

TOM AND JERRY

1.1 ที่มาและความสำคัญ

ปัจจุบันนี้ถือได้ว่าเกมมีบทบาทสำคัญสำหรับเยาวชนเป็นจำนวนมาก เนื่องจากเข้าถึงได้ง่าย ดังนั้น หากเล่นเกมที่ไม่ได้มีส่วนช่วยในการพัฒนาสมองก็อาจเป็นการเล่นเกมที่ส่งผลเสียได้ เราจึงได้พัฒนาเกมนี้ขึ้นมาเพื่อจะช่วยให้เยาวชนได้พัฒนาในด้านความรู้ ความคิด และความสามารถที่เพิ่มขึ้น

กระผมได้เห็นถึงเห็นว่าเกมนั้นมีข้อดีหลายประการเป็นอย่างมาก ซึ่งเกมนั้นอาจเป็นตัวช่วยบางตัวที่ทำให้เด็กมีความฉลาดได้เช่นกัน หรือยังเป็นตัวช่วยฝึกทักษะของเด็กได้ด้วยเช่นกัน

ซึ่งเกมที่กระผมเห็นไว้คือเกมที่ช่วยฝึกสมองเด็กให้รู้จักการแก้ปัญหาต่างๆได้และสามารถช่วยให้เด็กรู้จักที่จะควบคุมอารมณ์ของตนเอง กระผมจึงได้พัฒนาเกมตัวนี้ขึ้น

1.2 ประเภทของโครงการ : Game Development

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1.3.1 ทำให้ผู้เล่นได้เพลิดเพลินกับการเล่นเกม
- 1.3.2 ทำให้ผู้เล่นได้ฝึกทักษะในการแก้ปัญหา
- 1.3.3 ฝึกความอดทนของผู้เล่น
- 1.3.5 เป็นแนวทางในการพัฒนาต่อไป

1.4 ขอบเขตของโครงการ

- 1.4.1 การพัฒนาเกมอยู่ในรูปแบบของระบบปิด สามารถเล่นได้ครั้งละ 1 คน โดยเล่นกับเครื่องคอมพิวเตอร์ส่วนบุคคล
- 1.4.2 มีการกำหนดอุปสรรคต่างๆให้ผู้เล่นได้วางแผนในการแก้ไข
- 1.4.3 มีศัตรูที่หล่นลงมา 2 ตัวละครด้วยกัน โดยตัวที่ 1 จะเป็น Jerry ต้องรับเพื่อที่จะเก็บแต้มและป้องกันไม่ให้หนีได้ ตัวที่ 2 เป็น Bulldog หากรับแล้วจะทำให้ติดสตันระยะเวลาหนึ่ง
- 1.4.4 ระบบแสดงชีวิตที่เหลือและคะแนน

