

Java snake game project

6530300619 นายอติเทพ ชุสุวรรณ

03603112 Programming Fundamentals II

ผศ.ดร.กุลวดี สมบูรณ์วิวัฒน์

ปีการศึกษา 2565



Introduction

ชื่อ : นาย อติเทพ ชุสุวรรณ
สาขา : วิศวกรรมคอมพิวเตอร์
รหัสนิสิต : 6530300619

เหตุผลในการทำโปรเจค

เกมเป็นเกมคลาสสิกที่มีความนิยมมากในช่วงต้นของเกมคอมพิวเตอร์ ซึ่งเป็นเกมที่เล่นง่ายแต่มีความท้าทายในการเล่น นอกจากนี้ เกมยังเป็นเกมที่เหมาะสมสำหรับการฝึกความสามารถในการควบคุมการเคลื่อนไหวของวัตถุในเกม และการวางแผนเพื่อหลีกเลี่ยงอุปสรรคต่างๆ

ดังนั้น การทำโปรเจคเกมนี้จึงเป็นโอกาสที่ดีในการฝึกฝนทักษะในการเขียนโปรแกรมเกี่ยวกับ **user interface** และการจัดการกับเหตุการณ์ต่างๆ ในการพัฒนาเกม และยังสามารถฝึกทักษะในการวางแผนและการแก้ปัญหา เพื่อให้เกมทำงานได้ตามที่ต้องการ และยังสามารถฝึกความคิดสร้างสรรค์ในการปรับแต่งเกมให้มีความสนุกสนานและน่าเล่นมากยิ่งขึ้นได้

Algorithm

1. **GamePanel.java** ไฟล์นี้เป็นคลาสที่เป็นผู้ใช้งานหน้าจอเกม ซึ่งเป็นส่วนหนึ่งของ user interface ในการแสดงผลต่างๆ เช่น สีพื้นหลังและรูปภาพของงู ในไฟล์นี้คุณสร้างการจัดการสิ่งต่างๆ ที่เกี่ยวกับการเคลื่อนที่ของงู รวมถึงการตรวจจับการชนกับอาหารและขอบเขตของเกม
2. **GameFrame.java** - ไฟล์นี้เป็นคลาสที่แสดงหน้าต่างของเกม ซึ่งเป็นส่วนหนึ่งของ user interface ในการแสดงผลต่างๆ เช่น ปุ่มเริ่มเกม และคะแนนที่ผู้เล่นได้รับ ในไฟล์นี้สร้างการจัดการสิ่งต่างๆ ที่เกี่ยวกับการเริ่มเกมและการจัดการคะแนน
3. **SnakeGame.java** - ไฟล์นี้เป็นคลาสหลักของโปรเจ็ค ซึ่งเป็นส่วนที่ควบคุมการทำงานของเกม ในไฟล์นี้สร้างการเรียกใช้ GameFrame และ GamePanel เพื่อเริ่มเกม รวมถึงการสร้างการจัดการกับปุ่มเริ่มเกมและการหยุดเกม

GamePanel.java

Code ส่วนที่1

- Import แพคเกจที่จำเป็นต่อการใช้งาน
- ประกาศคลาส GamePanel และ implement interface ActionListener
- ระบุคลาสเป็น private และทำให้โค้ดของคลาสนั้นสามารถ serialize และ deserialize ได้ถูกต้อง
- ประกาศตัวแปรชนิดอาร์เรย์ของจำนวนเต็ม (int) โดยทั้ง x และ y มีขนาดเท่ากับค่าคงที่
- ประกาศตัวแปลทั้งหมดที่ใช้ในเกม

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.util.Random;
import javax.swing.JPanel;

public class GamePanel extends JPanel implements ActionListener{

    private static final long serialVersionUID = 1L;

    static final int WIDTH = 500;
    static final int HEIGHT = 500;
    static final int UNIT_SIZE = 20;
    static final int NUMBER_OF_UNITS = (WIDTH * HEIGHT) / (UNIT_SIZE * UNIT_SIZE);

    final int x[] = new int[NUMBER_OF_UNITS];
    final int y[] = new int[NUMBER_OF_UNITS];

    int length = 5;
    int foodEaten;
    int foodX;
    int foodY;
    char direction = 'D';
    boolean running = false;
    Random random;
    Timer timer;
```

GamePanel.java

Code ส่วนที่2

- ประกาศ constructor ของคลาส GamePanel
- ประกาศ เมธอด play() เพื่อเริ่มเกม
- ประกาศ เมธอด paintComponent() ซึ่งเป็นเมธอดที่ถูกเรียกโดยโค้ดของ Java โดยอัตโนมัติเมื่อต้องการวาดบน JPanel
- ประกาศเมธอดที่ใช้ในการเคลื่อนที่ของงู

