



Mini Project

เกม Racing Car

โดย

นางสาวพริภา กรสุวรรณ

รหัสนักศึกษา 6404062620168 ตอนเรียนที่ 3

รายงานนี้เป็นส่วนหนึ่งของวิชา 040613204 Object Oriented Programming

ภาควิชาวิทยาการคอมพิวเตอร์และสารสนเทศ คณะวิทยาศาสตร์ประยุกต์

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

ภาคการศึกษาที่ 1 ปีการศึกษา 2566

สารบัญ

	หน้า
บทที่ 1	1
ที่มาและความสำคัญ.....	1
ประเภทของโครงการ.....	1
ประโยชน์ที่ได้รับ.....	1
ขอบเขตของโครงการ.....	1
ตารางแผนการทำงาน.....	4
บทที่ 2	5
วิธีการเล่น.....	5
คลาสไดอะแกรม.....	6
รูปแบบการพัฒนา	7
Constructor.....	7
Encapsulation	7
Inheritance.....	8
Event handling.....	9
อัลกอริทึมที่สำคัญในโปรแกรม.....	10
บทที่ 3	11
ปัญหาที่พบบ่อยระหว่างการพัฒนา.....	11
จุดเด่นของโปรแกรมที่ไม่เหมือนใคร	11
คำแนะนำ	11

บทที่ 1

บทนำ

ที่มาและความสำคัญ

เกมที่ได้จัดทำได้สร้างขึ้นได้แรงบันดาลใจมาจากเกมแข่งรถที่เป็นตุ้เกมในห้างสรรพสินค้า ที่ทุกคนต้องรู้จักเพราะเมื่อในอดีตร้านเกม ร้านตุ้เกมเป็นที่นิยมเป็นอย่างมาก อีกทั้งเกมแข่งรถก็ยังเป็นเกมคลายเครียดของใครหลายคน แต่ตัวเกมมีข้อจำกัดเพราะต้องเล่นที่ตุ้เกมหรือต้องออกไปห้างสรรพสินค้าจนถึงปัจจุบันผู้จัดทำจึงอยากที่จะปรับเปลี่ยน ออกแบบเกมที่มีสีสันมากขึ้น หลากหลายมากขึ้น และสามารถเล่นที่บ้านได้

จึงได้สร้างเกมที่มีลักษณะคล้ายกับเกมดังกล่าวขึ้นด้วยภาษา Java เพื่อพัฒนาโปรแกรมให้มีภาพที่สวยงามและมีมิติมากขึ้น

ประเภทของโครงการ

เกมประเภทอาเคด เล่นคนเดียวโดยไม่ใช้อินเทอร์เน็ต

ประโยชน์ที่ได้รับ

1. เพื่อฝึกไหวพริบในการมองเห็น
2. เพื่อความเพลินเพลิน
3. เพื่อฝึกสมาธิ

ขอบเขตของโครงการ

1. รายละเอียดภายในเกม

เกม Racing Car เป็นเกมที่ให้ผู้เล่นควบคุมรถเพื่อหนีด่านตำรวจและรถตำรวจที่พยายามจะไล่ล่ารถของผู้เล่นที่ควบคุมโดยผู้เล่น

