



รายงาน

โปรแกรม Cat War

จัดทำโดย

นางสาวปรียาภรณ์ มุสิกะอินทร์ 6130300531 หมู่ 801/831

นำเสนอ

อาจารย์ กุลวดี สมบูรณ์วิวัฒน์

รายงานเล่มนี้เป็นส่วนหนึ่งของรายวิชา Programming Fundamentals II (03603112)

ภาคเรียนที่ 2 ปี 2563 คณะวิศวกรรมศาสตร์ศรีราชา

มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา

คำนำ

รายงานฉบับนี้เป็นส่วนหนึ่งของ Programming Fundamentals II (03603112) โดยมีจุดประสงค์เพื่อผู้จัดทำได้ฝึกการศึกษาค้นคว้าการทำโปรแกรม และนำสิ่งที่ได้ศึกษาค้นคว้ามาสร้างเป็นชิ้นงานเก็บไว้เป็นประโยชน์ต่อการเรียนการสอนของตนเองและอาจารย์ต่อไป ทั้งนี้ในรายงานเนื้อหาประกอบด้วยหลักการ เทคนิคการทำโปรแกรมและโค้ด

ผู้จัดทำได้ทำโปรแกรมตัวนี้ขึ้นมา เนื่องจากมีความน่าสนใจและขอขอบคุณอาจารย์ กุลวดี สมบูรณ์วิวัฒน์ อย่างสูงที่กรุณาตรวจ ให้คำแนะนำเพื่อแก้ไข ให้ข้อเสนอแนะตลอดการทำงาน ผู้จัดทำหวังว่ารายงานฉบับนี้คงมีประโยชน์ต่อผู้ที่นำไปใช้ให้เกิดผลตามความคาดหวัง หวังว่ารายงานฉบับนี้จะ เป็นประโยชน์แก่ผู้อ่านทุกท่าน หากมีข้อผิดพลาดประการใด ผู้จัดทำขออภัยมา ณ ที่นี้ด้วย

ผู้จัดทำ

สารบัญ

โปรแกรมเกมส์ Cat War	1
หลักการออกแบบโปรแกรม.....	1
ตัวอย่างการรันโปรแกรม	2
Source Code	4
ไฟล์ CatWar.java	4
ไฟล์ Games.java	6

โปรแกรมเกมส์ Cat War

โปรแกรมเกมส์ Cat War เป็นเกมส์หนึ่ง ที่มีตัวละคร คือ แมว 🐱 , หนู 🐭 และ
 เศษอาหาร 🍽️ โดยมีกฎก็คือ แมว ซึ่งก็คือผู้เล่น ต้องคอยปกป้องเศษอาหารจากพวกหนูและคอยจับ
 หนูกินเพื่อไม่ให้หนูเข้ามากินอาหารได้ ยิ่งแมวจับหนูกินได้มากเท่าไร คะแนนก็จะมากขึ้นเท่านั้น แต่ถ้า
 หนูสามารถเข้าถึงอาหารได้ค่าชีวิตของแมวก็นจะลดลง และเมื่อค่าชีวิตของแมวหมด ก็จะมีเกมโอเวอร์ทันที
 และก็จะมีความที่ผู้เล่นทำได้ทั้งหมดขึ้นมา วิธีการเล่นก็ง่ายๆ แค่กดปุ่มลูกศร ขึ้น ลง ซ้าย ขวา แค่ครั้ง
 เดียวแมวก็นจะเคลื่อนตัวเองโดยไม่ต้องกดปุ่มค้างไว้

หลักการออกแบบโปรแกรม

- ตัวละคร

ตัวละครแต่ตัวจะใช้วิธีการสร้างสี่เหลี่ยมผืนผ้ารอบ sprite ของแต่ละตัว ให้คั่นค่าเป็น
 Rectangle และเก็บค่าแกน x, y และขนาดความกว้าง ความสูงไว้ (ง่ายต่อการเช็คชน)

- การสุ่มเกิดของหนู

การเกิดของหนูจะใช้วิธีการสุ่ม โดยใช้ Math.random ในที่นี้จะทำการสุ่มหนูไว้ทั้งหมด 5 ตัว
 และสุ่มที่เกิดของหนู

