



เกม Flappy Bird

นายคัชตบดี เอี่ยมสอาด 6304062630032 Sec 1

เสนอ

อาจารย์สถิตย์ ประสมพันธ์

รายงานฉบับนี้เป็นส่วนหนึ่งของวิชา Object Oriented Programming  
สาขาวิทยาการคอมพิวเตอร์และสารสนเทศ คณะวิทยาศาสตร์ประยุกต์  
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

ปีการศึกษา 2565

# บทที่ 1

## บทนำ

### ที่มาและความสำคัญของโปรเจก

เริ่มแรกเดิมทีผู้พัฒนามีความต้องการที่จะสร้างเกมอยู่ก่อนหน้านี้แล้วและมีความฝันที่จะสร้างเกม และเกมนี้ยังเป็นส่วนหนึ่งของ Project วิชา OOP และการทำงานของตัวโปรแกรมนั้นมีการทำงานอย่างไรบ้าง เพราะเมื่อก่อนผู้พัฒนาก็สงสัยกระบวนการการทำเกมเหมือนกันจึงเลือกเรียนในสาขานี้และอยากจะลองสร้างเกมของตัวเองดูบ้าง เพื่อที่จะสามารถทำไปต่อยอดได้ในอนาคตซึ่งภาษาที่ผู้พัฒนาใช้สร้างก็คือ ภาษา JAVA ซึ่งก็ได้มีการค้นคว้าหาข้อมูลมาจากในอินเทอร์เน็ต และเพื่อเป็นตัวอย่างแก่น้อง ๆ รุ่นต่อ ๆ ไปด้วย เกมนี้ถูกสร้างมาเพื่อต้องการให้ผู้เล่นเกิดความสนุกสนาน ผ่อนคลาย และยังเป็นการฝึกสมาธิและความอดทนไปในตัวด้วย ซึ่งโปรแกรมที่ใช้ในการพัฒนา คือ Netbean

