

Squidding game

(เกมสึดกหมึก)

จัดทำโดย

ทักษพร ญาณสังวร

รหัสนักศึกษา 6404062630376

เสนอ

ผู้ช่วยศาสตราจารย์ สถิต ประสมพันธ์

รายวิชา

การเขียนโปรแกรมเชิงวัตถุ(Object-Oriented Programming)

040613204 ภาคการศึกษาที่ 1/2565

ภาควิชาวิทยาการคอมพิวเตอร์และสารสนเทศ

คณะวิทยาศาสตร์ประยุกต์

Squidding Game(เกมส์ตกหมึก)

บทที่ 1 บทนำ

- ที่มาและความสำคัญของโปรเจค

รายงานนี้เป็นส่วนหนึ่งในรายวิชา Object Oriented Programming (การเขียนโปรแกรมเชิงวัตถุ) เป็นการนำองค์ความรู้ตลอดภาคการศึกษามาใช้ในการสร้างเกมส์ และช่วยต่อยอดความสามารถในการเขียนโปรแกรมเชิงวัตถุและการเชื่อมต่อประสานกับผู้ใช้ (GUI) มีความสำคัญต่อการศึกษากการเขียนโปรแกรมอย่างยิ่ง ที่จะช่วยให้การเขียนโปรแกรมเชิงวัตถุสามารถนำมาใช้ป็นสื่อต่างๆได้ในยุคปัจจุบัน

- ประเภทของโครงการ

เกมส์เพื่อความบันเทิง และช่วยในการฝึกสมาธิโดยมีเวลาเป็นตัวควบคุม

- ประโยชน์

- 1) ฝึกความอดทน
- 2) ฝึกสมาธิ การควบคุม
- 3) ช่วยให้สนุก คลายเครียด
- 4) ฝึกการทำงานของมือและตาให้ประสานกัน

- ขอบเขตของโครงการ(แนบ Proposal)

วันที่ 1-8 ตุลาคม จัดทำรูปตัวละคร และภาพพื้นหลัง , จัดหาข้อมูลให้เหมาะสมกับเกมส์

วันที่ 9-22 พฤศจิกายน เขียนโปรแกรม , ทดลองเล่นจริง

วันที่ 20-25 พฤศจิกายน ตรวจสอบข้อผิดพลาด , จัดทำเอกสาร

บทที่ 2 ส่วนการพัฒนา

● เนื้อเรื่องย่อหรือวิธีการเล่น

ชาวประมงต้องการตกปลาหมึกไปขาย โดยต้องตกหมึกให้ได้ภายในเวลาที่กำหนด ให้ได้จำนวนหมึกมากที่สุดเพื่อนำไปขาย แต่ในบรรดาฝูงหมึกจะมีหมึกจำนวนหนึ่งที่โกรธการโดนตกอยู่ ต้องช่วยชาวประมงไม่ให้ตกได้หมึกที่โกรธ ถ้าตกหมึกได้ตัวที่โกรธจะทำให้ชาวประมงถูกนำหมึกสีดำฉีดยาใส่ทำให้พลังชีวิตของชาวประมงลดลง

