

**1.**

**1.1 Basic Shapes (1)**

(1) ofBackground(0): 바탕화면의 색을 검정색으로 설정한다.

(2) ofSetColor(255): 흰색으로 그림을 그린다.

(3) ofDrawRectangle(50,50,100,100): 좌표가 (50, 50) 위치에 가로가 100, 세로가 100인 정사각형을 그린다.

(4) ofDrawCircle(250, 100, 50): (250, 100)의 좌표를 중심으로 반지름이 50인 원을 그린다.

(5) ofDrawEllipse(400,100,80,100): (400,100)의 좌표를 중심으로 장축이 100, 단축이 80인 타원을 그린다.

(6) ofDrawTriangle(500,150,550,50,600,150): 좌표 (500,150), (550,50), (600,150)을 꼭지점으로 하는 삼각형을 그린다.

(7) ofDrawLine(700,50,700,150): 좌표 (700,50)에서 (700,150)까지 라인을 그린다.

**1.2 Basic Shapes (2)**

(1) ofBackground(0): 바탕화면의 색을 검정색으로 설정한다.

(2) ofSetColor(255): 흰색으로 그림을 그린다.

(3) ofFill(): 현재 색상으로 도형의 내부를 채운다.

(4) ofDrawRectangle(50,50,100,100): 좌표가 (50, 50) 위치에 가로가 100, 세로가 100인 정사각형을 그린다.

(5) ofDrawCircle(250, 100, 50): (250, 100)의 좌표를 중심으로 반지름이 50인 원을 그린다.

(6) ofDrawEllipse(400,100,80,100): (400,100)의 좌표를 중심으로 장축이 100, 단축이 80인 타원을 그린다.

(7) ofDrawTriangle(500,150,550,50,600,150): 좌표 (500,150), (550,50), (600,150)을 꼭지점으로 하는 삼각형을 그린다.

(8) ofDrawLine(700,50,700,150): 좌표 (700,50)에서 (700,150)까지 라인을 그린다.

(9) ofNoFill(): 도형의 내부를 채우지 않는다.

(10) ofDrawRectangle(50,200,100,100): 좌표가 (50,200) 위치에 가로가 100, 높이가 100인 정사각형을 그린다.

(11) ofDrawCircle(250,250,50): (250,250)의 좌표를 중심으로 반지름이 50인 원을 그린다.

(12) ofDrawEllipse(400,250,80,100): (400,250)의 좌표를 중심으로 장축이 100, 단축이 80인 타원을 그린다.

(13) ofDrawTriangle(500,300,550,200,600,300): 좌표 (500,300), (550,200), (600,300)을 꼭지점으로 하는 삼각형을 그린다.

(14) ofDrawLine(700,200,700,300): 좌표 (700,200)에서 (700,300)까지 라인을 그린다.

**1.3 Single Rectangle Brush: Using the Mouse(1) – draw()**

(1) if (ofGetMousePressed(OF\_MOUSE\_BUTTON\_LEFT)): 마우스의 왼쪽 버튼이 눌러진다면 아래의 항목을 실행한다.

(2) ofSetColor(255): 흰색으로 그림을 그린다.

(3) ofSetRectMode(OF\_RECTMODE\_CENTER): 사각형의 중심을 기준으로 사각형을 그리도록 설정한다.

(4) ofDrawRectangle(ofGetMouseX(), ofGetMouseY(), 50, 50): 현재 마우스 커서의 위치를 중심으로 가로가 50, 세로가 50인 정사각형을 그린다.

**1.4 Single Rectangle Brush: Using the Mouse(2) – setup()**

(1) ofSetBackgroundAuto(false): 자동으로 화면을 지우는 기능을 해제한다.

(2) ofBackground(0): 바탕 화면의 색상을 검정색으로 설정한다.

**1.5 Single Rectangle Brush: Using the Mouse(3) – draw()**

(1) if (ofGetMousePressed(OF\_MOUSE\_BUTTON\_LEFT): 마우스의 왼쪽 버튼이 눌러지면 아래의 항목들을 실행한다.