### ประเภทของโครงการ

โครงการนี้เป็นประเภทเกม ที่พัฒนามาจากภาษา JAVA โดยการเขียนโค้ด

### ประโยชน์

- ฝึกความอดทน
- ฝึกสมาธิ
- เกิดความสนุกสนาน

### ขอบเขตของโครงการ

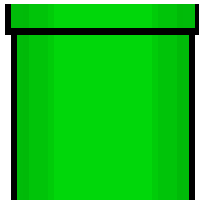
-ตัวเกม Flappy Bird

-Story board

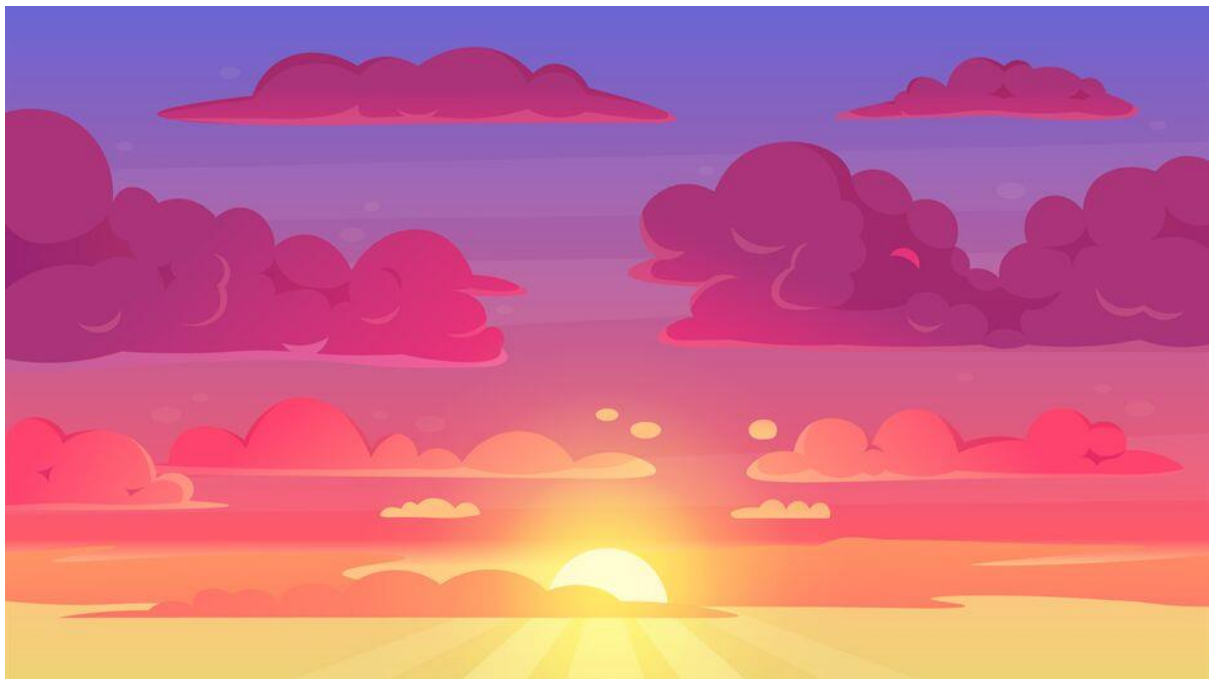
-ตัวละครในเกม



-สิ่งกีดขวาง



-ฉาก



1.ปัญหาที่พบเจอ

-ฉากไม่โหลด

## 2.วิธีแก้ปัญหา

-ทำการตรวจเช็คโค้ดและพบว่าการใส่ชื่อ Path ที่เก็บไฟล์ผิด

### ตารางการทำงานเดือนตุลาคม

ลำดับ	รายการ	14-20	21-28	29-30
1	หารูปตัวละครและฉาก			
2	ศึกษาข้อมูลเกี่ยวกับภาษา JAVA			
3	เขียนโปรแกรม			
4	ทำเอกสาร			
5	ตรวจสอบ BUG			

## บทที่ 2

### ส่วนของการพัฒนา

#### เรื่องย่อหรือวิธีการเล่นเกม

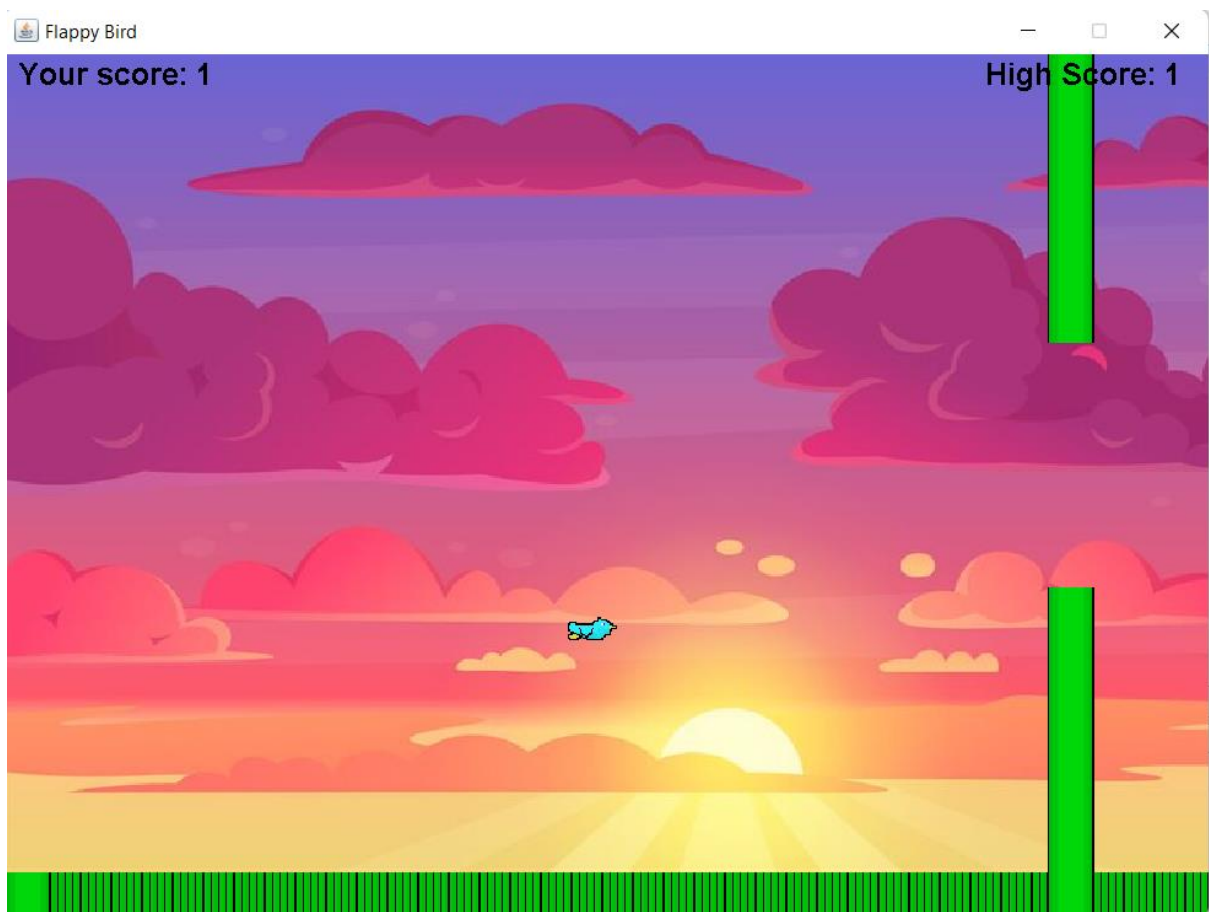
เกม Flappy Bird นั้นจะเป็นการช่วยเหลือเจ้านกตัวน้อยของเราให้ข้ามสิ่งกีดขวางไปได้ไกลที่สุด