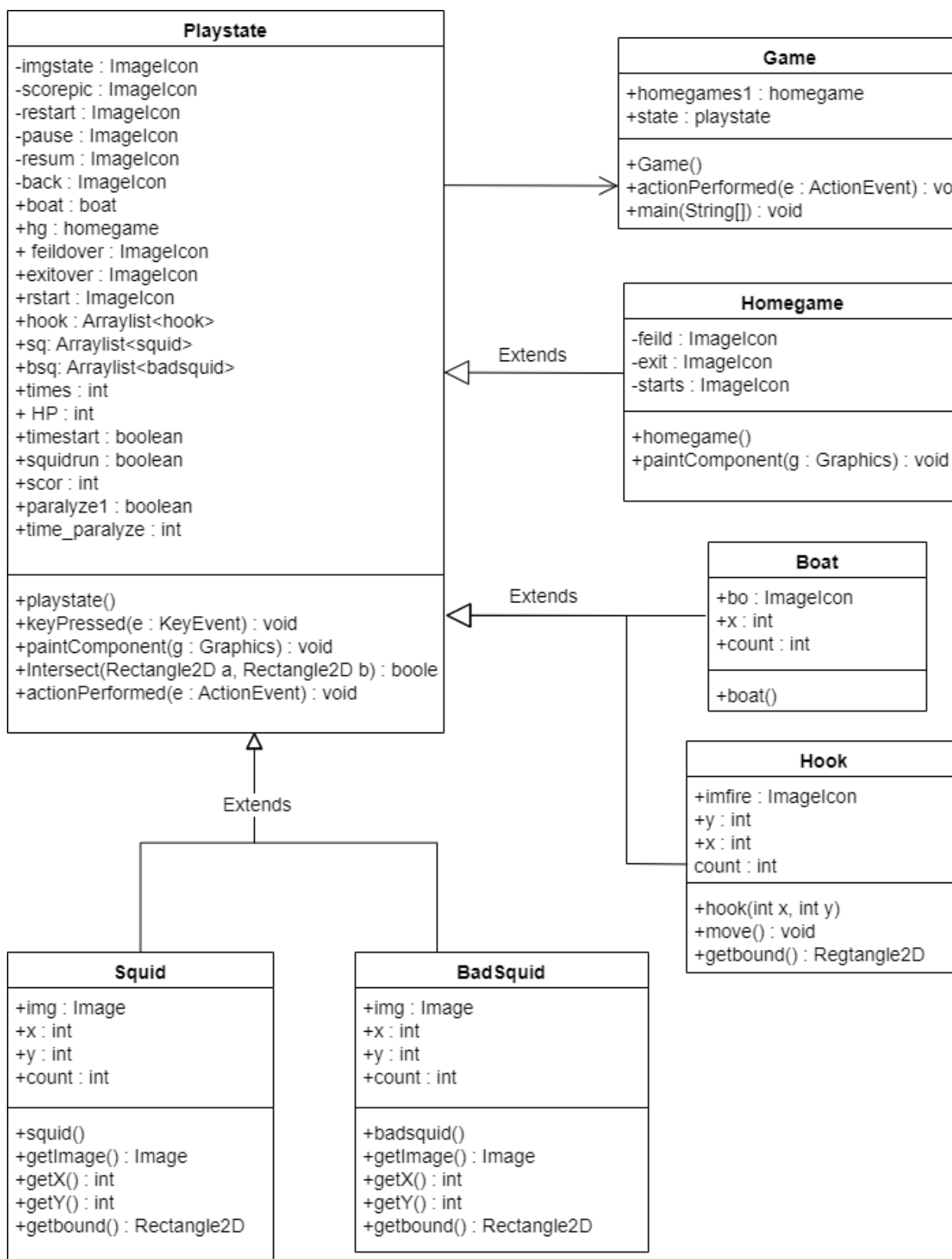
ใช้คีย์บอร์ดบังคับทิศทาง ลงช่วยบังคับเบ็ดตกหมึก และคีย์บอร์ดบังคับทิศทางซ้าย-ขวาของเรือ ต้องบังคับทิศทางให้ตรงกับหมึกที่ต้องการจริงๆ

● คลาสไดอะแกรม พร้อมคำอธิบาย

แบ่งเป็น 7 คลาส ได้แก่

1. **Game** extends JFrame implements ActionListener
 คลาส 1 จะเป็นหน้าหลักที่จะรันตัวเกมส์ออกมา
2. **Playstate** extends JPanel implements ActionListener
 คลาส 2 ตัวควบคุมด้าน ที่กำหนดหน้าต่างภายในเกมส์ เมื่อกดเริ่มต้น
3. **Homegame** extends JPanel
 คลาส 3 เป็นหน้าแรกที่เป็น จะมีปุ่มเริ่ม และปุ่มออกเกมส์
4. **Boat** คลาส 4 มีการใส่รูปเรือในเกมส์
5. **Hook** extends JPanel
 คลาส 5 เป็นการกำหนดเบ็ดตกหมึก ขนาดและทิศทางของเบ็ดตกหมึก
6. **Squid**
7. **Badsquid**
 คลาส 6 และ 7 เป็นการใส่รูปหมึกและกำหนดทิศทางการวิ่ง และความเร็วของหมึก