(2) float randomColor = ofRandom(50, 255): 50~255 사이의 값을 가지는 랜덤 색상을 float 형식의 randomColor 변수에 저장한다.

(3) ofSetColor(randomColor): randomColor 변수가 가지는 색상으로 도형을 그린다.

(4) ofSetRectMode(OF\_RECTMODE\_CENTER): 사각형의 중심을 기준으로 사각형을 그리도록 설정한다.

(5) ofDrawRectangle(ofGetMouseX(), ofGetMouseY(), 50, 50): 현재 마우스 커서의 위치를 중심으로 가로가 50, 세로가 50인 정사각형을 그린다.

**1.6 Bursting Rectangle Brush: Creating Randomized Bursts(1)**

**-** setup()

(1) ofSetFrameRate(60): 1초당 60개의 프레임을 설정한다.

(2) ofSetBackgroundAuto(false): 자동으로 화면을 지우는 기능을 해제한다.

(3) ofBackground(0): 바탕 화면의 색상을 검정색으로 설정한다.

- draw()

(1) if (ofGetMousePressed(OF\_MOUSE\_BUTTON\_LEFT): 마우스의 왼쪽 버튼이 눌러지면 아래의 항목들을 실행한다.

(2) ofSetRectMode(OF\_RECTMODE\_CENTER): 사각형의 중심을 기준으로 사각형을 그리도록 설정한다.

(3) for(int r = 0; r < numRects; r++): r이 0~numRects-1이 되는 동안 for문 안의 코드를 실행한다.

(4) ofSetColor(ofRandom(50, 255)): 50~255 사이의 값을 가지는 랜덤 색상으로 도형을 그린다.

(5) float width = ofRandom(5, 20)

float height = ofRandom(5, 20)

float xOffset = ofRandom(-40, 40)

float yOffset = ofRandom(-40, 40)

: float 타입을 가지는 width, height, xOffset, yOffset 변수에 5~20 또는 -40~40 사이의 값을 랜덤으로 지정하여 저장한다.

(6) ofDrawRectangle(ofGetMouseX() + xOffset, ofGetMouseY() + yOffset, width, height): 현재 마우스 커서의 위치에서 xOffset, yOffset 만큼을 더해준 위치에서 가로가 width, 세로가 height의 크기를 갖는 직사각형을 그린다.

**1.7 Glowing Circle Brush: Using Color**

(1) if (ofGetMousePressed(OF\_MOUSE\_BUTTON\_LEFT): 마우스의 왼쪽 버튼이 눌러지면 아래의 항목들을 실행한다.

(2) int maxRadius = 100: 브러쉬의 최대 두께

(3) int radiusStepSize = 5: 브러쉬의 원이 점점 작아지는 크기

(4) int alpha = 3: 브러쉬의 불투명도가 커지는 크기

(5) int maxOffsetDistance = 100: 원의 분포가 최대로 커지는 크기

(6) for (int radius = maxRadius; radius > 0; radius -= radiusStepSize): radius가 maxRadius에서 0까지 radiusStepSize만큼 작아지면서 for문 안의 코드를 실행한다.

(7) float angle = ofRandom(ofDegToRad(360.0)): 코드는 0부터 360도까지의 랜덤한 각도 값을 라디안 값으로 변환하여 float 타입의 angle 변수에 저장한다.

(8) float distance = ofRandom(maxOffsetDistance): 0~maxOffsetDistance 사이의 값을 랜덤으로 지정하여 float 타입의 distance 변수에 저장한다.

(9) float xOffset = cos(angle)\*distance

float yOffset = cos(angle)\*distance

: 극좌표에서 데카르트 좌표로 변환한다.

(10) ofColor myOrange(255, 132, 0, alpha): rgb를 (255, 132, 0)으로 갖는 주황색의 투명도를 alpha로 설정하여 myOrange라는 변수로 지정한다.

