## BỘ GIÁO DỤC VÀ ĐẠO TẠO TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI

-----000------

### KHOA CÔNG NGHỆ THÔNG TIN



## BÁO CÁO BÀI TẬP LỚN MÔN CÔNG NGHỆ JAVA.

Nhóm: 8

Chủ đề: Game Pikachu.

Sinh viên thực hiện: Nguyễn Hà Trường: 211213563.

Vũ Đức Khải: 211213620.

Lóp: CNTT 5 - K62.

Hà Nội ngày 01 tháng 05 năm 2023.

# Mục Lục

I)LỜI NÓI ĐẦU:	3
II)GAME PIKACHU	4
a) Giới thiệu chung về giao diện của game	
b) Mô tả cách chơi	
c) Code minh họa.	
d) Những điểm nổi bật của game pikachu	
III) HƯỚNG PHÁT TRIỂN GAME	
IV) ĐƯỜNG DẪN GITHUB	
,	

#### I)LỜI NÓI ĐẦU:

Pikachu có lẽ là tựa game quen thuộc với tất cả tuổi thơ của chúng ta. Từ lúc khi mà các game online chưa đa dạng và phát triển như hiện nay thì game pikachu là trò chơi đã làm mưa làm gió và gây sốt không chỉ với giới trẻ mà ngay cả trẻ nhỏ và người lớn. Với lối chơi đơn giản nhưng pikachu không nhàm chán mà thực sự cuốn hút. Vậy, cùng tìm hiểu kỹ hơn những điều nổi bật của game pikachu và cách chơi game pikachu ra sao để có thể đạt được số điểm cao nhất nhé!

Pikachu là một con vật đáng yêu và gắn liền với tuổi thơ của các thế hệ 8x và 9x. Không chỉ xuất hiện trong truyện tranh hay phim hoạt hình, nhân vật Pikachu còn xuất hiện trong game, điều này đem lại sự hứng thú rất lớn cho người chơi.

Mặc dù, cho tới thời điểm hiện tại, các trò chơi điện tử ngày càng đa dạng và phát triển với những tính năng hấp dẫn cùng lối chơi cuốn hút. Thế nhưng, pikachu vẫn là game mà cho dù cũ nhưng vẫn có một ý nghĩa nhất định và khó có thể thay thế trong lòng những người đã từng chơi trò chơi này.

#### II)GAME PIKACHU.

a) Giới thiệu chung về giao diện của game.



- Giao diện chính của game: là 64 hình ảnh trong đó có 21 hình ảnh khác nhau, là các hình nhân vật ngộ nghĩnh.
- New game: cho phép người chơi có thể bắt đầu một game mới.

- Score: là điểm số mà chúng ta đạt được khi tìm được các cặp hình giống nhau .Mỗi cặp hình giống nhau thì ta sẽ thu được 10 điểm công vào .
- Time: Chính là thanh thời gian chơi của chúng ta. Khi thanh thời gian chạy hết là lúc chúng ta hết thời gian chơi.
- b) Mô tả cách chơi.

Để chơi game pikachu cổ điển rất đơn giản bạn chỉ cần sử dụng chuột và tìm ra các cặp thú giống nhau để loại bỏ khỏi màn hình trò chơi và ghi điểm.

Đồng hồ bấm giờ sẽ chạy khi bạn bắt đầu chơi, hãy tinh mắt và nhanh nhẹn để kết thúc trò chơi trước giờ nếu không bạn sẽ bị thua đấy.

Bạn chọn các cặp thú sao cho khi trên màn hình không còn hình thú nào nữa hết, như vậy bạn đã chiến thắng và chuyển qua màn hình chơi khác