บทที่ 2 ส่วนการพัฒนา

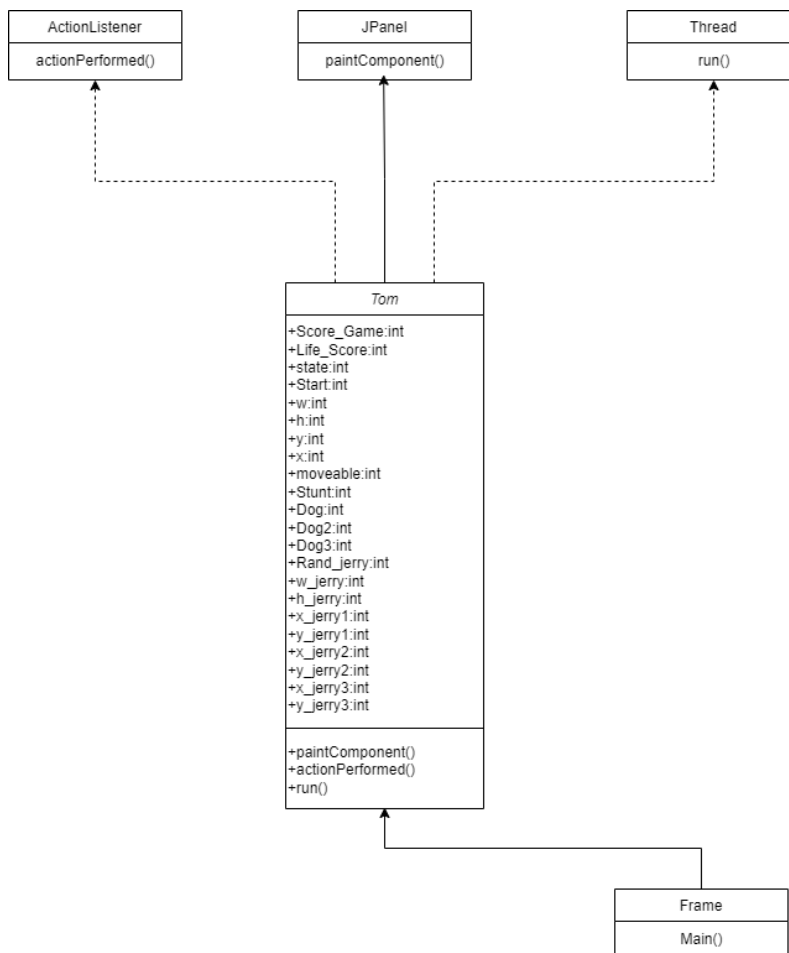
2.1 เนื้อเรื่อง

ณ บ้านแห่งหนึ่งมีคู่หูคู่กัดที่ต้องอยู่ร่วมกันนั่นคือ Jerry เจ้าหนูจอมซนชอบกินชีสเป็นชีวิตจิตใจในช่วงเวลาที่ไม่มีคนอยู่บ้านก็มักจะออกมาแอบขโมยของกินเป็นประจำ Tom แมวสุดแสบที่เจ้าของบ้านได้เลี้ยงไว้เมื่อเจ้าของบ้านไม่อยู่ก็จะเป็นแมวที่คอยเฝ้าบ้านและเมื่อ Jerry ออกมาขโมยอาหารทำให้เจ้าแมว Tom ต้องออกมาขัดขวางเสมอทำให้ทั้ง 2 ตัวละครเป็นคู่หูคู่กัดกันตลอดกาล

2.2 วิธีการเล่น

ใช้ปุ่ม A D ในการเลื่อนซ้าย-ขวา เพื่อไม่ให้เจ้าหนู Jerry หนีไปได้

2.3 คลาสไดอะแกรม



2.4 รูปแบบการพัฒนา Application: Native Application

2.5 ส่วนของโปรแกรม

- ส่วนที่ใช้ในการ Declaration

```
18 //Tom
19 class Tom extends JPanel implements ActionListener, Runnable {
20
21     Timer time = new Timer(delay: 30, this);
22     Timer Stunt_time = new Timer(delay: 2000, this);
23     int Score_Game = 0;
24     int Life_Score = 0;
25     int state = 1;
26     int Start = 0;
27     // Tom
28     int w = 70 + 90;
29     int h = 100 + 90;
30     int y = 480;
31     int x = 460;
32
33     int moveable = 1;
34     int Stunt = 0;
35
36     // Jerry
37     int Dog = 0;
38     int Dog2 = 0;
39     int Dog3 = 0;
40     int Rand_jerry = 0;
41
42     int w_jerry = 40 + 25;
43     int h_jerry = 60 + 25;
44
45     int x_jerry1 = 140;
46     int y_jerry1 = -35;
47
48     int x_jerry2 = 460;
49     int y_jerry2 = -95;
50
51     int x_jerry3 = 780;
52     int y_jerry3 = -135;
53
54     char keyChar = 'A';
55 }
```

สร้างตัวแปรที่เก็บในส่วนของรูปภาพต่างๆ

ที่จะนำไปใช้ต่อไป

สำหรับประกาศตัวแปรที่หากมีการเปลี่ยนแปลงจะมีผล

ต่อทุกส่วนในคลาส ในส่วนนี้จะมีทั้งการ:

