**【游戏编程实验报告 文件命名要求：班级-学号-姓名-n（n为Lab的编号）】**

**班级 NZ222 姓名 霍梓航 学号 225858**

**Labs 3**

**【文件首部格式：小三号、黑体、加粗、居中、1.5倍行距 】**

**实验内容**

1) Smiley Face

*package com.hamhuo.massey*;

*import java.awt.*\*;

*import java.awt.geom.*\*;

*import javax.swing.*\*;

*public class SmileyDrawing extends JPanel* {

*//-------------------------------------------------------*

*// Useful Functions for Drawing things on the screen*

*//-------------------------------------------------------*

*//My Definition of some colors*

*Color* black = *Color*.BLACK;

*Color* red = *Color*.RED;

*Color* blue = *Color*.BLUE;

*Color* green = *Color*.GREEN;

*Color* white = *Color*.WHITE;

*//Changes the background Color to the color c*

*public void* changeBackgroundColor(Graphics *g*, *Color c*) {

Graphics2D g2d = (Graphics2D)*g*;

g2d.setBackground(*c*);

}

*//Changes the background Color to the color (red,green,blue)*

*public void* changeBackgroundColor(Graphics *g*, *int red*, *int green*, *int blue*) {

Graphics2D g2d = (Graphics2D)*g*;

g2d.setBackground(*new* Color(*red*,*green*,*blue*));

}

*//Clears the background, makes the whole window whatever the background color is*

*public void* clearBackground(Graphics *g*, *int width*, *int height*) {

Graphics2D g2d = (Graphics2D)*g*;

g2d.clearRect(0, 0, *width*, *height*);

}

*//Changes the drawing Color to the color c*

*public void* changeColor(Graphics *g*, *Color c*) {

*g*.setColor(*c*);

}

*//Changes the drawing Color to the color (red,green,blue)*

*public void* changeColor(Graphics *g*, *int red*, *int green*, *int blue*) {

*g*.setColor(*new* Color(*red*,*green*,*blue*));

}

*//Functions to Draw Text on a window*

*//Takes a Graphics g, position (x,y) and some text*

*public void* drawText(Graphics *g*, *int x*, *int y*, *String s*) {

Graphics2D g2d = (Graphics2D)*g*;

g2d.setFont(*new* Font("Arial", *Font*.BOLD, 40));

g2d.drawString(*s*, *x*, *y*);

}

*//This function draws a line from (x1,y2) to (x2,y2)*

*void* drawLine(Graphics *g*, *int x1*, *int y1*, *int x2*, *int y2*) {

Graphics2D g2d = (Graphics2D)*g*;

g2d.draw(*new* Line2D.Double(*x1*, *y1*, *x2*, *y2*));

}

*//This function draws a rectangle at (x,y) with width and height*

*void* drawRectangle(Graphics *g*, *int x*, *int y*, *int width*, *int height*) {

Graphics2D g2d = (Graphics2D)*g*;

g2d.draw(*new* Rectangle2D.Double(*x*, *y*, *width*, *height*));

}

*//This function draws a rectangle at (x,y) with width and height*

*void* drawSolidRectangle(Graphics *g*, *int x*, *int y*, *int width*, *int height*) {

Graphics2D g2d = (Graphics2D)*g*;

g2d.fill(*new* Rectangle2D.Double(*x*, *y*, *width*, *height*));

}

*//This function draws a circle at (x,y) with radius*

*void* drawCircle(Graphics *g*, *int x*, *int y*, *double radius*) {

Graphics2D g2d = (Graphics2D)*g*;

g2d.draw(*new* Ellipse2D.Double(*x*-*radius*, *y*-*radius*, *radius*\*2, *radius*\*2));

}

*//This function draws a solid circle at (x,y) with radius*

*void* drawSolidCircle(Graphics *g*, *int x*, *int y*, *double radius*) {

Graphics2D g2d = (Graphics2D)*g*;

g2d.fill(*new* Ellipse2D.Double(*x*-*radius*, *y*-*radius*, *radius*\*2, *radius*\*2));

}

*//Function to create the window and display it*

*public void* setupWindow(*int width*, *int height*) {

*JFrame* frame = *new* JFrame();

frame.setSize(*width*, *height*);

frame.setLocation(200,200);

frame.setTitle("Window");

frame.setDefaultCloseOperation(*JFrame*.EXIT\_ON\_CLOSE);

frame.add(*this*);

frame.setVisible(*true*);