- c) Code minh hoa.
- Code giao diện của game.(MainFrame.java)

```
package controller;

import java.awt.BorderLayout;

public class MainFrame extends JFrame implements ActionListener, Runnable {
    private int MAX TIME = 300;
    public int time = MAX_TIME;
    private int row = 8;
    private int row = 8;
    private int width = 700;
    private int height = 500;
    public JLabel lbScore;
    private JBruton btnNewGame;
    private JButton btnNewGame;
    private JButton btnHelp;
    private JPanel mainPanel;
    private boolean pause=false;
    private boolean resume= false;
    public MainFrame() {
        add (mainPanel = createMainPanel());
        setTitle("Pokemon Game");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(width, height);
        setVisible(true);

private JPanel createMainPanel() {
        JBanel panel = new JPanel(new BorderLayout());
        private JPanel createMainPanel();
        private JPanel createMainPane
```

```
private JPanel createMainPanel() {

JPanel panel = new JPanel(new BorderLayout());

panel.add(createGraphicsPanel(), BorderLayout.CENTER);

panel.add(createControlPanel(), BorderLayout.EAST);

return panel;

private JPanel createGraphicsPanel()

graphicsPanel = new ButtonEvent(this, row, col);

JPanel panel = new JPanel(new GridBagLayout());

panel.setBackground(Color.pink);

panel.add(graphicsPanel);

return panel;

private JPanel createControlPanel()

for a progressTime = new JProgressBar(0, 100);

progressTime = new JProgressBar(0, 100);

progressTime.setValue(100);

JPanel panelLeft = new JPanel (new GridLayout(2, 1, 5, 5));

panelLeft.add(new JLabel("Time:"));

JPanel panelCenter = new JPanel (new GridLayout(2, 1, 5, 5));

panelCenter.add(lbScore);

panelCenter.add(progressTime);
```

```
panelCenter.add(progressTime);

JFanel panelScoreAndTime = new JFanel(new BorderLayout(5, 0));
panelScoreAndTime.add(panelLeft, BorderLayout.WEST);
panelScoreAndTime.add(panelCenter, BorderLayout.CENTER);

JFanel panelControl = new JFanel(new BorderLayout(10, 10));
panelControl.setBorder(new EmptyBorder(10, 3, 5, 3));
panelControl.add(panelScoreAndTime, BorderLayout.CENTER);
panelControl.add(panelScoreAndTime, BorderLayout.CENTER);

//thêm nút tro giúp
panelControl.add(btnNewGame = createButton("New Game"), BorderLayout.PAGE_START);

//thêm nút tro giúp
panelControl.add(btnHelp = createButton("Tro giúp"), BorderLayout.PAGE_END);

JFanel panel = new JFanel(new BorderLayout());
panel.setBorder(new TitledBorder("Good luck"));
panel.add(panelControl, BorderLayout.PAGE_START);
return panel;

// create a button
private JButton createButton(String buttonName)

JButton btn = new JButton(buttonName);
btn.addActionListener(this);
return btn;
```

- Code thuật toán của game.(Cotroller.java)

```
1 package controller;
3© import java.awt.Point;∏
7 public class Controller {
      private int col;
      private int[][] matrix;
       public Controller(int row, int col) {
          System.out.println(row + "," + col);
           createMatrix();
           showMatrix();
18●
       public void showMatrix() {
               for (int j = 0; j < col ; j++) {
                   System.out.printf("%3d", matrix[i][j]);
               System.out.println();
           matrix = new int[row][col];
           Random rand = new Random();
           int imgCount = 21;
           int max = 4;
           int arr[] = new int[imgCount + 1];
           ArrayList<Point> listPoint = new ArrayList<Point>();
           for (int i = 0; i < row; i++) {
```

Ta coi dao diện chính của game là một ma trận.

Ta có 21 hình ảnh khác nhau nên ta sẽ khai báo biến imgCount là 21.

Max = 4 là ta khai bao cho số ảnh giống nhau trong ma trận tối thiếu là 4

Int arr là để khai bao giá trị ban đầu của mỗi ảnh đều bằng 0.

Sau đó ta dùng ArrayList để lưu các listPoint vào trong Point bằng hai vòng lặp for.

Ta khai báo một biến i đại diện cho số cặp pikachu giống nhau .Ta dùng biến index để lấy giá trị ngẫu nhiên của 1 trong 21 ảnh khác nhau. Trong vòng lặp if ta có phép so sánh để so sánh số hình piakchu giống nhau tối thiểu trong một giao diện game.

arr[index] += 2;

nếu nhỏ hơn max thì ta cộng thêm 2 vào biến arr.

Vòng lặp for để tạo ra hai cặp pikachu giống nhau trong 1 ma trận.

int size = listPoint.size(); để tạo ra một biến mới lưu về kích thước của list.

int pointIndex = rand.nextInt(size); để tạo ra một biến mới lấy giá trị ngẫu nhiên trong khoảng từ 0 đến size -1.

matrix[listPoint.get(pointIndex).x][listPoint.get(pointIndex).y] = index;

lúc nãy sẽ lưu điểm có tọa độ x,y trên ma trận là index (mục đích để chọn hình ảnh đã lưu ở trên).

loại bỏ điểm này khỏi listP để tránh lần lấy PointIndex sau sẽ trùng số lần trước lưu giá trị.