- Extends
 - Extends JPanel เข้ามาใช้
- Implement
 - Implement ActionListener ในคลาสนี้จึงจำเป็นต้องมี Method Actionperformed ประกอบอยู่ในคลาส
 - Implement Runnable เป็นการเรียกใช้ Thread อีกรูปแบบหนึ่งในคลาสนี้จึงจำเป็นต้องมี Method run ประกอบอยู่ในคลาส

```
56 // Import Image:
57 URL imageURL = this.getClass().getResource(name: "car.png");
58 Image image = new ImageIcon(imageURL).getImage(); // Contain Image
59 // bg1
60 URL imageURL_bg = this.getClass().getResource(name: "bgv2.jpg");
61 Image image_bg = new ImageIcon(imageURL_bg).getImage(); // Contain Image
62 // bg2
63 URL imageURL_bg2 = this.getClass().getResource(name: "bg1.jpg");
64 Image image_bg2 = new ImageIcon(imageURL_bg2).getImage(); // Contain Image
65 // StartGame bg;
66 URL imageURL_bgStart = this.getClass().getResource(name: "bgStart.jpg");
67 Image image_bgStart = new ImageIcon(imageURL_bgStart).getImage(); // Contain Image
68 // Jerry;
69 URL imageURL_jerry = this.getClass().getResource(name: "jerry.png");
70 Image image_jerry = new ImageIcon(imageURL_jerry).getImage();
71 // Dog;
72 URL imageURL_Dog = this.getClass().getResource(name: "Dog.png");
73 Image image_Dog = new ImageIcon(imageURL_Dog).getImage();
74 // TomDie;
75 URL imageURL_TomStunt = this.getClass().getResource(name: "tomStunt.png");
76 Image image_TomStunt = new ImageIcon(imageURL_TomStunt).getImage();
77 // GameOver Bg;
78 URL imageURL_GameOver = this.getClass().getResource(name: "GameOver.png");
79 Image image_GameOver = new ImageIcon(imageURL_GameOver).getImage();
80
81 // ArrList
82 ArrayList<Image> JerryArr1 = new ArrayList<Image>();
83 ArrayList<Image> JerryArr2 = new ArrayList<Image>();
84 ArrayList<Image> JerryArr3 = new ArrayList<Image>();
```

ประตัวแปรสำหรับกาองค์ประกอบที่จะเพิ่มเข้าไปใน Frame

```
// JFrame
JFrame j = new JFrame();
JButton btn = new JButton(text: "RESTART");
JButton Start_btn = new JButton(text: "START");
JLabel Score = new JLabel("SCORE: " + "20");
JLabel Life = new JLabel("LIFE: " + "5");
JPanel Score_bar = new JPanel(new GridLayout(rows: 1, cols: 1));
```

- Constructor

```

96 // Constructor
97 public Tom() {
98     JerryArr1.add(image_jerry);
99     JerryArr1.add(image_Dog);
100     Score_bar.setBackground(new Color(r: 151, g: 71, b: 70));
101     btn.setBackground(new Color(r: 151, g: 71, b: 70));
102     btn.setForeground(Color.white);
103     Start_btn.setBackground(new Color(r: 151, g: 71, b: 70));
104     Start_btn.setForeground(Color.white);
105     Score_bar.setPreferredSize(new Dimension(width: 5, height: 30));
106     Score.setFont(new Font(name: "CocsoLas", Font.PLAIN, size: 15));
107     Life.setFont(new Font(name: "CocsoLas", Font.PLAIN, size: 15));
108     Score_bar.add(Score);
109     Score_bar.add(Life);
110     Score_bar.add(Start_btn);
111     Score_bar.add(btn);
112     // btn.setBounds(50, 50, 50, 50);
113     j.setLayout(new BorderLayout());
114
115     j.add(this);
116     j.add(Score_bar, BorderLayout.NORTH);
117     // j.add(btn, BorderLayout.SOUTH);
118
119     j.setTitle(title: "Tom and Jerry");
120     j.setSize(width: 1000, height: 800);
121     j.setResizable(resizable: false);
122     j.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
123     j.setLocationRelativeTo(c: null);
124     j.setVisible(b: true);
125
126     setFocusable(focusable: true);
127     requestFocus();
128     setVisible(aFlag: true);
129

```

ส่วนนี้ใช้ในการเพิ่มองค์ประกอบหรือ Add สิ่ง
ต่างๆ ที่เตรียมไว้เข้าไปใน JFrame หรือ
Jpanel และเป็นส่วนในการตั้งค่ารูปแบบจอ
และการแสดงผล

```

130 // Key Listener-----
131 btn.addActionListener(new ActionListener() {
132     public void actionPerformed(java.awt.event.ActionEvent e) {
133         Life_Score = 5;
134         requestFocus();
135     }
136 });
137
138 Start_btn.addActionListener(new ActionListener() {
139     public void actionPerformed(java.awt.event.ActionEvent e) {
140         Start = 1;
141         Life_Score = 5;
142         y_jerry1 = -35;
143         y_jerry2 = -95;
144         y_jerry3 = -135;
145         requestFocus();
146     }
147 });
148