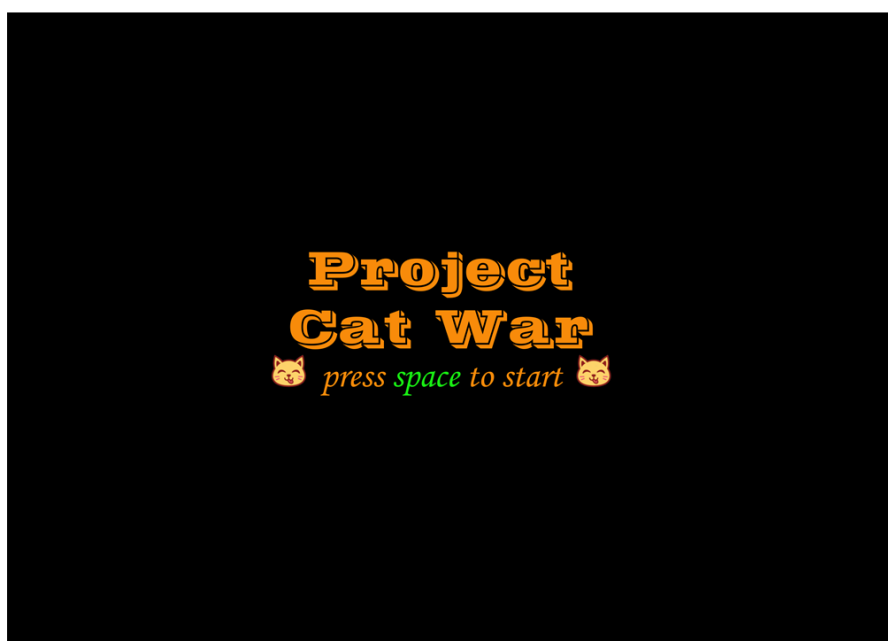
- เช็คการชน

ในส่วนนี้จะมีการเช็คการชน 2 อันคือ เช็คการชนระหว่างหนูกับแมว และเช็คการชนระหว่างหนู
 กับอาหาร โดยใช้ Rectangle.intersects ในการเช็คชน แล้วทำเงื่อนไขว่าหนูชนกับอะไรแล้วจะ
 เกิดผลอย่างไรในแต่ละอัน

- ค่าพลังชีวิต

เกมส์นี้จะให้พลังชีวิตมา 10 ชีวิต โดยจะเช็คหนูว่าหนูชนกับอาหารไหม? ถ้าเกิดการชนของหนู
 กับอาหาร ค่าพลังชีวิตก็จะลดทีละ 1 ชีวิต

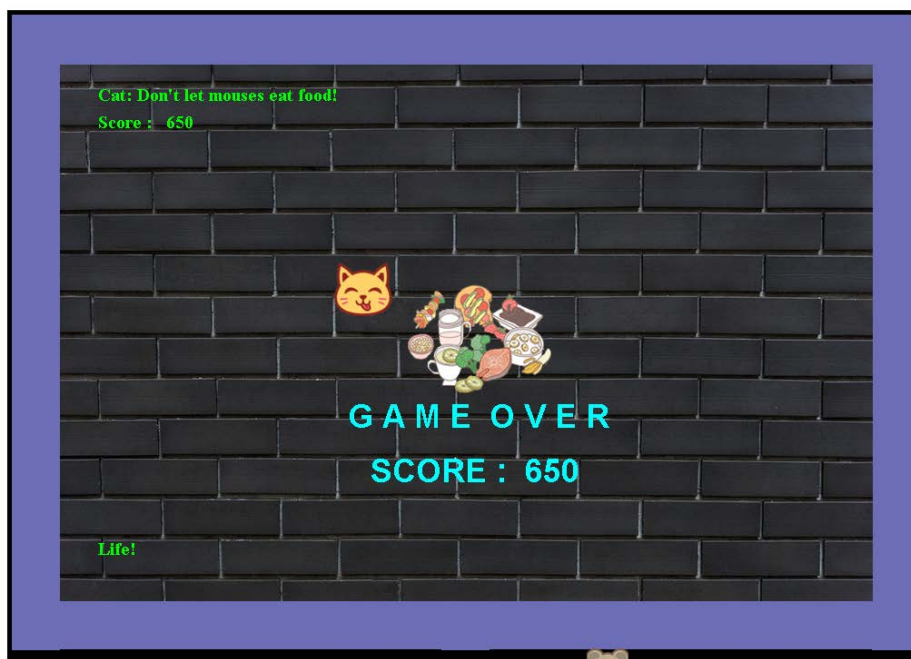
ตัวอย่างการรันโปรแกรม



หน้านี้เป็นหน้าต่างเริ่มโปรแกรมเกม ซึ่งจะให้กด space bar เพื่อที่จะเข้าสู่หน้าเกมและเริ่มเล่นเกม