#### วิธีการเล่น

-กด Spacebar เพื่อกระโดดหลบสิ่งกีดขวาง

-หากชนสิ่งกีดขวางจะแพ้ในทันที

-กด Enter เพื่อทำการ Restart game



## Class Diagram

<b>Class Controller</b>  public void controller(Bird bird ,KeyEvent kevent)  public void controllerReleased(Bird bird,KeyEvent kevent)	<b>Class Bird</b>  public bird(int x, inty)  public void tick()  public void jump()  private void checkWindowBorder()  public Rectangle getBounds()  public void render(Graphics2D g,ImageObserver obs)
<b>Class Tube</b>  public tube()  public void tick()  public void render(Graphics2D g, ImageObserver obs)  public Rectangle getBounds()	<b>Class TubeColumn</b>  public TubeColumn()  private void InitTubes()  public void tick()  public void render(Graphics2D g, ImageObserver obs)  public List<Tube> getTubes()  public void setTubes(List<Tube>) tubes)  public int getPoints()  public void setPoints(int points)
<b>Class Game</b>  public game()  public void actionPerfomed(ActionEvent e)  public void paint(Graphics g)  private void restartGame()	<b>Class Window</b>  public Window(int width, int height, String title, Game game)  public static void main(String[] args)

<pre>private void endgame() private void checkCollision() public gameKeyAdapter() public void keyPressed(KeyEvent e) public void keyReleased(keyEvent e)</pre>	
--	--

## Class Controller

เป็นคลาสที่ใช้กำหนดปุ่มกระโดดของนก

## Class Bird

คลาสที่สร้างตัวนกและมีเมธอดการทำงานต่าง ๆ ของนก การวาดรูปนกในแต่ละเฟรม, เมธอดเช็คการชนของนกกับสิ่งกีดขวาง

## Class Tube

เป็นคลาสที่ import รูปภาพสิ่งกีดขวาง

## Class TubeColumn

เป็นคลาสที่สุ่มตำแหน่งการสร้างสิ่งกีดขวาง,การทำลายสิ่งกีดขวางเมื่อเราผ่านสิ่งกีดขวางนั้นไปแล้ว,เมธอดกำหนดความเร็วเมื่อผ่านสิ่งกีดขวาง

## Class Game

เป็นคลาสที่กำหนด Background ของตัวเกม มีเมธอดการรับ input จากคีย์บอร์ด, การชนสิ่งกีดขวางและการ Restart game

## Class Window

เป็นคลาสที่กำหนดขนาดหน้าต่างของโปรแกรมและเป็นคลาสที่ไว้ใช้รันเกม

## ส่วนการทำงานของโปรแกรม

### Constuctor

```
public class Controller implements IStrategy {  
  
    @Override  
    public void controller(Bird bird, KeyEvent kevent) {  
    }  
  
    @Override  
    public void controllerReleased(Bird bird, KeyEvent kevent) {  
        if(kevent.getKeyCode() == KeyEvent.VK_SPACE) {  
            bird.jump();  
        }  
    }  
}
```

