

Diamond Hunter



บทที่ 1 บทนำ

ที่มา

ฮันเตอร์สาว ได้ออกตามล่าเพชรในป่าใหญ่ โดยมีอุปสรรคคือ หมาป่า ผู้โหดร้าย
ฮันเตอร์ต้องหลบหลีก โดยห้ามเดินไปชนกับหมาป่าเด็ดขาด และต้องออกตามล่าเพชรเป็นจำนวน 10 เม็ด
ถึงจะชนะ แต่ทว่า เธอต้องแข่งกับเวลาภายใน 4 นาที อีกด้วย

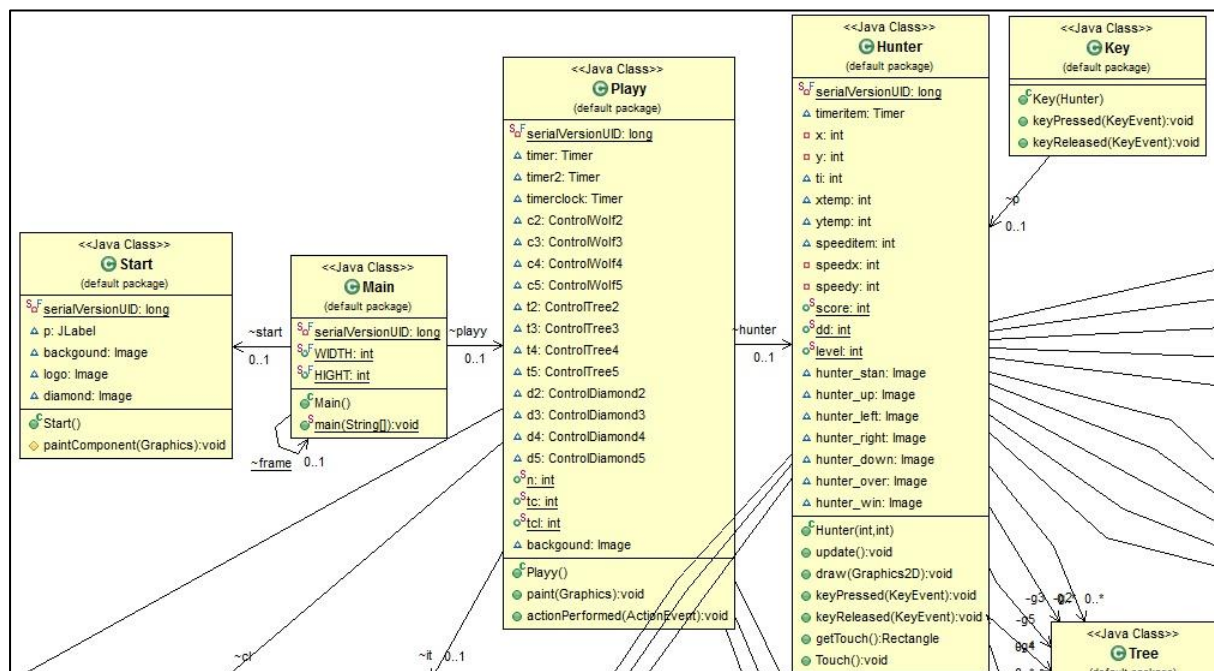
ประเภทของงาน เกมส์

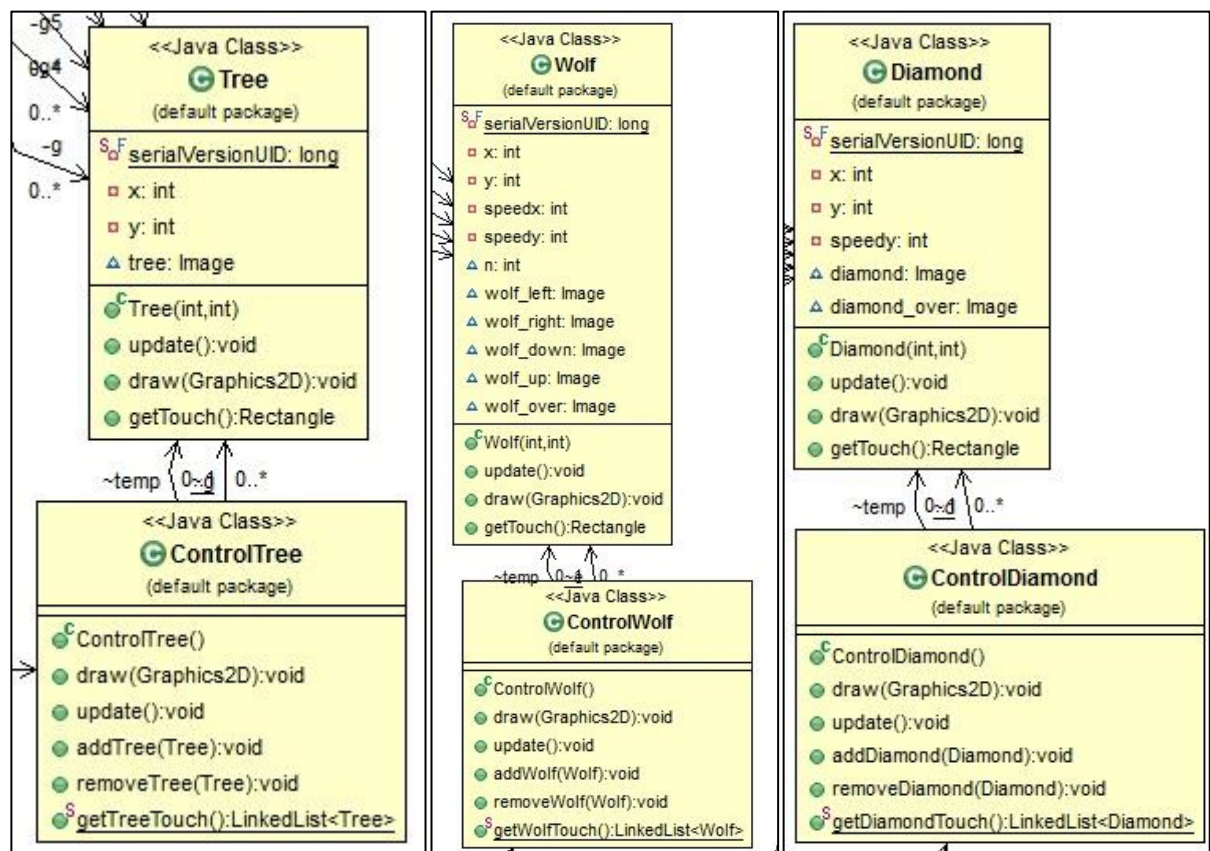
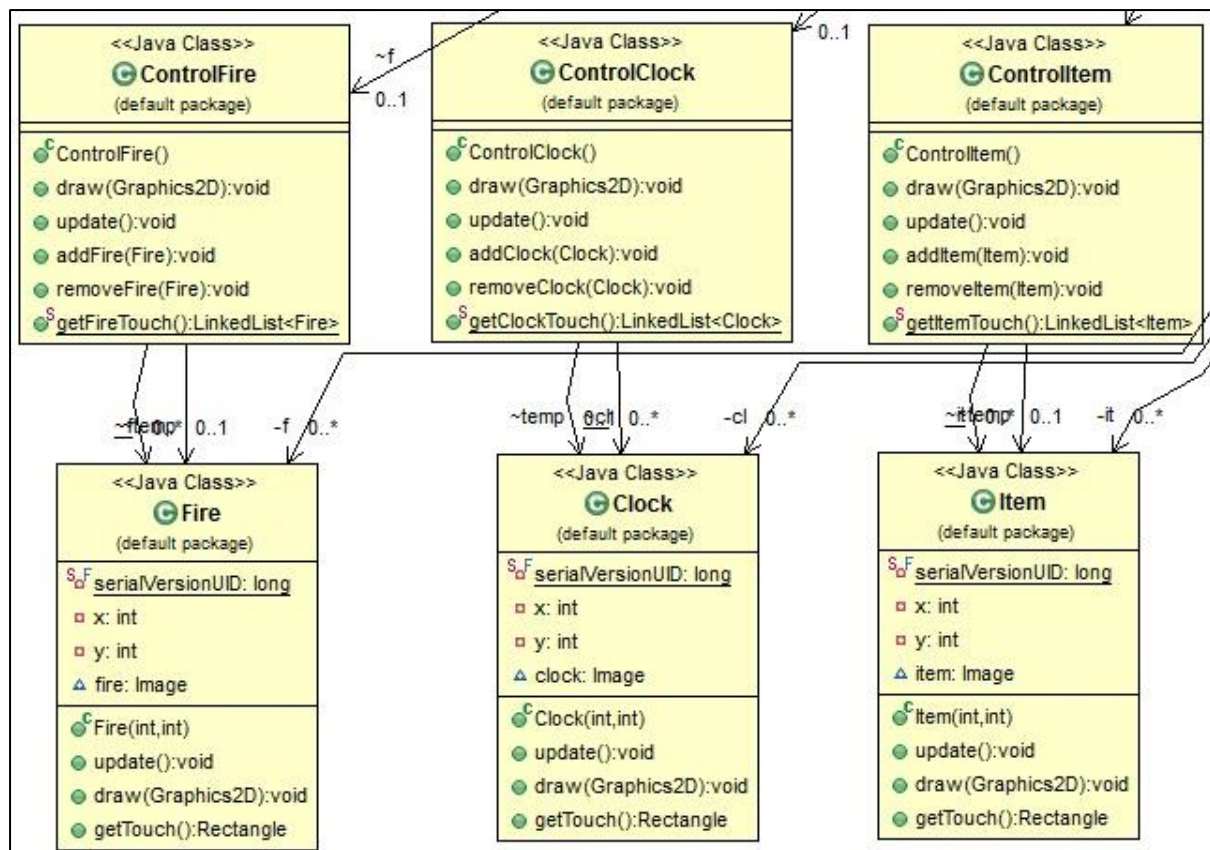
ประโยชน์