หน้านี้เป็นเล่นเกม ซึ่งจะแสดงข้อความ คะแนนและค่าชีวิตของแมว โดยเราจะบังคับแมว ด้วยการกดลูกศรบน ล่าง ซ้าย ขวา แค่ครั้งเดียวไม่ต้องกดค้าง แมวก็จะเดินตามปุ่มที่เรากด ส่วนหนูจะทำการ Random ออกมา และจะวิ่งเข้าหาอาหารของแมว



หน้านี้คือถ้าหนูสามารถเข้าถึงอาหารได้ จนค่าพลังชีวิตของแมวหมด เกมก็จะจบ เมื่อเกมจบก็จะแสดงข้อความและคะแนนที่ผู้เล่นทำได้ทั้งหมดในตานี้ๆ

Source Code

ไฟล์ CatWar.java

```
import javax.swing.*;

import java.awt.*;

public class CatWar extends JFrame{

    public CatWar(){

        initCatWar();

    }

    private void initCatWar(){

        add(new Games());

        setTitle("Cat War!!");

        setSize(1024,760);

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        setLocationRelativeTo(null);

        setResizable(false);

        setBackground(Color.BLACK);

    }

    public static void main(String[] args) {

        EventQueue.invokeLater(new Runnable(){

            @Override
```

```
public void run(){  
  
    CatWar ex = new CatWar();  
  
    ex.setVisible(true);  
  
}  
  
});  
  
}  
  
}
```


ไฟล์ Games.java

```
import javax.swing.*;
```

```
import java.awt.*;
```

```
import java.awt.event.*;
```

```
import java.util.Random;
```

```
import javax.sound.sampled.*;
```

```
import java.net.URL;
```

```
import java.io.*;
```

```
class Games extends JPanel implements ActionListener, KeyListener {
```

```
    private Timer timer;
```

```
    private final int DELAY = 33;
```

```
    private Clip clip;
```

```
    private Clip intro;
```

```
    ImageGames game;
```

```
    int move = 0;
```

```
    int gstart = 0;
```

```
    int life = 10;
```

```
    int score = 0;
```

```
    int crash = 0;
```

```
    int choicem = 0;
```

```
    int livem = 0;
```

```
    int mouseX = 0;
```

```
int mouseY = 0;
```

```
int timeintro=0;
```

```
int mxcatch = 0;
```

```
int mycatch = 0;
```

```
public Games(){
```

```
    initGames();
```

```
}
```

```
private void initGames(){
```

```
    addKeyListener(this);
```

```
    setFocusable(true);
```

```
    setBackground(Color.BLACK);
```

```
    game = new ImageGames();
```

```
    timer = new Timer(Delay, this);
```

```
    timer.start();
```

```
}
```

```
public void paintComponent(Graphics g){
```

```
    super.paintComponent(g);
```

```
    draw(g);
```

```
    Messages(g);
```

```
    if (life<1) gameOver(g);
```

```

Toolkit.getDefaultToolkit().sync();

}

private void draw (Graphics g){

    Graphics2D g2 = (Graphics2D) g;

    if(gstart == 1){

        g2.drawImage(game.getBG(), 5, 5, this);

        g2.drawImage(game.getCAT(), game.getX(), game.getY(),this);

        if (crash == 1) g2.drawImage(game.getBLOOD(),mxcatch,mycatch,this);

        if (livem == 0) cremouse();

        if (choicem==0 && livem > 0 && livem < 11)
g2.drawImage(game.getMOUSE(),mouseX,mouseY,this);

        if (choicem==1 && livem > 0 && livem < 11)
g2.drawImage(game.getMOUSE(),mouseX,mouseY,this);

        if (choicem==2 && livem > 0 && livem < 11)
g2.drawImage(game.getMOUSE(),mouseX,mouseY,this);

        if (choicem==3 && livem > 0 && livem < 11)
g2.drawImage(game.getMOUSE(),mouseX,mouseY,this);

        if (choicem==4 && livem > 0 && livem < 11)
g2.drawImage(game.getMOUSE(),mouseX,mouseY,this);


        g2.drawImage(game.getFOOD(), 440, 300,this);

    }

    else if (gstart == 0){