โดยในแต่ละคลาสจะมี คอนสตรัคเตอร์ และ เมธอด ดังรูป



- รูปแบบการพัฒนา Application / Applet

Java By Apache NetBeans IDE 12.6

Java.awt(Abstract Window Toolkits) และ javax.swing

- อธิบายส่วนของโปรแกรมที่มี

1) Constructor

การประกาศคอนสตรัคเตอร์ให้คลาส hook จะมี Default Constructors ด้านบน

และ Constructor ที่รับค่า parameters (int x, int y)

```
public class hook extends JPanel{
    public ImageIcon imfire = new ImageIcon();
    public int y;
    public int x;
    public int count=0;

    public hook() {
    }

    public hook(int x,int y){
        imfire = new ImageIcon(this.getClass().getResource("image/hook.png"));
        this.x=x;
        this.y=y;
    }

    public void move() {
        this.y+=1;
    }

    public Rectangle2D getbound(){
        return (new Rectangle2D.Double(x,y,30,30));
    }
}
```

2) Encapsulation

เป็นการแบ่งลำดับการมองเห็นของคลาส homegame ที่มีการประกาศตัวแปร private final ทำให้ใช้ได้ภายใน และไม่ต้องการให้คลาสสืบทอดได้

```

public class homename extends JPanel{
    private final ImageIcon feild = new ImageIcon(this.getClass().getResource("image/bg.jpg"));
    private final ImageIcon exit = new ImageIcon(this.getClass().getResource("image/exit.jpg"));
    private final ImageIcon starts = new ImageIcon(this.getClass().getResource("image/start.jpg"));
    public JButton BStart = new JButton(starts);
    public JButton BExit1 = new JButton(exit);
    homename() {
        setLayout(null);
        BExit1.setBounds(500, 500, 300,100);
        add(BExit1);
        add(BStart);
        BStart.setBounds(500,350,300,100);
        add(BStart);
    }
    public void paintComponent(Graphics g){
        super.paintComponent(g);
        g.drawImage(feild.getImage(),0,0,1000,800,this);
    }
}

```

3) Composition

เป็นการประกอบกันจากหลายคลาสอย่างคลาส playstate

ที่มีการรวมกันของ คลาส boat และ คลาส homename

```

public class playstate extends JPanel implements ActionListener{
    private final ImageIcon imgstate = new ImageIcon(this.getClass().getResource("image/bgstate.jpg"));
    private final ImageIcon scorepic = new ImageIcon(this.getClass().getResource("image/score.jpg"));
    private ImageIcon restart = new ImageIcon(this.getClass().getResource("image/restart.jpg"));
    private ImageIcon pause = new ImageIcon(this.getClass().getResource("image/pause.jpg"));
    private ImageIcon resum = new ImageIcon(this.getClass().getResource("image/resume.jpg"));
    private ImageIcon back = new ImageIcon(this.getClass().getResource("image/close.jpg"));
    boat boat = new boat();
    homename hg = new homename();
    ImageIcon feildover = new ImageIcon(this.getClass().getResource("image/bgover.jpg"));
    ImageIcon exitover = new ImageIcon(this.getClass().getResource("image/exit.jpg"));
    ImageIcon rstart = new ImageIcon(this.getClass().getResource("image/start.jpg"));
    JButton BStartover = new JButton(rstart);
    JButton BExitover = new JButton(exitover);
    private JLabel score = new JLabel();
    public JButton BPause = new JButton(pause);
    public JButton BExithome = new JButton(back);
    public JButton Bresum = new JButton(resum);
    public JButton Brestart = new JButton(restart);
    public ArrayList<hook> hook = new ArrayList<hook>();
    public ArrayList<squid> sq = new ArrayList<squid>();
    public ArrayList<badsquid> bsq = new ArrayList<badsquid>();
    public int times ;
    public int HP = 3;
    boolean timestart = true;
    boolean squidrun = false;
    public int scor = 0;
    boolean paralyze = false;
    int time_paralyze=5;
}

```

4) Polymorphism

เป็นการเขียนชุดเมธอดเหมือนกันแต่มีพารามิเตอร์ต่างกัน อย่างคลาส hook และคลาส

homegame ก็ต่างมีการ extends JPanel เหมือนกัน

```
public class hook extends JPanel{
    public ImageIcon imfire = new ImageIcon();
    public int y;
    public int x;
    public int count=0;

public class homegame extends JPanel{
    private final ImageIcon feild = new ImageIcon(this.getClass().getResource("image/bg.jpg"));
    private final ImageIcon exit = new ImageIcon(this.getClass().getResource("image/exit.jpg"));
    private final ImageIcon starts = new ImageIcon(this.getClass().getResource("image/start.jpg"));
    public JButton BStart = new JButton(starts);
    public JButton BExit1 = new JButton(exit);
```