- 1.เป็นเกมส์ที่ฝึกทักษะการหลบหลีก ความว่องไว
- 2.เป็นเกมส์เบาสมอง ทำให้ผู้เล่นผ่อนคลายความเครียดได้

ขอบเขตของโครงการ

- 1.ตัวละครสามารถเก็บไอเทมเสริม และเพิ่มทักษะหรือเวลาได้
- 2.สามารถกำหนดขอบเขตตัวละครห้ามเดินชนต้นไม้ได้





อธิบายโปรแกรม

1. Class Main เป็นส่วนที่ตั้งค่าหน้าจอ และส่งต่อไปยัง Class Start
2. Class Start หน้าแรกของเกมส์ เป็น JLabel มีการเช็คปุ่ม MouseListener ปุ่มกดเพื่อเข้าเกมส์

```
p.setBounds(340,430,200,100);
add(p);
Main.frame.repaint();

p.addMouseListener(new MouseAdapter() {
    public void mouseClicked(MouseEvent e){
        Main.frame.remove(Main.frame.start);
        if(Main.frame.playy == null){
            Main.frame.play = new Playy();
        }
        Main.frame.add(Main.frame.playy);
        Main.frame.playy.setSize(Main.WIDTH,Main.HIGHT);
        Main.frame.playy.requestFocusInWindow();
        Main.frame.repaint();
    }
});
```

เซตตำแหน่งและขนาดของปุ่ม Start
เข้าเช็คว่ามีเมาส์กดปุ่มหรือยัง
หากกดแล้วจะเช็คต่อว่าหน้า Playy
ว่างหรือไม่ หากว่างจะสร้างหน้าใหม่
พร้อมเซตขนาดหน้าจอ แล้วส่งต่อไปยัง
Playy

3. Class Playy เป็น JPanel และ implement กับ ActionListener เพื่อใช้ Timer ในการนับเวลา และการเดินของตัวละคร มี Method paint สำหรับการวาด แบ็กกราวนด์ และตัวละครต่างๆ และยังมี Method actionPerformed สำหรับส่งไปเช็ค ว่าตัวละครมีการขยับ อัปเดตอะไรบ้าง
4. Class Hunter เป็นส่วนเซตค่าตัวละครหลัก ก็คือฮันเตอร์ มี Method keyPressed เช็ค ว่าผู้เล่นกดแป้นพิมพ์อะไร และ keyReleased หากผู้เล่นไม่ได้กดแป้นก็จะไปเซตค่าให้ตัวละครหยุดเดิน Method Touch สำหรับเช็ค ว่า ขณะเล่น ตัวละครชนอะไรบ้าง


```

public void Touch() {
    if(Hunter.Level == 1) {
        for(int i = 0; i < e.size(); i++) {
            if(getTouch().intersects(e.get(i).getTouch())) {
                Hunter.dd = 1;
            }
        }
        for(int i = 0; i < g.size(); i++) {
            if(getTouch().intersects(g.get(i).getTouch())) {
                speedx = 0;
                speedy = 0;

                if (Playy.n == 1) {
                    x--;
                }
                if (Playy.n == 2) {
                    x++;
                }
                if (Playy.n == 3) {
                    y--;
                }
                if (Playy.n == 4) {
                    y++;
                }
            }
        }
        for(int i = 0; i < d.size(); i++) {
            if(getTouch().intersects(d.get(i).getTouch())) {
                d.remove(i);
                score++;
            }
        }
    }
}