```

ActionListener ของปุ่มที่ชื่อ “btn”

```

Start_btn.addActionListener(new ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent e) {
        Start = 1;
        Life_Score = 5;
        y_jerry1 = -35;
        y_jerry2 = -95;
        y_jerry3 = -135;
        requestFocus();
    }
});

```

ActionListener ของปุ่มที่ชื่อ “Start_btn”

```

149         addKeyListener(new KeyAdapter() {
150             @Override
151             public void keyPressed(KeyEvent e) {
152                 if (moveable == 1) {
153                     switch (e.getKeyCode()) {
154                         case KeyEvent.VK_A:
155                             x -= 20;
156                             break;
157                         case KeyEvent.VK_D:
158                             x += 20;
159                             break;
160                         default:
161                             keyChar = e.getKeyChar();
162                     }
163                     if (x >= 780) {
164                         x = 780;
165                     } else if (x <= 140) {
166                         x = 140;
167                     }
168                     repaint();
169                 }
170             }
171         });
172
173     } // End constructor;

```

Key Listener ใช้สำหรับรอรับค่าจากแป้นพิมพ์

- ส่วนที่ใช้ในการวาดรูปวัตถุต่างๆ (paintComponent)

```

175 // Paint
176 protected void paintComponent(Graphics g) {
177     Thread ts = new Thread(new Runnable() {
178         public void run() {
179             while (Stunt != 0) {
180                 try {
181                     Thread.sleep(millis: 1000);
182                     Stunt = 0;
183                     moveable = 1;
184                 } catch (InterruptedException e) {
185                 }
186             }
187         }
188     });
189     Graphics2D g2d = (Graphics2D) g;
190     super.paintComponent(g);
191     if (Start == 0) {
192         // Start Dog;
193         g.drawImage(image_bgStart, x: 0, y: 0, getWidth(), getHeight(), this);
194     } else if (Life_Score > 0 && Start == 1) {
195         // Background game;
196         g.drawImage(image_bg, -30, y: 0, getWidth() + 40, getHeight(), this);
197         if (Stunt == 0) {
198             g.drawImage(image, x, y, w, h, this);
199         } else if (Stunt == 1) {
200             moveable = 0;
201             g.drawImage(image_TomStunt, x, y, w, h, this);
202             ts.start();
203         }
204         Rectangle tomRect = new Rectangle(x, y, w, h);
205         // g2d.draw(tomRect);
206         // 1
207         Rectangle JerryRect = new Rectangle(x_jerry1, y_jerry1, w_jerry, h_jerry);
208         // g2d.draw(JerryRect);
209         g.drawImage(JerryArr1.get(Dog), x_jerry1, y_jerry1, w_jerry, h_jerry, this);
210         // 2

```

```

211         Rectangle JerryRect2 = new Rectangle(x_jerry2, y_jerry2, w_jerry, h_jerry);
212         // g2d.draw(JerryRect2);
213         g.drawImage(JerryArr1.get(Dog2), x_jerry2, y_jerry2, w_jerry, h_jerry, this);
214         // 3
215         Rectangle JerryRect3 = new Rectangle(x_jerry3, y_jerry3, w_jerry, h_jerry);
216         // g2d.draw(JerryRect3);
217         g.drawImage(JerryArr1.get(Dog3), x_jerry3, y_jerry3, w_jerry, h_jerry, this);
218
219         if (tomRect.intersects(JerryRect)) {
220             if (Dog == 0) {
221                 Score_Game += 1;
222                 y_jerry1 = 0;
223                 double rand_Dog = Math.random();
224                 if (rand_Dog > 0.5) {
225                     Dog = 1;
226                 } else if (rand_Dog < 0.5) {
227                     Dog = 0;
228                 }
229             } else if (Dog == 1) {
230                 y_jerry1 = 0;
231                 Stunt = 1;
232                 double rand_Dog = Math.random();
233                 if (rand_Dog > 0.5) {
234                     Dog = 1;
235                 } else if (rand_Dog < 0.5) {
236                     Dog = 0;
237                 }
238             }