2. วิธีเล่น

ใช้คีย์บอร์ดเพื่อควบคุมทิศทางของรถและคอยหลบหนีตำรวจ

3. เนื้อหาของเกม

ตัวละคร



รถ



รถตำรวจ



ที่กั้นด่านตรวจ



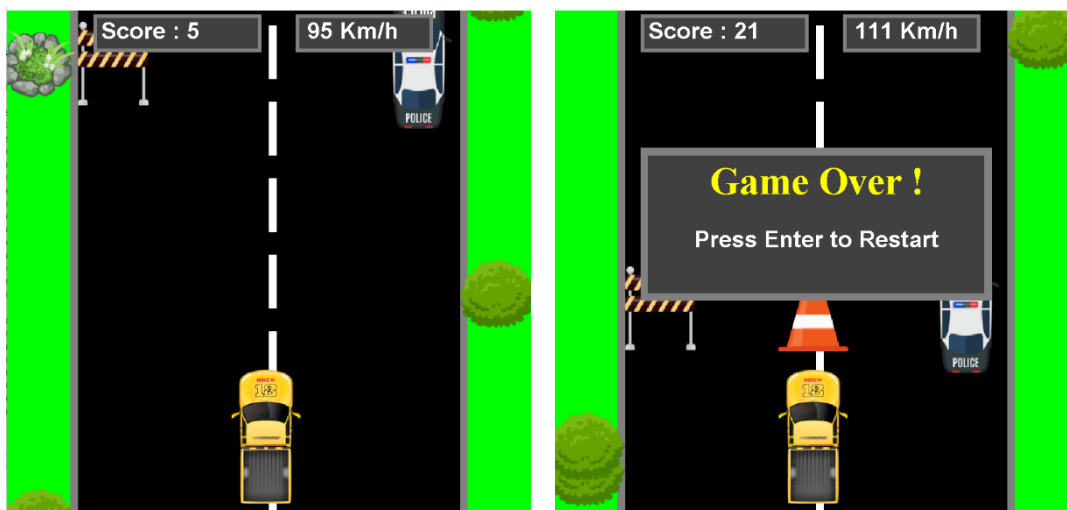
กรวย

ฉาก

- เริ่มเกม



- ถ้ารถไม่สามารถหลบรถตำรวจ ด้านตรงขางได้ จะแสดง Game Over



- ถ้าควบคุมรถได้ไม่ชนสิ่งกีดขวาง กิโลเมตรในการขับเพิ่มมากขึ้นรถจะขับเร็วขึ้น เพื่อให้หายากมากขึ้น



ตารางแผนการทำงาน

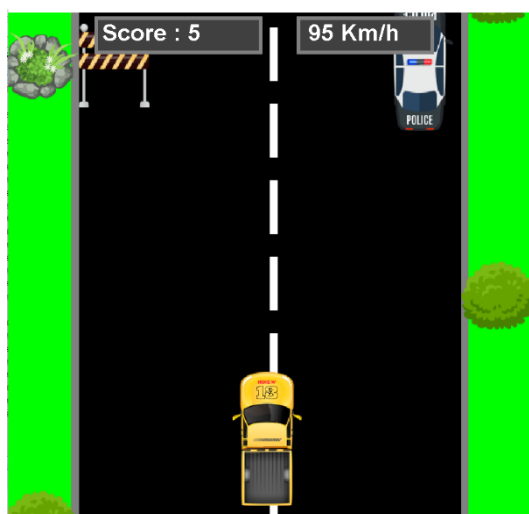
ลำดับ	รายการ	สัปดาห์ที่ 1	สัปดาห์ที่ 2	สัปดาห์ที่ 3	สัปดาห์ที่ 4
1	ทำกราฟฟิกภายในเกมและตัวละคร	←→			
2	ศึกษาเอกสารที่เกี่ยวข้อง	←→	→		
3	ลงมือเขียน โปรแกรม		←→	→	
4	ตรวจสอบและแก้ไขข้อผิดพลาด			←→	→
5	จัดทำเอกสาร				←→