```

Method Touch

- เข้าเช็คค่า level ว่าอยู่ในหน้าที่ level เท่าไหร่แล้ว
- เช็คการชน ระหว่าง ฮันเตอร์กับ ตัวละครอีกตัว ที่ใส่ค่าไว้ใน Linkedlist (ในลูปนี้หมายถึงตัวหมาป่า)
- เมื่อชนกัน จะเซตค่า dd = 1 (dd คือตัวแปรเช็คการจบเกมส์)
- เช็คการชน ระหว่าง ฮันเตอร์กับ ต้นไม้ หากชนจะบวกลบค่า x y ให้ฮันเตอร์ถอยห่างจากต้นไม้มา 1 เพื่อเป็นการเซตไม่ให้ฮันเตอร์เดินเข้าต้นไม้
- เช็คการชน ระหว่าง ฮันเตอร์กับ เพชร เมื่อชนจะลบค่าเพชรออก และเพิ่มสกอร์

5. Class Key รับค่าการกดเป็นพิกซ์ และส่งกลับไปยัง Hunter

6. Class Wolf , Tree , Diamond , Fire , Clock , Item เป็นส่วนเซตค่าตัวละคร

(หมาป่า, ต้นไม้, เพชร, ไฟ, นาฬิกาการเพิ่มเวลา) มี Method draw ที่รับค่าจาก Playy มาเพื่อวาด และมี Method getTouch สำหรับตั้งค่า พื้นที่ ที่จะให้โดนชน แล้วส่งค่ากลับไป

```

public Rectangle getTouch() {
    return new Rectangle(x, y, 40, 40);
}

```

7. Class ControlWolf , ControlTree , ControlDiamond , ControlFire , ControlClock , ControlItem สร้าง Linkedlist เก็บค่า x y ตำแหน่งการเกิดของหมาป่า

```
public class ControlWolf {
    static LinkedList<Wolf> e = new LinkedList<>();
    Wolf temp;

    public ControlWolf() {
        addWolf(new Wolf(53,143));
        addWolf(new Wolf(443,413));
    }

    public void draw(Graphics2D g2d) {
        for(int i = 0; i < e.size(); i++) {
            temp = e.get(i);
            temp.draw(g2d);
        }
    }

    public void update() {
        for(int i = 0; i < e.size(); i++) {
            temp = e.get(i);
            temp.update();
        }
    }

    public void addWolf(Wolf wolf) {
        e.add(wolf);
    }

    public void removeWolf(Wolf wolf) {
        e.remove(wolf);
    }
}
```

Method ControlWolf

- ประกาศ LinkedList

- ส่งค่าเข้า Method addWolf

ระบุตำแหน่งการเกิดของหมาป่า

- Method draw สำหรับการวาดหมาป่า

จาก LinkedList โดยวนลูป for

- Method update สำหรับเช็คการชน
และการเคลื่อนที่ของหมาป่า

8. Class ControlClock

```
public void update() {
    for(int i = 0; i < cl.size(); i++) {
        if (Playy.tcl <= 0){
            cl.remove(temp);
        }else{
            temp = cl.get(i);
        }
        temp.update();
    }
}
```

Method update

วนลูปเช็คค่าตามจำนวนตัวละครใน

LinkedList เช็ค ตัวแปร tcl

ตัวแปรเวลาว่หมดหรือยัง หากหมด

ตัวไอเทมเวลาจะหายไป

พบได้ในด่านที่ 3

บทที่ 3 บทสรุป

ปัญหาที่พบระหว่างการพัฒนา

การเดินชนต้นไม้ แล้วตั้งค่าไม่ให้ชน

จุดเด่นของโปรแกรม

เกมฝึกความว่องไว การใช้ทักษะ แข่งกับเวลา สนุกเฮฮาแน่นอน

คำแนะนำผู้สอน

อาจารย์สอนดี เข้าใจง่ายครับ แต่บางเรื่องสอนไวไปครับ กำลังพิมพ์เงยหน้ามา
อาจารย์พิมพ์เสร็จพร้อมรันแล้ว