While để kiểm tra i là số cặp pikachu trung nhau nên ta chỉ cho giới hạn chúng chạy tới đó.

public PointLine checkTwoPoint (Point p1, Point p2) dùng để kiếm tra hai biến pikachu có giống nhau không.

```
public int getRow() {
    return row;
}

public void setRow(int row) {
    this.row = row;
}

public int getCol() {
    return col;
}

public void setCol(int col) {
    this.col = col;
}

public int[][] getMatrix() {
    return matrix;
}

public void setMatrix(int[][] matrix) {
    this.matrix = matrix;
}
```

- Code nhận thông tin về các click chuột trên các button và xử lý theo hàm đã xây dựng (ButtonEvent.java).

```
1 package controller;
20 import java.io.BufferedWriter;
21 public class ButtonEvent extends JPanel implements ActionListener {
22     private int row;
23     private int col;
24     private int size = 50;
25     private int size = 50;
26     private Point pl = null;
27     private Point pl = null;
28     private Point pl = null;
29     private JButton[][] btn;
30     private Mutton[][] btn;
31     private JButton[][] btn;
32     private Controller controller;
33     private MainFrame frame;
34     private MainFrame frame;
37     public ButtonEvent (MainFrame frame, int row, int col)
39     {
40          this.frame = frame;
41          this.row = row;
42          this.col = col;
43          item = row * col / 2;
44          setBackground (backGroundColor);
45          setLayout (new GridLayout (row, col, bound, bound));
46          setBackground (backGroundColor);
47          setPreferredSize (new Dimension ((size + bound) * col, (size + bound) * row));
48          setBorder (new EmptyBorder (10, 10, 10, 10));
49          setAlignmentY (JPanel.CENTER_ALIGNMENT);
```

```
newGame();
55●
      public void newGame()
         controller = new Controller(this.row, this.col);
         addArrayButton();
62●
      private void addArrayButton()
         btn = new JButton[row][col];
                btn[i][j] = createButton(i + ", " + j);
                Icon icon = getIcon(controller.getMatrix()[i][j]);
                btn[i][j].setIcon(icon);
                add(btn[i][j]);
             }
770
         int width = 48, height = 48;
```

```
writer.write(score + "\n");
                writer.close();
                e.printStackTrace();
155●
        public void Help()
            for (int i = 0; i < row; i++)
                for (int j = 0; j < col-1; j++)
                    Point p1 = new Point(i, j);
                    Point p2 = new Point(i, j + 1);
                    PointLine check = controller.checkTwoPoint(p1, p2);
                    if (check != null) //
                        System.out.println("Tro giúp thành công");
                        controller.getMatrix()[p1.x][p1.y] = 0;
                        controller.getMatrix()[p2.x][p2.y] = 0;
                        controller.showMatrix();
                        execute(p1, p2);
                        item--;
                        frame.lbScore.setText(score + "");
```

- Code kiểm tra các cặp giống nhau (PointLine.java).

```
1 package controller;
2
3 import java.awt.Point;
4
5
6 public class PointLine {
7    public Point p1;
8    public Point p2;
9
10    public PointLine(Point p1, Point p2) {
11         this.p1 = p1;
12         this.p2 = p2;
13    }
14
15
16 }
17
```

- Code chính để chạy game (Main.java).

d) Những điểm nổi bật của game pikachu.

Không phải tự dưng mà một trò chơi có lối chơi đơn giản như Pikachu lại nhanh chóng được công chúng đón nhận và cho đến thời điểm hiện tại thì vẫn là tựa game được yêu thích nhất. Điều này xuất phát từ việc trò chơi này sở hữu những tính năng đặc biệt mà không phải trò chơi nào cũng có.

- Là một tựa game có thể nói là vừa cổ điển vừa hiện đại. Bạn có thể chơi mà không cần đến việc kết nối với mạng.
- Sử dụng hình ảnh đa dạng với các linh vật khác nhau trong pokemon. Màu sắc bắt mắt, thu hút.
- Lối chơi đơn giản, dễ hiểu và cuốn hút. Điều này khiến người chơi khó có thể dứt ra được khi chưa giành được phần thắng cho mình.
- Hiện nay, game Pikachu phù hợp với đa dạng các nền tảng khác nhau.
- Người chơi có thể chơi Pikachu một cách hoàn toàn miễn phí.
- Có thêm các tính năng tiện ích cho người chơi như gợi ý, xoay bàn chơi,...giúp người chơi trong thế bí có thể sử dụng để tiếp tục phần chơi của mình.
- Có bảng xếp hạng giúp người chơi có thể ghi tên mình vào đó với số điểm cao. Tạo động lực cho người chơi cố gắng để vượt qua được các mốc điểm của mình.

#### III) HƯỚNG PHÁT TRIỂN GAME.

- Chúng ta có thể thiết kế thuật toán làm sao cho khi ta chọn hai hình nhân vật hoạt hình giống nhau thì nó sẽ hiện đường đi của hai hình đó.
- Chúng ta có thể mở rộng kích thước chơi của game thành các cỡ ma trận lớn hơn để thêm độ khó cho game và tăng mức điểm của game lên.

#### IV) ĐƯỜNG DẪN GITHUB.

Link github source code: <a href="https://github.com/TomVNok/BTL\_JAVA">https://github.com/TomVNok/BTL\_JAVA</a>.