(11) ofColor myRed(255, 6, 0, alpha): rgb를 (255, 6, 0)으로 갖는 빨강색의 투명도를 alpha로 설정하여 myRed라는 변수로 지정한다.

(12) ofColor inBetween = myOrange.getLerped(myRed, ofRandom(1.0)): myOrange와 myRed 두 색상을 0부터 1.0까지의 랜덤한 가중치 값을 사용하여 선형 보간한 결과를 inBetween 변수에 저장하여 myOrange와 myRed 사이의 새로운 색상을 설정한다.

(13) ofSetColor(inBetween): inBetween 색상으로 도형을 그린다.

(14) ofCircle(ofGetMouseX() + xOffset, ofGetMouseY() + yOffset, radius): 현재 마우스의 커서에서 xOffset, yOffset 만큼 더해준 위치에서 radius의 반지름의 크기를 갖는 원을 그린다.

**1.8 Fleeing Triangle Brush: Vectors**

(1) if (ofGetMousePressed(OF\_MOUSE\_BUTTON\_LEFT): 마우스의 왼쪽 버튼이 눌러지면 아래의 항목들을 실행한다.

(2) for (int t = 0; t < numTriangles; ++t): t가 0~numTriangles(10) 사이의 값일 동안 증가시키면서 for문 안의 코드를 실행한다.

(3) float offsetDistance = ofRandom(minOffset, maxOffset) : minOffset(5)에서 maxOffset (70) 사이의 값을 가지는 랜덤 값을 지정하여 float 타입의 offsetDistance 변수에 저장한다.

(4) ofVec2f mousePos(ofGetMouseX(), ofGetMouseY()): 현재 마우스의 커서 위치를 2차원 벡터로 표현하여 mousePos 변수에 저장한다.

(5) ofVec2f p1(0, 25.0): 벡터의 x좌표가 0, y좌표가 25.0으로 설정된 2차원 벡터를 p1에 저장한다.

(6) ofVec2f p2(100, 0): 벡터의 x좌표가 100, y좌표가 0으로 설정된 2차원 벡터를 p2에 저장한다.

(7) ofVec2f p3(0, -25.0): 벡터의 x좌표가 0, y좌표가 -25.0으로 설정된 2차원 벡터를 p3에 저장한다.

(8) float rotation = ofRandom(360): 0~360 사이의 값을 랜덤으로 지정하여 rotation에 저장한다.

(9) p1.rotate(rotation)

p2.rotate(rotation)

p3.rotate(rotation)

: p1, p2, p3 벡터의 회전시킬 각도를 지정한다.

(10) p1 += mousePos

p2 += mousePos

p3 += mousePos

: p1, p2, p3의 위치를 현재 마우스가 위치한 곳으로 옮긴다.

(11) ofSetColor(255, 50): (255, 50) 색상으로 도형을 그린다.

(12) ofTriangle(p1.x, p1.y, p2.x, p2.y, p3.x, p3.y): p1, p2, p3를 꼭짓점으로 갖는 삼각형을 그린다.

(13) if (ofGetMousePressed(OF\_MOUSE\_BUTTON\_RIGHT)): 마우스의 오른쪽 버튼이 눌러지면 아래의 코드를 실행한다.

(14) ofBackground(0): 바탕 화면의 색상을 검정색으로 지정한다.

**1.9 update**

- setup()

(1) xPos = 5: 수평 시작의 위치를 5로 설정한다.

(2) ofBackground(ofColor::black): 바탕 화면의 색상을 검정색으로 설정한다.

-update()

(1) xPos += 2 : update 함수가 실행되면서 수평시작의 위치를 2만큼 이동한다.

(2) if (ofGetWidth() < xPos): 현재 사용중인 창의 너비가 xPos보다 작으면 아래의 코드를 실행한다.

(3) xPos = 5: xPos를 5로 지정한다.

- draw()

(1) ofSetColor(ofColor::red): 빨간색으로 도형을 그린다.

(2) ofDrawCircle(xPos, 100, 10): 좌표 (xPos, 100)의 위치에서 반지름이 10인 원을 그린다.