คลาส Controller จะมีการรับ input ปุ่ม space และมีการเรียกเมธอด jump เพื่อให้นกกระโดด



## Encapsulation

```
public class TubeColumn {  
  
    private int base = Window.HEIGHT - 60;  
  
    private List<Tube> tubes;  
    private Random random;  
    private int points = 0;  
    private int speed = 5;  
    private int changeSpeed = speed;  
  
    public TubeColumn() {  
        tubes = new ArrayList<>();  
        random = new Random();  
        initTubes();  
    }  
  
    private void initTubes() {  
  
        int last = base;  
        int randWay = random.nextInt(10);  
  
        for (int i = 0; i < 20; i++) {
```

ในคลาส TubeColumn จะมีการห่อหุ้มตัวแปรและเมธอดที่เป็น Private เช่น points, speed, random เป็นต้น

## Composition

```
public interface IStrategy {  
  
    public void controller(Bird bird, KeyEvent kevent);  
    public void controllerReleased(Bird bird, KeyEvent kevent);  
}
```

ในคลาส IStragtegy จะมีการประกาศชื่อเมธอดที่จะให้คลาสอื่นทำไปใช้ เช่น  
controller, controllerReleased

## Polymorphism

```
public class Game extends JPanel implements ActionListener {  
  
    private boolean isRunning = false;  
    private ProxyImage proxyImage;  
    private Image background;  
    private Bird bird;  
    private TubeColumn tubeColumn;  
    private int score;  
    private int highScore;
```

คลาสเกมก็จะมี Polymorphism คือการสืบทอดมาจาก JPanel แต่มีการประกาศตัวแปรเพิ่มเติมเข้าไป

## Inheritance

```
public class Bird extends GameObject {

    private ProxyImage proxyImage;
    private Tube[] tube;
    public Bird(int x, int y) {
        super(x, y);
        if(proxyImage == null) {
            proxyImage = new ProxyImage("/object/bird.gif");
        }
        this.image = proxyImage.loadImage().getImage();
        this.width = image.getWidth(null);
        this.height = image.getHeight(null);
        this.x -= width;
        this.y -= height;
        tube = new Tube[1];
        tube[0] = new Tube(900, Window.HEIGHT - 60);
        this.dy = 4;
    }
}
```