5) Abstract

เป็นการสืบทอดเมธอดที่ยังไม่สมบูรณ์ มีการใส่เมธอดให้ keyPressed ที่มาจากคลาส

ActionListener

```
this.addKeyListener(new KeyAdapter() {
    public void keyPressed(KeyEvent e) {
        int a = e.getKeyCode();
        if (!paralyze) {
            if (a == KeyEvent.VK_LEFT) {
                boat.x -= 10;
                boat.count++;
            }
            else if (a == KeyEvent.VK_RIGHT) {
                boat.x += 10;
                boat.count++;
            }

            if (boat.count > 3) {
                boat.count = 0;
            }
            else if (a == KeyEvent.VK_DOWN) {
                boat.count = 4;
                hook.add(new hook(boat.x + 30, 350));
            }
        }
    }
    public void keyReleased(KeyEvent e) {
        boat.count = 0;
    }
});
```

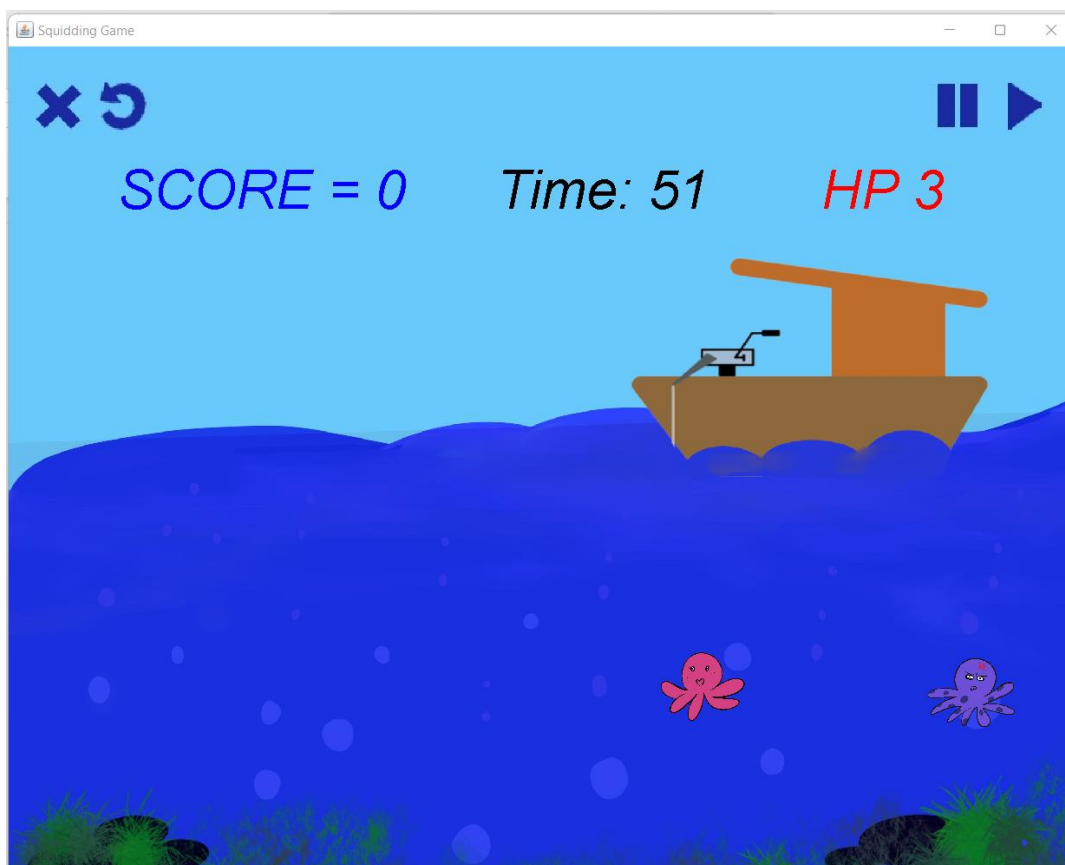
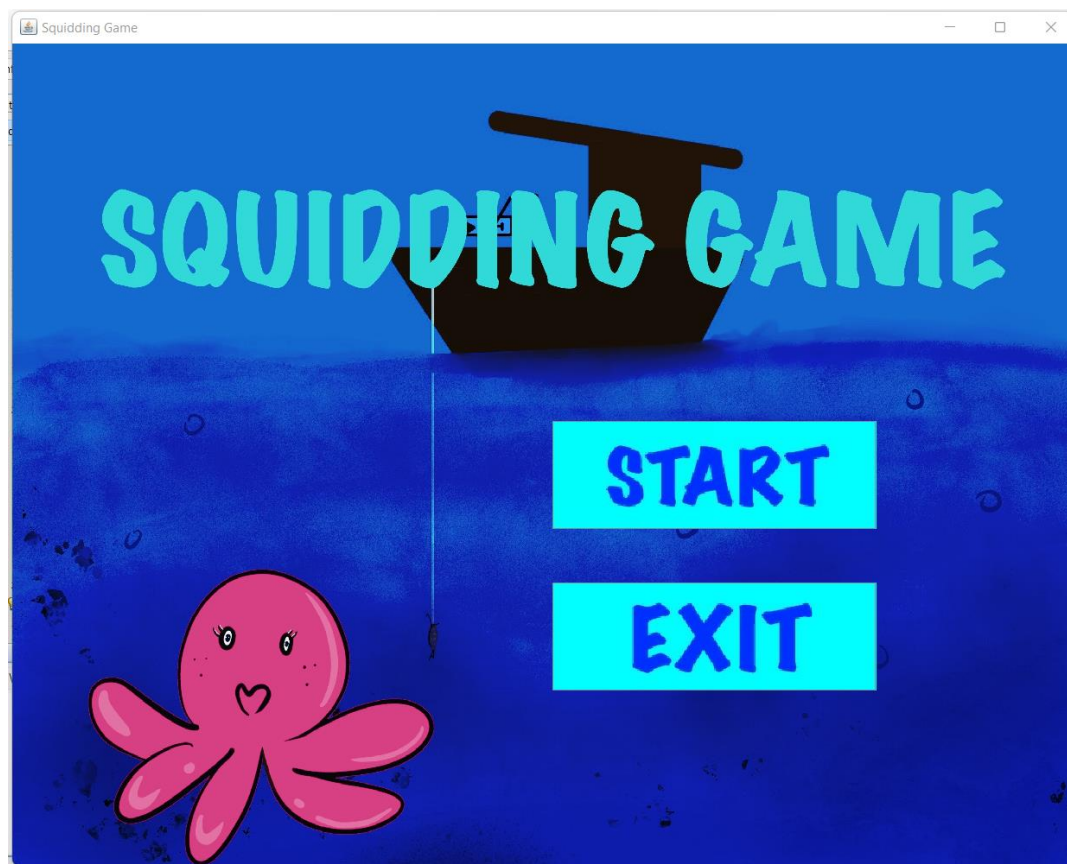
6) Inheritance