```

```

        g2.drawImage(game.getLOGO(),290,250,this);

    }

}

public void Messages(Graphics g){

    Graphics2D g2 = (Graphics2D) g;

    Font font = new Font("Serif", Font.BOLD, 20);

    g2.setFont(font);

    g2.setColor(Color.GREEN);

    if (gstart == 1){

        g2.drawString("Score : ",100,130);

        String Score2 = Integer.toString(score);

        g2.drawString(Score2, 175, 130);

        g2.drawString("Cat: Don't let mouses eat food!", 100, 100);

        g2.drawString("Life!", 100, 600);

        int heart = life;

        while(heart > 0){

            g2.drawString("X", (150+(heart*15)), 600);

            heart -= 1;

        }

    }

}

```

```

public void gameOver(Graphics g){

    Graphics2D g2 = (Graphics2D) g;

    Font font = new Font("Helvetica", Font.BOLD, 36);

    g2.setColor(Color.CYAN);

    g2.setFont(font);

    g2.drawString("G A M E   O V E R", 375, 460);

    g2.drawString("SCORE :", 400, 520);

    String endScore = Integer.toString(score);

    g2.drawString(endScore, 570, 520);

}

```

@Override

```

public void actionPerformed(ActionEvent e){

    If (life > 0){

        if(gstart == 0 && timeintro < 1){

            timeintro += 1;

            playIntro();

        }

        if(gstart == 1){

            moveCat();

            moveMouse();

            crashFood();

            crashCat();

        }

    }
}

```

```
}

    repaint();
}

public void keyPressed(KeyEvent ke){

    int key = ke.getKeyCode();

    if (key == KeyEvent.VK_LEFT){

        move = 1;

    }

    if (key == KeyEvent.VK_RIGHT){

        move = 2;

    }

    if (key == KeyEvent.VK_UP){

        move = 3;

    }

    if (key == KeyEvent.VK_DOWN){

        move = 4;

    }

    if (key == KeyEvent.VK_SPACE){

        gstart = 1;

        if (intro.isRunning()) intro.stop();

    }

}

public void keyReleased(KeyEvent ke){

}
```

```

public void keyTyped(KeyEvent ke){

    }

}

public void moveCat(){

    if (move==1) game.moveLeft();

    else if (move==2)game.moveRight();

    else if (move==3) game.moveUp();

    else if (move==4)game.moveDown();

}

```

Random rand;

```

public void cremouse(){

    rand = new Random();

    choicem = rand.nextInt(5);

    choicem += 1;

    if (choicem == 1 || choicem == 3) mouseX = rand.nextInt(1000);

    if (choicem == 1) mouseY = 1;

    if (choicem == 2) mouseX = 1000;

    if (choicem == 2 || choicem == 4) mouseY = rand.nextInt(700);

    if (choicem == 3) mouseY = 700;

    if (choicem == 4) mouseX = 1;

```

```
lives = 1;
```

```
}
```

```
public void moveMouse(){
```

```
    if (lives > 0 && mouseX < 480) mouseX+=9;
```

```
    if (lives > 0 && mouseX > 500) mouseX-=9;
```

```
    if (lives > 0 && mouseY < 340) mouseY+=9;
```

```
    if (lives > 0 && mouseY > 340) mouseY-=9;
```

```
    lives += 1;
```

```
    if (lives == 11) lives = 1; crash = 0;
```

```
}
```

```
public Rectangle getBoundmouse(){
```

```
    game.getImageSizeMouse();
```

```
    return new Rectangle(mouseX, mouseY, game.getWidthMouse, game.getHeightMouse);
```

```
}
```

```
public void crashFood(){
```

```
    Rectangle r1 = getBoundmouse();
```

```
    Rectangle r2 = game.getBoundFood();
```

```
    if (lives == 1){
```



```

        if(r1.intersects(r2)) livem = 0;

        if(r1.intersects(r2)) life -= 1;

    }

}

public void crashCat(){

    Rectangle r1 = getBoundmouse();

    Rectangle r2 = game.getBoundcat();

    if (crash == 0 && r1.intersects(r2) && livem >1) crash = 1; mxcatch= mouseX; mycatch =
mouseY;

    if (r1.intersects(r2)) livem = 0;

    if (r1.intersects(r2)) score += 10;

    if (r1.intersects(r2)) playSound();

}

public void playSound() {

    try {

        URL url = this.getClass().getClassLoader().getResource("res/Sound/bite.wav");

        AudioInputStream in = AudioSystem.getAudioInputStream(url);

        clip = AudioSystem.getClip();

        clip.open(in);

        if (clip.isRunning())

            clip.stop();

        clip setFramePosition(0);