```

```

239     } else if (tomRect.intersects(JerryRect2)) {
240         if (Dog2 == 0) {
241             Score_Game += 1;
242             y_jerry2 = 0;
243             double rand_Dog = Math.random();
244             if (rand_Dog > 0.5) {
245                 Dog2 = 1;
246             } else if (rand_Dog < 0.5) {
247                 Dog2 = 0;
248             }
249         } else if (Dog2 == 1) {
250             Stunt = 1;
251             y_jerry2 = 0;
252             double rand_Dog = Math.random();
253             if (rand_Dog > 0.5) {
254                 Dog2 = 1;
255             } else if (rand_Dog < 0.5) {
256                 Dog2 = 0;
257             }
258         }
259     } else if (tomRect.intersects(JerryRect3)) {
260         if (Dog3 == 0) {
261             Score_Game += 1;
262             y_jerry3 = 0;
263             double rand_Dog = Math.random();
264             if (rand_Dog > 0.5) {
265                 Dog3 = 1;
266             } else if (rand_Dog < 0.5) {
267                 Dog3 = 0;
268             }

```

```

268     }
269     } else if (Dog3 == 1) {
270         Stunt = 1;
271         y_jerry3 = 0;
272         double rand_Dog = Math.random();
273         if (rand_Dog > 0.5) {
274             Dog3 = 1;
275         } else if (rand_Dog < 0.5) {
276             Dog3 = 0;
277         }
278     }
279 }
280
281 } else if (Life_Score <= 0 && Start == 1) {
282     g.drawImage(image_bg2, x: 0, y: 0, getWidth(), getHeight(), this);
283     // g.drawImage(image_TomStunt, 460, 400, w + 50, h + 50, this);
284     // g.drawImage(image_GameOver, 380, 90, w + 180, h + 180, this);
285     Score_Game = 0;
286     y_jerry1 = -35;
287     y_jerry2 = -95;
288     y_jerry3 = -135;
289
290     time.start();
291     repaint();
292 }

```

- ส่วนที่คุมแอดชั่นต่างๆ(actionPerformed)

```

294 // Action
295 @Override
296 public void actionPerformed(java.awt.event.ActionEvent e) {
297     y_jerry1 += 3 + (Score_Game / 10);
298     y_jerry2 += 3 + (Score_Game / 10);
299     y_jerry3 += 3 + (Score_Game / 10);
300
301     Score.setText("SCORE: " + Score_Game);
302     Life.setText("LIFE: " + Life_Score);
303 }

```

- Thread

```

306 public void run() {
307     // Background Music;
308     try {
309         AudioInputStream inputStream = AudioSystem.getAudioInputStream(new File(pathname: "BGM.wav"));
310         Clip clip = AudioSystem.getClip();
311         clip.open(inputStream);
312         clip.loop(count: 10);
313     } catch (UnsupportedAudioFormatException | IOException e1) {
314         // TODO: Auto-generated catch block
315         e1.printStackTrace();
316     } catch (LineUnavailableException e) {
317         // TODO: Auto-generated catch block
318         e.printStackTrace();
319     }
320     while (true) {
321         double rand_dog = Math.random();
322         double rand_dog2 = Math.random();
323         double rand_dog3 = Math.random();
324         if (Start == 1) {
325             if (y_jerry1 >= 700) {
326                 if (Dog == 0) {
327                     y_jerry1 = 0;
328                     Life_Score -= 1;
329                 } else if (Dog == 1) {
330                     y_jerry1 = 0;
331                 }
332                 // double rand_dog = Math.random();
333                 if (rand_dog > 0.5) {
334                     Dog = 1;
335                 } else if (rand_dog < 0.5) {
336                     Dog = 0;
337                 }
338                 try {
339                     Thread.sleep(millis: 1000);
340                 } catch (Exception e) {

```

```

340     } catch (Exception e) {
341     }
342     } else if (y_jerry2 >= 700) {
343         if (Dog2 == 0) {
344             y_jerry2 = 0;
345             Life_Score -= 1;
346         } else if (Dog2 == 1) {
347             y_jerry2 = 0;
348         }
349         if (rand_dog2 > 0.5) {
350             Dog2 = 1;
351         } else if (rand_dog2 < 0.5) {
352             Dog2 = 0;
353         }
354         try {
355             Thread.sleep(millis: 1000);
356         } catch (Exception e) {
357         }
358     } else if (y_jerry3 >= 700) {
359         if (Dog3 == 0) {
360             y_jerry3 = 0;
361             Life_Score -= 1;
362         } else if (Dog3 == 1) {
363             y_jerry3 = 0;
364         }
365         if (rand_dog3 > 0.5) {
366             Dog3 = 1;
367         } else if (rand_dog3 < 0.5) {
368             Dog3 = 0;
369         }
370         try {
371             Thread.sleep(millis: 1000);
372         } catch (Exception e) {
373         }