ในคลาส playstate มีการสืบทอดคุณสมบัติจาก paintComponent ซึ่งเป็น Superclass

```
public void paintComponent(Graphics g){
    super.paintComponent(g);
    g.drawImage(imgstate.getImage(),0,0,1000,800,this);
    if(times <= 0 || HP<=0){
        this.remove(BPause);
        this.remove(Bresum);
        this.setLayout(null);

        if(scor>0){
            g.drawImage(scorepic.getImage(),0,0,1000,800,this);
            g.setColor(Color.BLACK);
            g.setFont(new Font("Hobo Std",Font.HANGING_BASELINE,100));
            g.drawString(""+scor,430,500);
        }
        else{
            g.drawImage(feildover.getImage(),0,0,1000,800,this);
        }
    }
}
```

- หน้าจอ GUI อธิบายส่วนประกอบของ GUI โครงสร้างของ GUI ประกอบด้วย Component อะไรบ้าง
 - หน้า GUI ของ homepage ประกอบด้วย JPanel, JButton, ImageIcon, PaintComponent
 - หน้า GUI ของ playstate ประกอบด้วย JPanel, JButton, ImageIcon, PaintComponent, JLabel และการใช้ Graphic ได้แก่
 - setFont, setColor, DrawString
 - หน้า GUI ของ Game ประกอบด้วย JFrame มีการ setSize, setTitle, setDefaultCloseOperation, setVisible(true), setLocationRelativeTo