*//Resize the window (insets are just the boards that the Operating System puts on the board)*

*Insets* insets = frame.getInsets();

frame.setSize(*width* + insets.left + insets.right,

*height* + insets.top + insets.bottom);

}

*//Main function that takes care of some Object Oriented stuff*

*public static void* main(*String args*[]) {

*SmileyDrawing* w = *new* SmileyDrawing();

}

*//-------------------------------------------------------*

*// Your Program*

*//-------------------------------------------------------*

*public* SmileyDrawing() {

setupWindow(500,500);

}

*//This gets called any time the Operating System*

*//tells the program to paint itself*

*public void* paintComponent(Graphics *g*) {

changeBackgroundColor(*g*, white);

clearBackground(*g*, 500, 500);

changeColor(*g*, black);

drawText(*g*, 50, 50, "Hello World");

*// 画黄色的主体（吃豆人）*

changeColor(*g*, *Color*.YELLOW);

drawSolidCircle(*g*, 200, 200, 100); *// 吃豆人的头*

*// 画嘴巴（用两条线段形成一个开口）*

changeColor(*g*, black);

drawSolidRectangle(*g*, 200, 200, 100, 50);

drawLine(*g*, 200, 200, 250, 150); *// 上嘴角*

drawLine(*g*, 200, 200, 250, 250); *// 下嘴角*

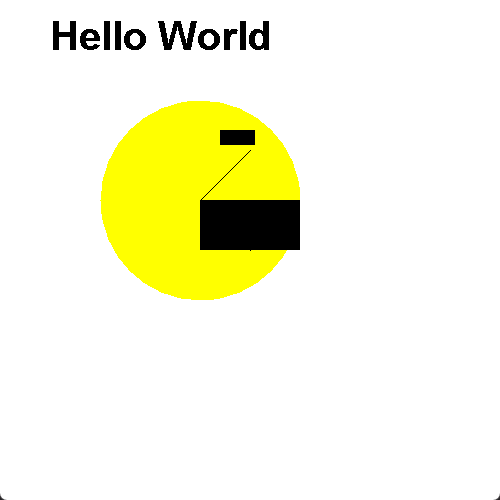
*// 画眼睛（用小矩形表示）*

changeColor(*g*, black);

drawSolidRectangle(*g*, 220, 130, 35, 15); *// 眼睛*

}

}



2. 滚小球

*package com.hamhuo.massey*;

*import java.awt.*\*;

*import java.awt.geom.*\*;

*import javax.swing.*\*;

*import java.awt.event.*\*;

*public class BouncingBall extends JPanel implements KeyListener* {

*// -------------------------------------------------------*

*// Useful Functions for Drawing things on the screen*

*// -------------------------------------------------------*

*// My Definition of some colors*

*double* backgroundDepth = 0;

*Color* black = *Color*.BLACK;

*Color* red = *Color*.RED;

*Color* blue = *Color*.BLUE;

*Color* green = *Color*.GREEN;

*Color* white = *Color*.WHITE;

*// Changes the background Color to the color c*

*public void* changeBackgroundColor(Graphics *g*, *Color c*) {

Graphics2D g2d = (Graphics2D) *g*;

g2d.setBackground(*c*);

}

*// Changes the background Color to the color (red,green,blue)*

*public void* changeBackgroundColor(Graphics *g*, *int red*, *int green*, *int blue*) {

Graphics2D g2d = (Graphics2D) *g*;

g2d.setBackground(*new* Color(*red*, *green*, *blue*));

}

*// Clears the background, makes the whole window whatever the background color is*

*public void* clearBackground(Graphics *g*, *int width*, *int height*) {

Graphics2D g2d = (Graphics2D) *g*;

g2d.clearRect(0, 0, *width*, *height*);

}

*// Changes the drawing Color to the color c*

*public void* changeColor(Graphics *g*, *Color c*) {

*g*.setColor(*c*);

}

*// Changes the drawing Color to the color (red,green,blue)*

*public void* changeColor(Graphics *g*, *int red*, *int green*, *int blue*) {

*g*.setColor(*new* Color(*red*, *green*, *blue*));