```
GamePanel() {
    random = new Random();
    this.setPreferredSize(new Dimension(WIDTH, HEIGHT));
    this.setBackground(Color.black);
    this.setFocusable(true);
    this.addKeyListener(new MyKeyAdapter());
    play();
}

public void play() {
    addFood();
    running = true;

    timer = new Timer(delay:80, this);
    timer.start();
}

@Override
public void paintComponent(Graphics graphics) {
    super.paintComponent(graphics);
    draw(graphics);
}

public void move() {
    for (int i = length; i > 0; i--) {
        x[i] = x[i-1];
        y[i] = y[i-1];
    }

    if (direction == 'L') {
        x[0] = (x[0] - UNIT_SIZE + WIDTH) % WIDTH;
    } else if (direction == 'R') {
        x[0] = (x[0] + UNIT_SIZE) % WIDTH;
    } else if (direction == 'U') {
        y[0] = (y[0] - UNIT_SIZE + HEIGHT) % HEIGHT;
    } else {
        y[0] = (y[0] + UNIT_SIZE) % HEIGHT;
    }
}
```

GamePanel.java

Code ส่วนที่3

- checkFood() ใช้สำหรับตรวจสอบว่าูกินอาหารหรือยัง
- ต่อมาเป็นการวาดภาพของเกม โดยทำการใช้ Graphics เพื่อวาดหน้าจอเกม
- ประกาศ method ที่ใช้สำหรับสุมตำแหน่งของอาหาร (food)
- ต่อมาตรวจสอบว่าูกสามารถโดนชนเข้ากับตัวเองหรือเข้ากับกำแพงหรือไม่

```
public void checkFood() {
    if(x[0] == foodX && y[0] == foodY) {
        length++;
        foodEaten++;
        addFood();
    }
}

public void draw(Graphics graphics) {
    if(!running) {
        graphics.setColor(new Color(0, 0, 0));
        graphics.fillRect(foodX, foodY, UNIT_SIZE, UNIT_SIZE);
        graphics.setColor(Color.blue);
        graphics.fillRect(x[0], y[0], UNIT_SIZE, UNIT_SIZE);

        for (int i = 1; i < length; i++) {
            graphics.setColor(new Color(0, 0, 0));
            graphics.fillRect(x[i], y[i], UNIT_SIZE, UNIT_SIZE);
        }

        graphics.setColor(Color.white);
        graphics.setFont(new Font("Sans serif", Font.ROMAN_BASELINE, 12));
        FontMetrics metrics = getFontMetrics(graphics.getFont());
        graphics.drawString("Score: " + FoodEaten, (WIDTH - metrics.stringWidth("Score: " + FoodEaten)) / 2, graphics.getFont().getSize());
    } else {
        gameOver(graphics);
    }
}

public void addFood() {
    foodX = random.nextInt((int)(WIDTH / UNIT_SIZE))*UNIT_SIZE;
    foodY = random.nextInt((int)(HEIGHT / UNIT_SIZE))*UNIT_SIZE;
}

public void checkHit() {
    // check if head run into its body
    for (int i = length; i > 0; i--) {
        if (x[0] == x[i] && y[0] == y[i]) {
            running = false;
        }
    }

    // check if head run into walls
    if (x[0] < 0 || x[0] > WIDTH || y[0] < 0 || y[0] > HEIGHT) {
        running = false;
    }

    if(!running) {
        timer.stop();
    }
}
```

GamePanel.java

Code ส่วนที่4

- ประกาศ method ที่ถูกเรียกเมื่อเกมเสร็จสิ้นหรือ Game Over แล้ว
- ประกาศ method ที่เกี่ยวกับการตอบสนองกับการกระทำในเกม
- สร้าง inner class ชื่อ MyKeyAdapter ซึ่งเป็น subclass ของ KeyAdapter โดยมีเมทอด keyPressed ซึ่งจะถูกเรียกเมื่อมีการกดปุ่มบนคีย์บอร์ด

```
public void gameOver(Graphics graphics) {
    graphics.setColor(Color.red);
    graphics.setFont(new Font(name:"Sans serif", Font.ROMAN_BASELINE, size:50));
    FontMetrics metrics = getFontMetrics(graphics.getFont());
    graphics.drawString(str:"Game Over", (WIDTH - metrics.stringWidth(str:"Game Over")) / 2, HEIGHT / 2);

    graphics.setColor(Color.white);
    graphics.setFont(new Font(name:"Sans serif", Font.ROMAN_BASELINE, size:25));
    metrics = getFontMetrics(graphics.getFont());
    graphics.drawString("Score: " + foodEaten, (WIDTH - metrics.stringWidth("Score: " + foodEaten)) / 2, graphics.getFont().getSize());
}