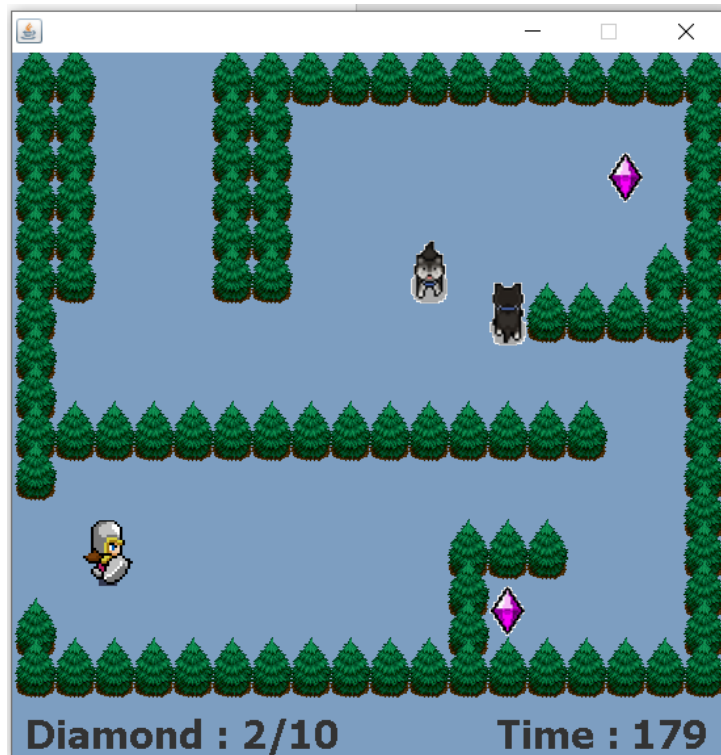
อธิบายหน้าจอแสดงผล



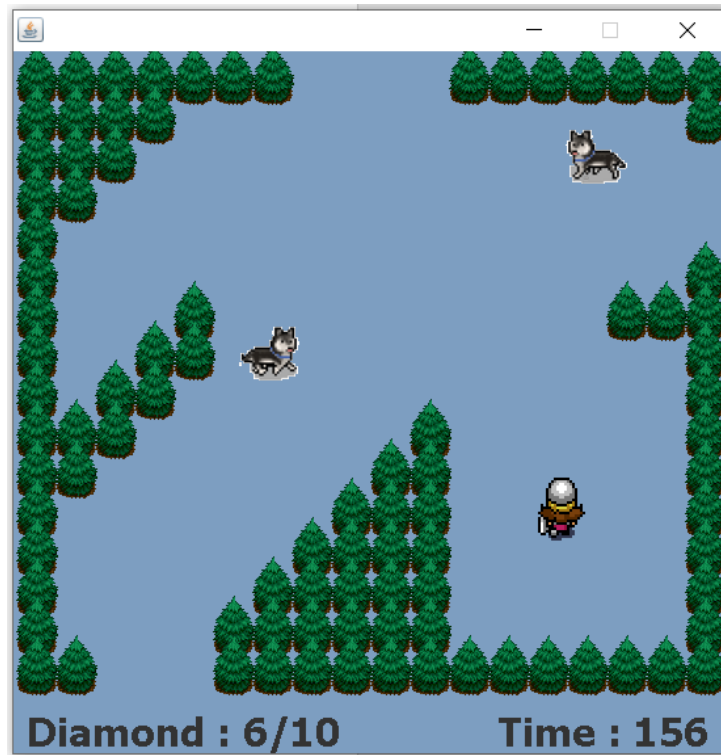
หน้าแรก



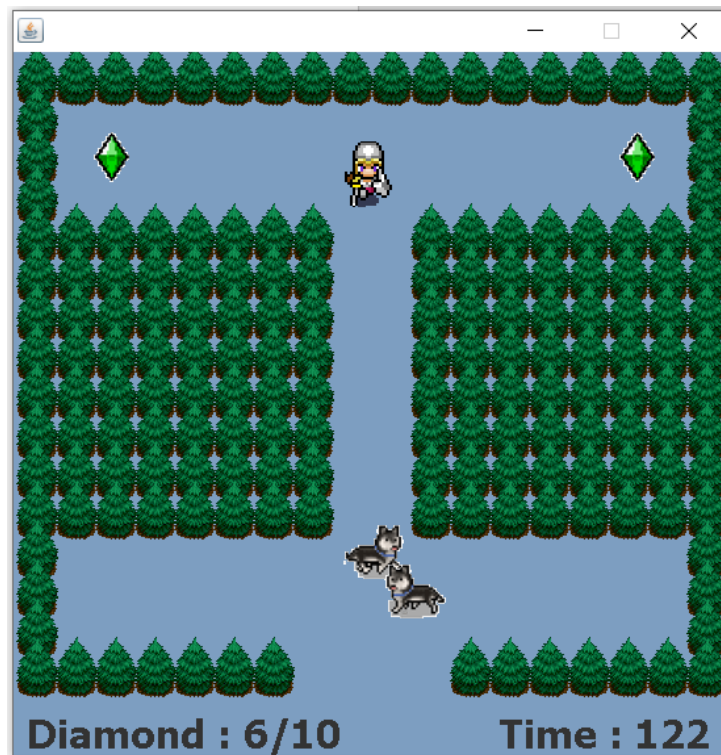