}

*// This function draws a rectangle at (x,y) with width and height*

*void* drawRectangle(Graphics *g*, *double x*, *double y*, *double width*, *double height*) {

Graphics2D g2d = (Graphics2D) *g*;

g2d.draw(*new* Rectangle2D.Double(*x*, *y*, *width*, *height*));

}

*// This function fills a rectangle at (x,y) with width and height*

*void* drawSolidRectangle(Graphics *g*, *double x*, *double y*, *double width*, *double height*) {

Graphics2D g2d = (Graphics2D) *g*;

g2d.draw(*new* Rectangle2D.Double(*x*, *y*, *width*, *height*));

}

*// This function draws a rectangle at (x,y) with width and height*

*void* drawCircle(Graphics *g*, *double x*, *double y*, *double radius*) {

Graphics2D g2d = (Graphics2D) *g*;

g2d.fill(*new* Ellipse2D.Double(*x* - *radius*, *y* - *radius*, *radius* \* 2, *radius* \* 2));

}

*// This function draws a rectangle at (x,y) with width and height*

*void* drawSolidCircle(Graphics *g*, *double x*, *double y*, *double radius*) {

Graphics2D g2d = (Graphics2D) *g*;

g2d.fill(*new* Ellipse2D.Double(*x* - *radius*, *y* - *radius*, *radius* \* 2, *radius* \* 2));

}

*// Functions to Draw Text on a window*

*// Takes a Graphics g, position (x,y) and some text*

*public void* drawText(Graphics *g*, *double x*, *double y*, *String s*) {

Graphics2D g2d = (Graphics2D) *g*;

g2d.setFont(*new* Font("Arial", *Font*.BOLD, 40));

g2d.drawString(*s*, (*int*) *x*, (*int*) *y*);

}

*// Translate Function, moves the drawing context*

*// by (x,y)*

*public void* translate(Graphics *g*, *int x*, *int y*) {

Graphics2D g2d = (Graphics2D) *g*;

g2d.translate(*x*, *y*);

}

*// Rotate Function, rotates the drawing context by angle*

*public void* rotate(Graphics *g*, *double angle*) {

Graphics2D g2d = (Graphics2D) *g*;

g2d.rotate(*Math*.*toRadians*(*angle*));

}

*AffineTransform* transform = *null*;

*public void* saveTransform(Graphics *g*) {

Graphics2D g2d = (Graphics2D) *g*;

transform = g2d.getTransform();

}

*//Restores the last transform*

*public void* restoreTransform(Graphics *g*) {

Graphics2D g2d = (Graphics2D) *g*;

*if* (transform != *null*) {

g2d.setTransform(transform);

}

}

*// Converts an integer to a string*

*public String* intToString(*int i*) {

*return new* Integer(*i*).toString();

}

*// Converts an float to a string*

*public String* floatToString(*float f*) {

*return new* Float(*f*).toString();

}

*// Function to create the window and display it*

*public void* setupWindow(*int width*, *int height*) {

*JFrame* frame = *new* JFrame();

frame.setSize(*width*, *height*);

frame.setLocation(200, 200);

frame.setTitle("Window");

frame.setDefaultCloseOperation(*JFrame*.EXIT\_ON\_CLOSE);

frame.add(*this*);

frame.setVisible(*true*);

frame.addKeyListener(*this*);

*//todo : how to resize background*

frame.setResizable(*true*);

*//Q*

setDoubleBuffered(*true*);

*// Resize the window (insets are just the boards that the Operating System puts on the board)*

*Insets* insets = frame.getInsets();

frame.setSize(*width* + insets.left + insets.right, *height* + insets.top + insets.bottom);

}

*// Returns the time in milliseconds*

*public long* getTime() {

*return System*.*currentTimeMillis*();

}

*// Waits for ms milliseconds*

*public void* sleep(*double ms*) {

*try* {

*Thread*.*sleep*((*long*) *ms*);

} *catch* (*Exception e*) {

*// Do Nothing*

}

}

*// Main function that takes care of some Object-Oriented stuff*

*public static void* main(*String*[] *args*) {

*BouncingBall* w = *new* BouncingBall();

}

*// Very simple way of controlling the framerate*

*// calculate frame time between two beginning*

*double* lastBeginTime = getTime();