บทที่ 2

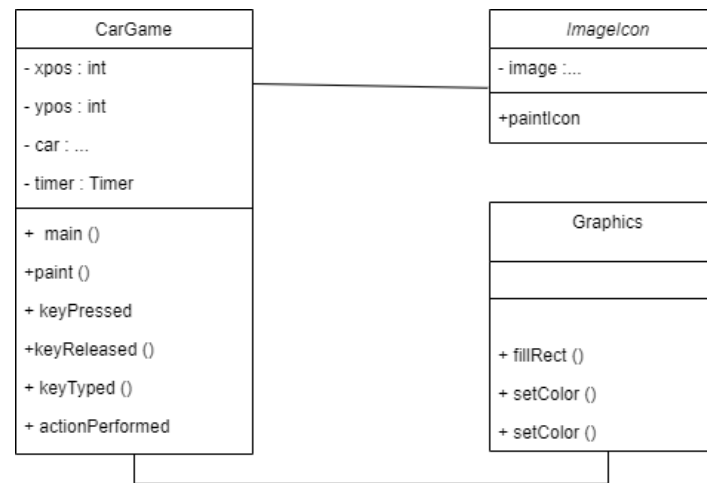
ส่วนการพัฒนา

วิธีการเล่น

ใช้ลูกศรบนแป้นพิมพ์เพื่อกำหนดให้รถขับ ซ้าย ขวา เมื่อชนสิ่งกีดขวางจะแสดง Game Over
รถจะขับเร็วขึ้นเรื่อย ๆ หากเรา ผ่านสิ่งกีดขวางได้นาน



คลาสไดอะแกรม



- คลาส CarGame เป็นคลาสหลักของเกม มีหน้าที่หลักในการเริ่มต้นเกมและจัดการองค์ประกอบต่างๆ ของเกม คลาสนี้มีคุณสมบัติ xpos และ ypos เพื่อระบุตำแหน่งของรถบนหน้าจอ และมีคุณสมบัติ car เพื่อเก็บข้อมูลเกี่ยวกับรถ
- คลาส ImageIcon เป็นคลาสที่ใช้แสดงภาพบนหน้าจอ คลาสนี้มีคุณสมบัติ image เพื่อเก็บข้อมูลเกี่ยวกับภาพ และมีคุณสมบัติ paintIcon() เพื่อแสดงภาพบนหน้าจอ
- คลาส Graphics เป็นคลาสที่ใช้วาดรูปบนหน้าจอ คลาสนี้มีคุณสมบัติ paint() เพื่อวาดรูปบนหน้าจอ และมีคุณสมบัติอื่นๆ ที่ใช้สำหรับวาดรูป เช่น fillRect() และ setColor()
- ความสัมพันธ์ระหว่างคลาส CarGame และ ImageIcon เป็นความสัมพันธ์แบบ aggregation ซึ่งหมายความว่าคลาส CarGame สามารถใช้คลาส ImageIcon ได้ แต่คลาส ImageIcon ยังสามารถใช้งานได้โดยไม่ต้องมีคลาส CarGame
- ความสัมพันธ์ระหว่างคลาส CarGame และ Graphics เป็นความสัมพันธ์แบบ dependency ซึ่งหมายความว่าคลาส CarGame ต้องการคลาส Graphics เพื่อทำงาน แต่คลาส Graphics ไม่จำเป็นต้องมีคลาส CarGame

รูปแบบการพัฒนา

เกมเป็นภาษา java ชนิด java application

Constructor

ตัวอย่าง

```
public CarGame(String title) {

    super(title);

    setBounds(300, 10, 700, 700);

    setVisible(true);

    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    setLayout(null);

    addKeyListener(this);

    setFocusable(true);

    setResizable(false);

    timer = new Timer(100, this);

    timer.start();

}
```

Encapsulation

ตัวอย่าง

```
public class CarGame extends JFrame implements KeyListener, ActionListener {

    private int xpos = 300;

    private int ypos = 700;
```

```

private ImageIcon car;

private Timer timer;

Random random = new Random();

}