คลาส Bird ก็มีการ extends มาจากคลาส GameObject ซึ่งประกาศชื่อเมธอดเอาไว้

## GUI

```
public void paint(Graphics g) {
    Graphics2D g2 = (Graphics2D) g;
    g2.drawImage(background, 0, 0, null);

    if (isRunning) {

        this.bird.render(g2, this);
        this.tubeColumn.render(g2, this);
        g2.setColor(Color.black);
        g.setFont(new Font("Arial", 1, 20));
        g2.drawString("Your score: " + this.tubeColumn.getPoints(), 10, 20);

    } else {

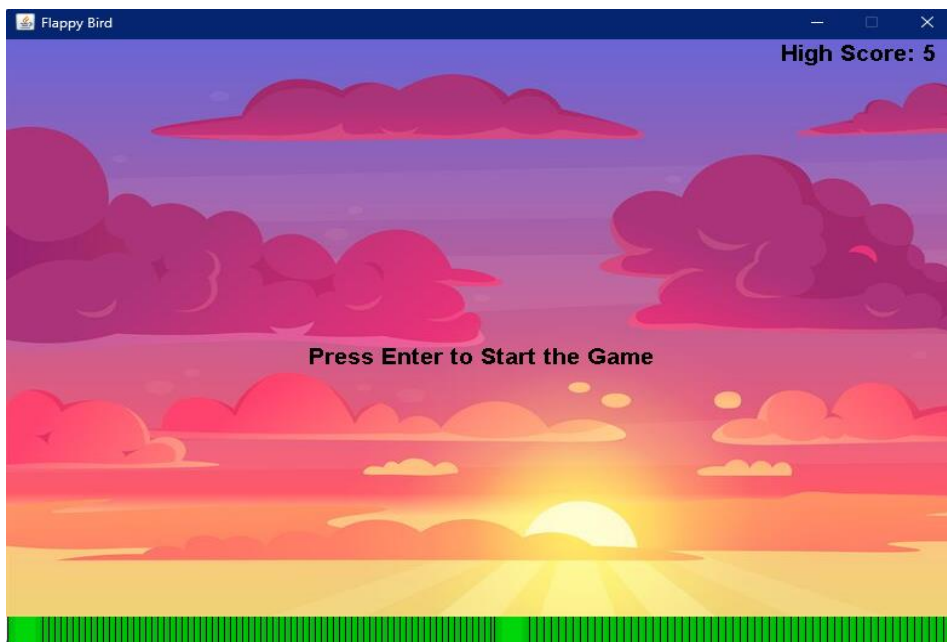
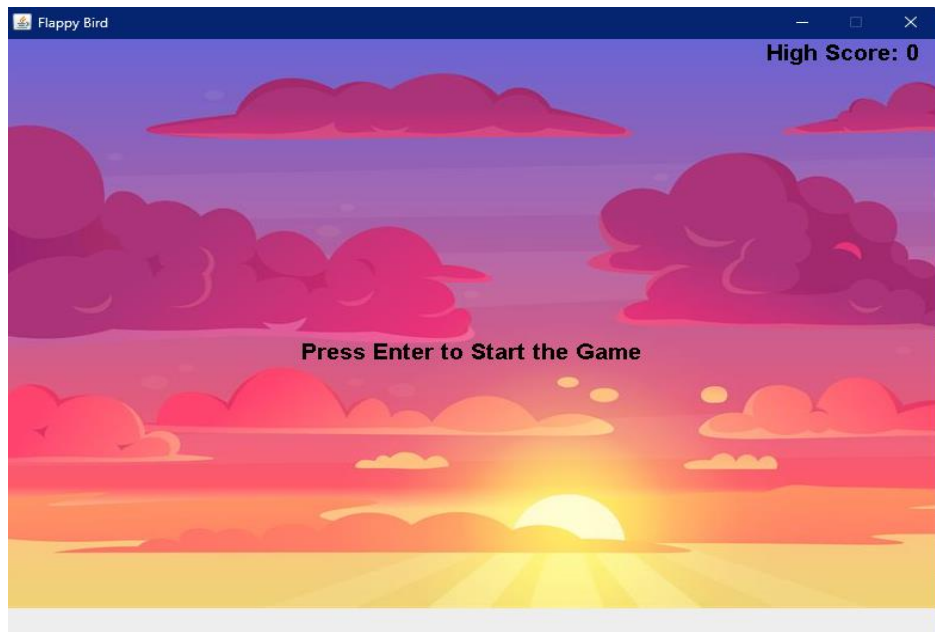
        g2.setColor(Color.black);
        g.setFont(new Font("Arial", 1, 20));
        g2.drawString("Press Enter to Start the Game", Window.WIDTH / 2 - 150, Window.HEIGHT / 2);

    }

    g2.setColor(Color.black);
    g.setFont(new Font("Arial", 1, 20));
    g2.drawString("High Score: " + highScore, Window.WIDTH - 160, 20);

    g.dispose();
}
```

GUI จะแสดงผลตอนเริ่มเกมโดยจะบอกว่าต้องกด Enter เพื่อเริ่มเกม



หน้าจอแสดงผลเมื่อชนสิ่งกีดขวางก็จะมีการอัปเดต High Score

## บทที่ 3

### สรุป

#### ปัญหาที่พบระหว่างการพัฒนา

- 1.ระหว่างการพัฒนาโปรแกรมเจอบั๊กไม่ทำลายสิ่งกีดขวางและตัวละครไม่ตายเมื่อชนสิ่งกีดขวางและฉากไม่แสดงผล
- 2.มีปัญหาตอนกดกระโดดแล้วระยะกระโดดนั้นสั้นเกินไป ทำให้ต้องกดกระโดดหลายรอบ

#### จุดเด่นของโปรแกรมนี้นี้

- 1.มีระบบการนับ High Score
- 2.ระบบเกมเข้าใจง่ายไม่ซับซ้อน
- 3.นมือนิเมชันการขยับปีกเพราะไม่ใช่ไฟล์รูปภาพปกติแต่เป็นไฟล์ GIF

#### คำแนะนำสำหรับผู้สอนที่อยากให้อธิบาย

อยากให้อธิบายตรงส่วนของเรื่อง Composition และเรื่อง Constructor ให้มากกว่านี้