@Override
public void actionPerformed(ActionEvent arg0) {
    if (running) {
        move();
        checkFood();
        checkHit();
    }
    repaint();
}

public class MyKeyAdapter extends KeyAdapter {
    @Override
    public void keyPressed(KeyEvent e) {
        switch(e.getKeyCode()) {
            case KeyEvent.VK_LEFT:
                if (direction != 'R') {
                    direction = 'L';
                }
                break;

            case KeyEvent.VK_RIGHT:
                if (direction != 'L') {
                    direction = 'R';
                }
                break;

            case KeyEvent.VK_UP:
                if (direction != 'D') {
                    direction = 'U';
                }
                break;

            case KeyEvent.VK_DOWN:
                if (direction != 'U') {
                    direction = 'D';
                }
                break;
        }
    }
}
```


GameFrame.java

Code ส่วนที่1

- Import แพคเกจที่จำเป็นต่อการใช้งาน
- ประกาศ class GameFrame ซึ่ง extends จาก JFrame ซึ่งเป็นคลาสของ Java Swing ที่ใช้สำหรับสร้างหน้าต่างแอปพลิเคชัน
- สร้าง คอนสตรัคเตอร์ของคลาส GameFrame ซึ่งเป็น JFrame ที่สร้างหน้าต่างของเกม

```
import javax.swing.JFrame;

public class GameFrame extends JFrame{

    private static final long serialVersionUID = 1L;

    GameFrame() {
        GamePanel panel = new GamePanel();
        this.add(panel);
        this.setTitle(title:"snake");
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setResizable(resizable:false);
        this.pack();
        this.setVisible(b:true);
        this.setLocationRelativeTo(c:null);
    }
}
```

SnakeGame.java

Code ส่วนที่1

- โค้ดส่วนโปรแกรมหลักที่เรียกใช้เมื่อต้องการเริ่มเล่นเกม

```
public class SnakeGame {  
    Run | Debug  
    public static void main(String[] args) {  
        new GameFrame();  
    }  
}
```

UI

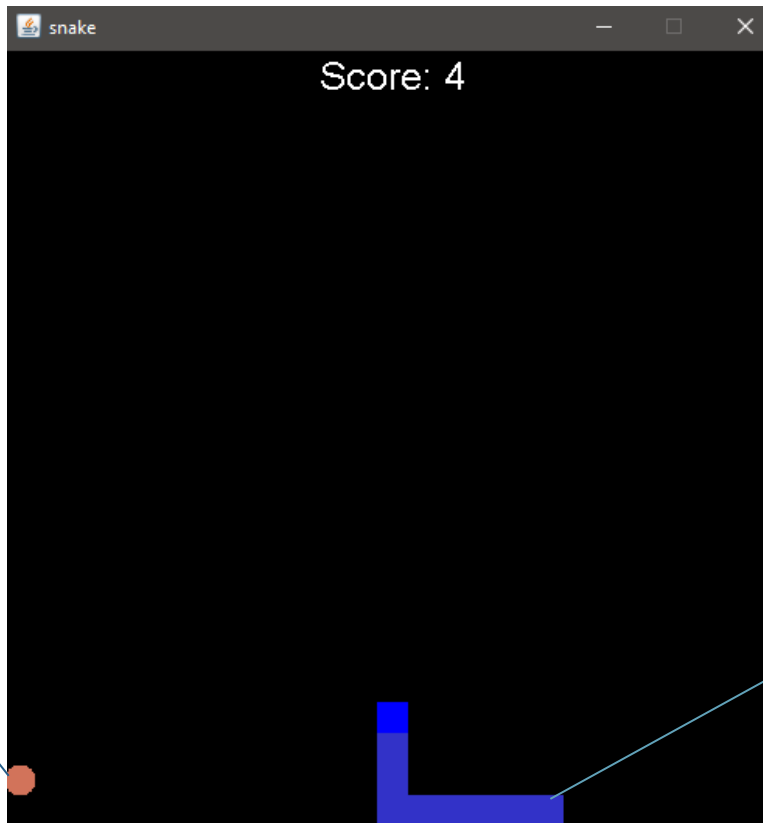
GameFrame

GamePanel

Food

GamePanel

Snake



วิธีการติดตั้งโปรแกรม

จำเป็นต้องติดตั้งโปรแกรมที่รองรับและสามารถรันภาษา JAVA ได้ เพื่อให้ได้ลองเล่น Snake game ที่ผู้จัดทำได้ สร้างขึ้นมาได้ จึงจำเป็นต้องโปรแกรม 2 ตัวนี้ ได้แก่

- Visual Studio Code โปรแกรมสำหรับพัฒนาซอฟต์แวร์นอกจากนี้ ยังมีคุณสมบัติที่ช่วยในการจัดการโค้ด ดัดแปลงโค้ด ทดสอบ และ ตรวจสอบ ข้อผิดพลาด
- JDK (Java Development Kit) เพื่อใช้ในการพัฒนาและรัน โปรแกรมภาษา Java

Thank you :)

ลิงก์ youtube : [\(83\) Boom Java project presentation - YouTube](#)

ลิงก์ github : [LilBoom04/Boom \(github.com\)](#)