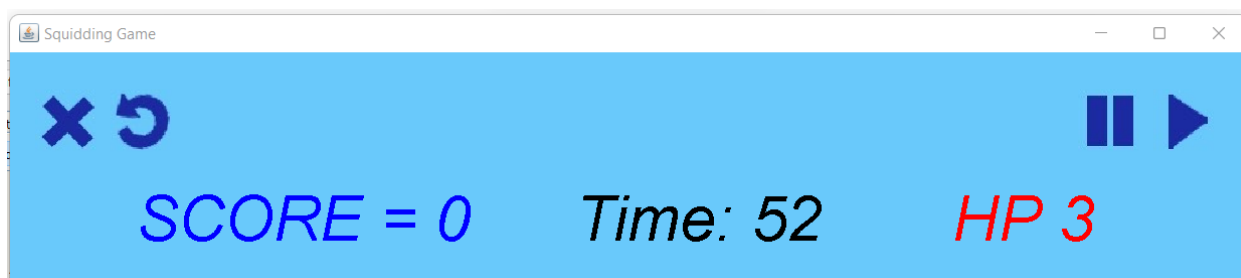
- อธิบาย Event handling ที่มีในหน้าจอ

เมื่อกดปุ่ม Start เกมจะเริ่มขึ้น และถ้ากดปุ่ม Exit หน้าต่างจะปิดลง



หน้าจอเกมเมื่อกดปุ่ม Start แล้ว จะมีปุ่ม 4 ปุ่มเรียงจากซ้ายไปขวา

1. ปุ่มปิด จะเป็นการปิดเกมส์แล้วไปที่หน้าจอหลัก
2. ปุ่มรีเกมส์ จะเป็นการรีเซตหน้าเกมส์ใหม่ทุกอย่าง
3. ปุ่มหยุดชั่วคราว จะหยุดเวลาและตัวหมึกที่วิ่ง
4. ปุ่มเล่นต่อ จะเริ่มนับเวลาต่อและหมึกจะเริ่มวิ่งต่อจากเดิม



- อธิบายอัลกอริทึมที่สำคัญในโปรแกรม

เป็นการกำหนดว่าเกมส์จะสิ้นสุดอย่างไร และต้องแสดงผลอะไรต่อไป โดยมีการใช้ if-else

```
public void paintComponent(Graphics g) {
    super.paintComponent(g);
    g.drawImage(imgstate.getImage(), 0, 0, 1000, 800, this);
    if(times <= 0 || HP<=0) {
        this.remove(BPause);
        this.remove(Bresum);
        this.setLayout(null);

        if(scor>0) {
            g.drawImage(scorepic.getImage(), 0, 0, 1000, 800, this);
            g.setColor(Color.BLACK);
            g.setFont(new Font("Hobo Std", Font.HANGING_BASELINE, 100));
            g.drawString(""+scor, 430, 500);
        }
        else{
            g.drawImage(feildover.getImage(), 0, 0, 1000, 800, this);
        }

    }else if(times <= 0 && HP<=0) {
        this.remove(BPause);
        this.remove(Bresum);
        this.setLayout(null);
        g.drawImage(feildover.getImage(), 0, 0, 1000, 800, this);
    }
```

การตรวจสอบการชนกัน ของสี่เหลี่ยม

```
,
public boolean Intersect(Rectangle2D a, Rectangle2D b){
    return (a.intersects(b));
}
```

บทที่ 3 สรุป

- ปัญหาที่พบระหว่างการพัฒนา

1. นามสกุลของไฟล์รูปและชื่อของรูป ต้องตรวจสอบให้ได้ว่าใส่ถูกต้องหรือไม่
2. ขนาดของรูปที่ไม่พอดี ทำให้ในเกมสับสนภาพแตก
3. ไม่สามารถแสดงปุ่มได้เมื่อเริ่มเกมส์ใหม่ ปุ่มจะหายไปเลย

- จุดเด่นของโปรแกรมที่ไม่เหมือนใคร

วิธีการตกหมึกที่โยนเบ็ดดทิ้งลงไปเลย ไม่มีการดึงหมึกขึ้นมา และรูปร่างหน้าตาของหมึกรวมถึงวิธีและทิศทางที่หมึกวิ่งมีความเร็ว

- คำแนะนำสำหรับผู้สอนที่อยากให้อธิบาย หรือที่เรียนแล้วไม่เข้าใจ หรืออยากให้เพิ่มสำหรับน้อง ๆ รุ่นต่อไป

ความแตกต่างของ Concept OOP มีการใช้งานที่ต่างกันอย่างไร สามารถอธิบายรวมกับ GUI ได้อย่างไร