Level 1



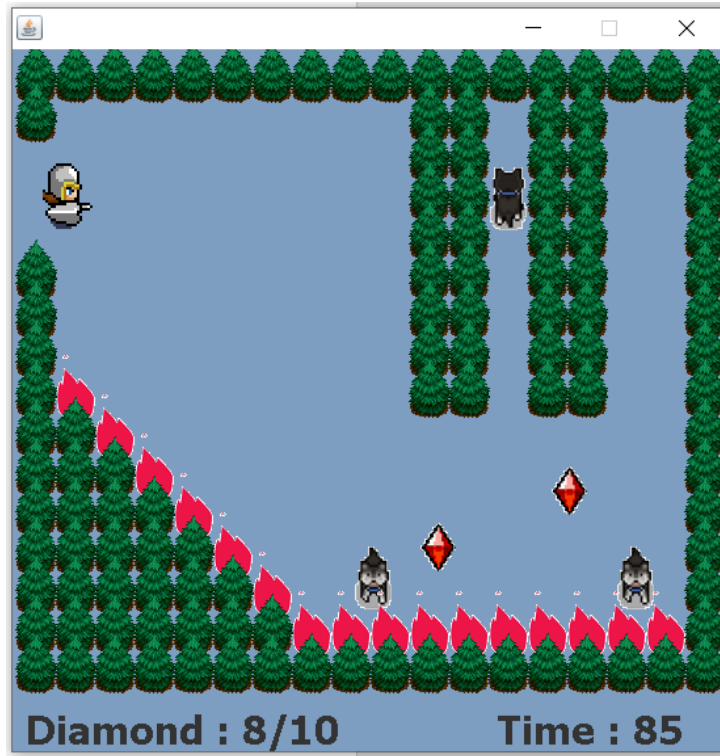
Level 2



Level 3



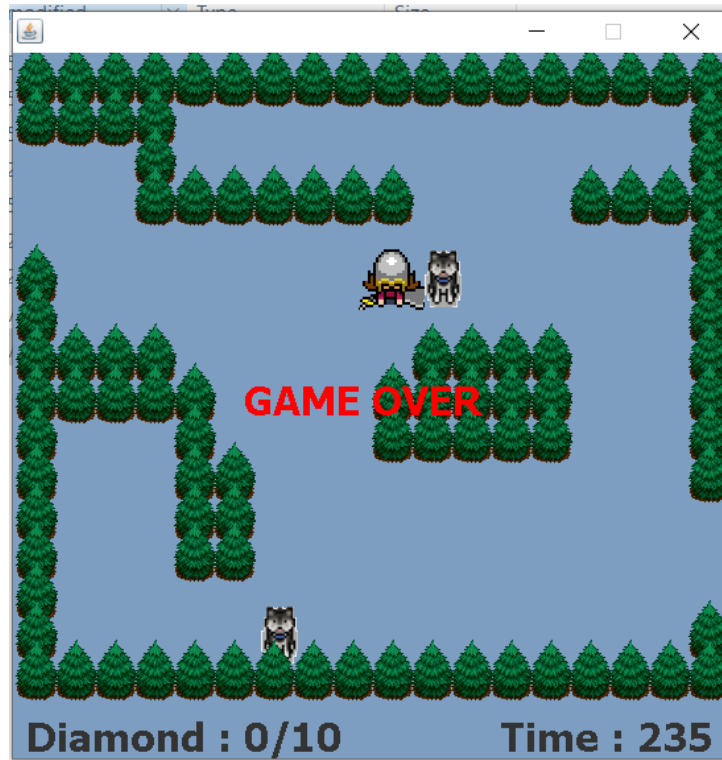
Level 4



Level 5



Win



Game Over