```

```

        clip.start();

    }catch (IOException e) {

        e.printStackTrace();

    }catch (LineUnavailableException e) {

        e.printStackTrace();

    }catch (UnsupportedAudioFileException e) {

        e.printStackTrace();

    }

}

public void playIntro() {

    try{

        URL url = this.getClass().getClassLoader().getResource("res/Sound/intro.wav");

        AudioInputStream in = AudioSystem.getAudioInputStream(url);

        intro = AudioSystem.getClip();

        intro.open(in);

        intro.start();

    }catch (IOException e) {

        e.printStackTrace();

    }catch (LineUnavailableException e) {

        e.printStackTrace();

    }catch (UnsupportedAudioFileException e) {

        e.printStackTrace();

    }

}

}

```

```
class ImageGames{

    Image bg;

    Image cat;

    Image mouse;

    Image blood;

    Image food;

    Image logo;

    Image mouse2;


    int x = 480;

    int y = 530;


    int widthfood;

    int heightfood;

    int widthcat;

    int heightcat;

    int widthmouse;

    int heightmouse;


    ImageGames(){

        initImageGames();

    }

}
```

```
public void initImageGames(){

    ImageIcon BG = new ImageIcon("res/Images/BG.png");

    ImageIcon CAT = new ImageIcon("res/Images/CAT.png");

    ImageIcon MOUSE = new ImageIcon("res/Images/MOUSE.png");

    ImageIcon BLOOD = new ImageIcon("res/Images/BLOOD.png");

    ImageIcon FOOD = new ImageIcon("res/Images/FOOD.png");

    ImageIcon LOGO = new ImageIcon("res/Images/LOGO.png");


    bg = BG.getImage();

    cat = CAT.getImage();

    mouse = MOUSE.getImage();

    blood = BLOOD.getImage();

    food = FOOD.getImage();

    logo = LOGO.getImage();

}


public Image getBG(){

    return bg;

}

public Image getCAT(){

    return cat;
```

```
}

public Image getMOUSE(){

    return mouse;

}

public Image getBLOOD(){

    return blood;

}

public Image getFOOD(){

    return food;

}

public Image getLOGO(){

    return logo;

}

public int getX(){

    return x;

}

public int getY(){

    return y;

}


public void moveUp(){

    if(y>70)

        y -= 10;

}
```

```
public void moveDown(){  
  
    if(y<580)  
  
        y += 10;  
  
}  
  
public void moveLeft(){  
  
    if(x>70)  
  
        x -= 10;  
  
}  
  
public void moveRight(){  
  
    if(x<880)  
  
        x += 10;  
  
}  
  
  
  
public void getImagesizecat(){  
  
    whidtcacat = cat.getWidth(null);  
  
    heightcat = cat.getHeight(null);  
  
}  
  
  
  
public Rectangle getBoundcat(){  
  
    getImagesizecat();  
  
    return new Rectangle(x, y, whidtcacat, heightcat);  
  
}
```

```
public void getImagesizemouse(){  
  
    whidtmouse = mouse.getWidth(null);  
  
    heightmouse = mouse.getHeight(null);  
  
}
```

```
public void getImagesizefood(){  
  
    widtfoot = food.getWidth(null);  
  
    heightfood = food.getHeight(null);  
  
}
```

```
public Rectangle getBoundfood(){  
  
    getImagesizefood();  
  
    return new Rectangle(480, 330, widtfoot, heightfood);  
  
}  
  
}  
  
}
```