```

- Main Program

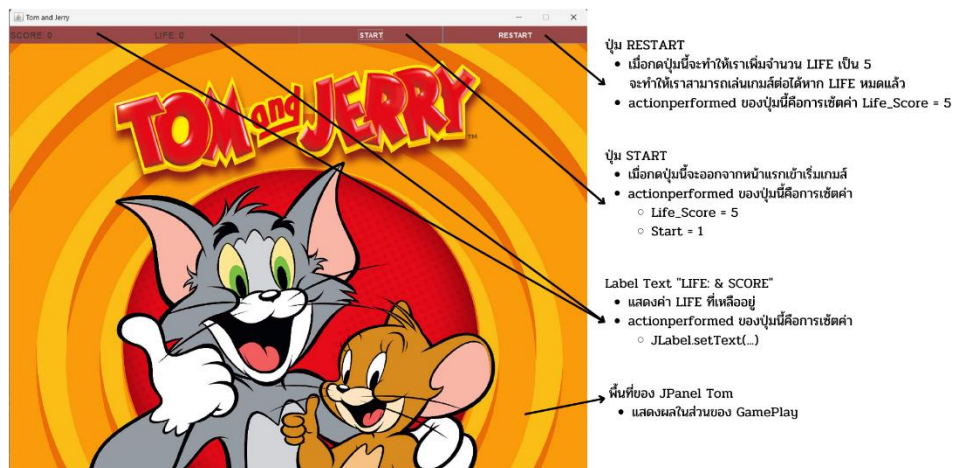
```

382 // Main Program:
383 public class Frame {
384     Run | Debug
385     public static void main(String[] args) {
386         Tom tt = new Tom();
387         Thread t1 = new Thread(tt);
388         t1.start();
389     }

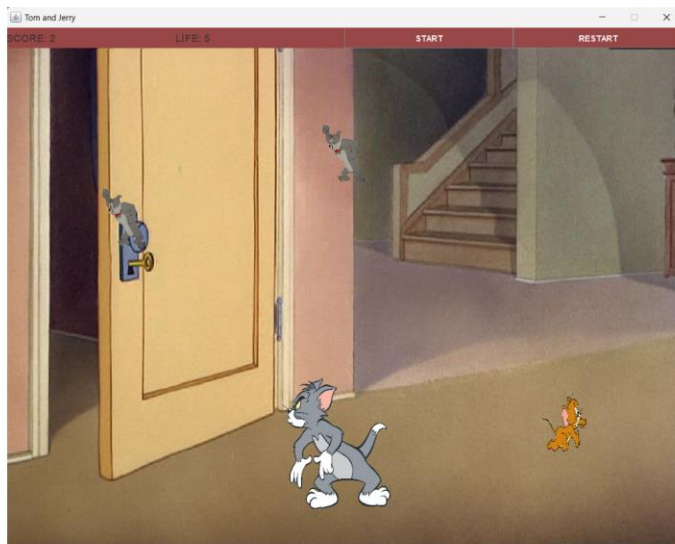
```