*public double* simpleFramerate(*double framerate*) {

*double* currentTime = getTime();

*// gap time between two frame*

*double* deltaTime = (currentTime - lastBeginTime);

*// update current begin time*

lastBeginTime = currentTime;

*//target time*

*// 目标帧时间（毫秒）*

*double* targetDeltaTime = 1000.0 / *framerate*; *// 计算目标帧间隔*

*// 计算 sleep 时间*

*long* sleepTime = (*long*) (targetDeltaTime - deltaTime);

*if* (sleepTime > 0) { *// 避免负数*

*try* {

*Thread*.*sleep*(sleepTime);

} *catch* (*InterruptedException e*) {

*throw new* RuntimeException(*e*);

}

}

*return* deltaTime;

}

*// -------------------------------------------------------*

*// Your Program*

*// -------------------------------------------------------*

*public* BouncingBall() {

*// Create a window of size 500x500*

setupWindow(500, 500);

*while* (*true*) {

*// Control the framerate based on frames*

*double* dt = simpleFramerate(240);

*// Update the Game*

update(dt);

*// Tell the window to paint itself*

repaint();

}

}

*int* direction = 1; *// 1 表示变深，-1 表示变浅*

*double* acceleration = 9.8;

*// update position*

*// todo how to sync background and circle*

*public void* update(*double dt*) {

*// This function updates your game*

*// 背景颜色加深*

backgroundDepth += direction \* *dt* \* 0.05; *// 控制变换速度*

*// 反转方向*

*if* (backgroundDepth >= 255) {

backgroundDepth = 255;

direction = -1; *// 变浅*

} *else if* (backgroundDepth <= 90) {

backgroundDepth = 90;

direction = 1; *// 变深*

}

*// 外圈缩小*

*if* (scale > 0) {

scale -= *dt* \* 0.0001;

}

*// border*

*int* bottom = 500;

centerY += (direction)\*(0.5)\*(acceleration)\**dt*\*0.09;

*if*(centerY + scaledOuterRadius > bottom) {

centerY = bottom - scaledOuterRadius;

direction = -1;

}*else if*(centerY + scaledOuterRadius < 15) {

centerY = 15 + scaledOuterRadius;

direction = 1;

}

*int* direction2 = 0;

centerX += (direction)\*(0.5)\*(acceleration)\**dt*\*0.4;

*if*(centerX + scaledOuterRadius > bottom) {

centerX = bottom - scaledOuterRadius;

direction2 = -1;

}*else if*(centerX + scaledOuterRadius < 0) {

centerX = 0 + scaledOuterRadius;

direction2 = 1;

}

}

*// This gets called any time the Operating System*

*// tells the program to paint itself*

*// update display*

*double* scale = 1;

*double* outerRadius = 100;

*double* innerRadius = 50;

*// todo try uncoupling circle*

*// circle center*

*double* centerX = 250, centerY = 250;

*double* scaledOuterRadius = 1;

*public void* paintComponent(Graphics *g*) {

*// 设置背景颜色，随时间变深*

changeBackgroundColor(*g*, 0, 0, (*int*) backgroundDepth);

clearBackground(*g*, 500, 500);

scaledOuterRadius = outerRadius \* scale;

*// 画黑色缩小圆圈*

changeColor(*g*, black);

*if* (scaledOuterRadius > innerRadius) {

drawCircle(*g*, centerX, centerY, scaledOuterRadius);

} *else* {

scale = 1;

}

*// 画白色外圈*

changeColor(*g*, white);

drawSolidCircle(*g*, centerX, centerY, innerRadius + 10);

*// 画红色实心圆*

changeColor(*g*, red);

drawSolidCircle(*g*, centerX, centerY, innerRadius);

*// 画数字 "1"*

changeColor(*g*, black);

drawText(*g*, centerX - 10, centerY + 10, "1");

}

*// Called whenever the user presses a key*

*public void* keyPressed(*KeyEvent e*) {

}

*// Called whenever the user releases a key*

*public void* keyReleased(*KeyEvent e*) {

}

*// Called whenever the user presses and releases a key*

*public void* keyTyped(*KeyEvent e*) {

}

}

**二、遇到的问题**

**笑脸画的太丑**