```

Inheritance

ตัวอย่าง

```

public class CarGame extends JFrame implements KeyListener, ActionListener {

    private int xpos = 300;

    private int ypos = 700;

    private ImageIcon car;

    private Timer timer;

    Random random = new Random();

    private int num1 = 400, num2 = 0, num3 = 0;

    private int tree1ypos = 400, tree2ypos = -200, tree3ypos = -500, tree4ypos = 100,
    tree5ypos = -300, tree6ypos = 500;

    private int roadmove = 0;

    private int carxpos[] = { 100, 200, 300, 400, 500 };

    private int carypos[] = { -240, -480, -720, -960, -1200 };

    private int cxpos1 = 0, cxpos2 = 2, cxpos3 = 4;

    private int cypos1 = random.nextInt(5), cypos2 = random.nextInt(5), cypos3 =
    random.nextInt(5);

    int y1pos = carypos[cypos1], y2pos = carypos[cypos2], y3pos = carypos[cypos3];

```

```

private ImageIcon car1, car2, car3;

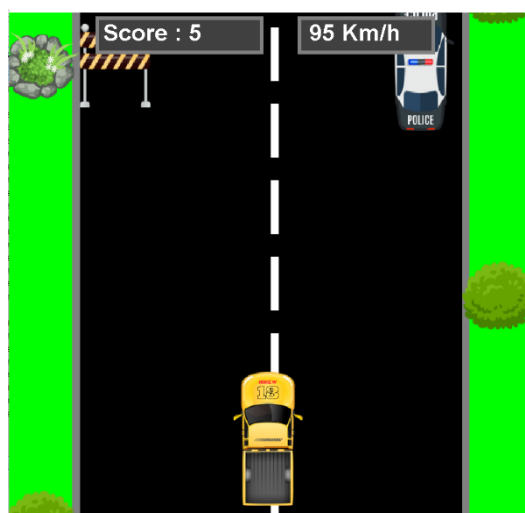
private int score = 0, delay = 100, speed = 0;

private ImageIcon tree1, tree2, tree3;

private boolean rightrotate = false, gameover = false, paint = false;
}

```

Event handling



หน้าเริ่มต้นจะทำการเริ่มเกม Event Handle ที่ปรากฏในเกมมีการเรียกใช้ ActionListener ในการควบคุมการเคลื่อนไหวรถผ่าน Timer และมีการเรียกใช้ KeyListener ผ่านการกดปุ่มโดยปุ่มที่กำหนด Event ไว้จะมี

- ลูกศรซ้าย = จะทำการกำหนด direction ให้กับรถเป็น “L”
- ลูกศรขวา = จะทำการกำหนด direction ให้กับรถเป็น “R”
- ปุ่ม Enter = กดเพื่อเริ่มเกมครั้งต่อไปหากแพ้

อัลกอริทึมที่สำคัญในโปรแกรม

- การจับบริดจ์โดยใช้ Keylistener เป็นตัวกำหนดปุ่มโดยจะใช้ลูกศร ซ้าย ขวา ในการควบคุมรถ
- การแสดงรถตำรวจที่ไล่ล่า ด้านตำรวจ กรวย มีการใช้ Keylistener แต่ User ไม่จำเป็นต้องควบคุมเอง แต่จะมีฟังก์ชันที่คอยกำหนดสิ่งกีดขวางแบบสุ่ม โดยใช้ Math.random

บทที่ 3

บทสรุป

ปัญหาที่พบระหว่างการพัฒนา

- การ paint ตำแหน่งถนนต้องนั่งคำนวณและกำหนดค่าเองค่อนข้างยาก
- เวลาไม่เพียงพอต่อ scope ของงาน

จุดเด่นของโปรแกรมที่ไม่เหมือนใคร

- มีการสร้างถนนที่กำหนดตำแหน่งโดยใช้ JPanel มาใช้งาน
- มีรถและกราฟฟิคที่สวยงาม

คำแนะนำสำหรับผู้สอนที่อยากให้อธิบาย หรือที่เรียนแล้วไม่เข้าใจ หรืออยากให้เพิ่มสำหรับน้อง ๆ รุ่นต่อไป

- อยากให้ลดความยากของตัวงานลงหรือมีการสอนการทำเกมมากขึ้นกว่าเดิมค่ะ