2.6 GUI

หน้าแรก



หน้า Game Play



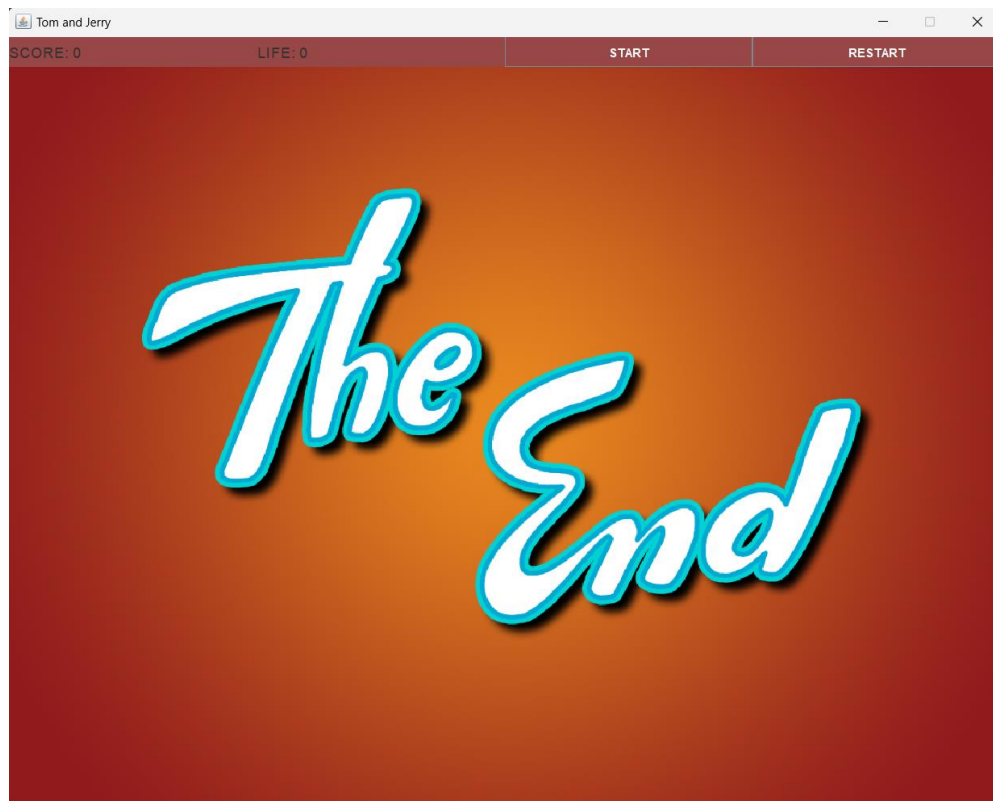
หน้า Game Play

รูปแบบเกมส์

- ตัวละคร
 - Tom(แมว)
 - Jerry(หนู)
 - Bulldog(สุนัข)
- โดยเราจะเล่นเป็น Tom หน้าที่ยังเราคือการจับหนูไม่ให้ผ่านลงไประดับล่างสุดได้ ถ้าหากจับได้ SCORE จะเพิ่มขึ้น แต่ถ้าหากหลุดผ่านไปได้ค่า LIFE จะลดลงซึ่งถ้าหาก LIFE หมดเท่ากับ GAME OVER
- Bulldog เป็นตัวละครพิเศษถ้าหากเราเลื่อนไปจับ Bulldog จะทำให้ Tom ติดสตันเป็นระยะเวลาหนึ่ง



หน้า Game Over



2.7 Algorithm ที่สำคัญ

1. การบวกแต้มเมื่อจับ Jerry ได้
2. การลบแต้มเมื่อ Jerry หลุดไปได้
3. การติดสตันเมื่อไปจับ Bulldog
4. การสุ่มตัวละครระหว่าง Jerry และ Bulldog
5. Game Over เมื่อค่า LIFE หมด

บทที่ 3 สรุป

3.1 ปัญหาที่พบระหว่างการพัฒนา

- 3.1.1 ปัญหาการใช้ Thread ทำให้โปรแกรมกระตุกหรือช้าลงบางจังหวะ
- 3.1.2 ปัญหาเสียงเพลงพื้นหลังไม่สามารถควบคุมระดับเสียงได้
- 3.1.3 ฟังก์ชันการติดสตันไม่เกิดการตอบสนองบางช่วงเวลา
- 3.1.4 ปัญหาในการดักจับการชนกันระหว่าง Tom กับ Jerry และ Bulldog
- 3.1.5 ปัญหาเมื่อมี Event เกิดขึ้นเช่นกดปุ่ม RESTART ทำให้เกิดการไม่ FOCUS ที่จอเกมส์

3.2 จุดเด่นของโปรแกรม

- 3.2.1 มีเพลงประกอบในการเล่น
- 3.2.2 มีภาพตัวละครที่สวยงามและให้ความรู้สึก Classic นึกถึงสมัยเด็ก
- 3.2.3 มีฟังก์ชันติดสตันทำให้เพิ่มความท้าทายในการเล่นเกมส์

3.3 คำแนะนำสำหรับผู้สอน: Lab อาจจะค่อนข้างเยอะนิดนึงครับส่วนในเรื่องการสอนในชั้นเรียนถือว่าดีมากครับเหมือนกับได้เรียนคอร์สดีๆ คอร์สนึงเลยครับ