**TRƯỜNG ĐẠI HỌC THÔNG TIN LIÊN LẠC.**

*Khoa Công Nghệ Thông Tin*

*Bộ môn: Kỹ thuật lập trình*

…………..o0o…………..



**BÀI BÁO CÁO**

**ĐỀ TÀI: LẬP TRÌNH JAVA GAME CON RẮN.**

**Sinh viên thực hiện:**

1.Phạm Văn Nam(nt)

2.Pham Hồng Thái

3.Nguyễn Lữ Lâm Thao

4.Lê Thị Khang

Nha Trang. ngày 05 tháng 06 năm 2017.

***Phần I.. Giới thiệu mục đích về bài báo cáo :***

Java là một môn công nghệ xây dựng ứng dụng phần mềm có vị trí rất lớn trong những năm cuối thế kỉ 20, đầu thế kỉ 21. Nó được coi là công nghệ mang tính cách mạng và kahr thi nhất trong việc tạo ra các ứng dụng có khả năng chạy thống nhất trên nhiều nền tảng mà chỉ cần biên dịch một lần.

Lần đầu tiên xuất hiện vào năm 1992 như là một ngôn ngữ dung trong nội bộ tập đoàn Sun Microsystems để xây dựng ứng dụng điều khiển các bộ xử lí bên trong máy điện thoại cầm tay,lò vi song, các thiết bị điện tử khác. Không chỉ là một ngôn ngữ, Java còn là một nền tảng phát triển và triển khai ứng dụng trong đó máy ảo Java bộ thông dịch có vai trò trung tâm.

Sun, công ty đã phát minh ra ngôn ngữ Java,chính thức ban hành bản Java Development Kit1.0 vào năm 1996 hoàn toàn miễn phí để các nhà phát triển có thể tải về ,học Java, xây dựng các ứng dụng Java và triển khai chúng trên các hệ điều hành có hỗ trợ Java. Khi nhắc đến Java người ta không chỉ nhắc đến java như một ngôn ngữ lập trình mà nhắc đến như là một công nghệ hay một nền tảng phát triển.

Cũng như nhiều trường đại học khac, trường đại học Thông Tin Liên Lạc cũng đã đưa Java vào chương trình giảng dạy cho sinh viên. Bài báo cáo này nhằm mục đích giúp sinh viên tìm hiểu vầ bước đầu làm quen với lập trình Java, lập trình ứng dụng Java đơn giản có mục đích và tính hữu dụng trong thực tế.

***Phần II.. Giới thiệu về bài báo cáo : mục đích cụ thể cuả ứng dụng game con rắn.***

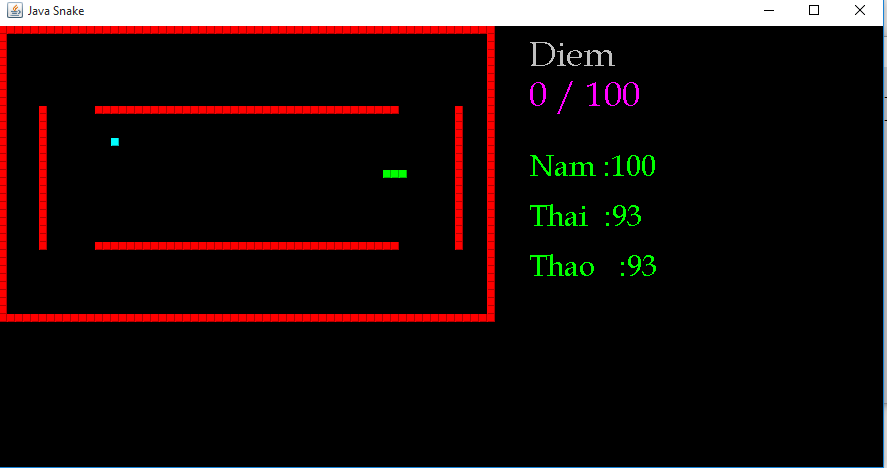
Thông tin:

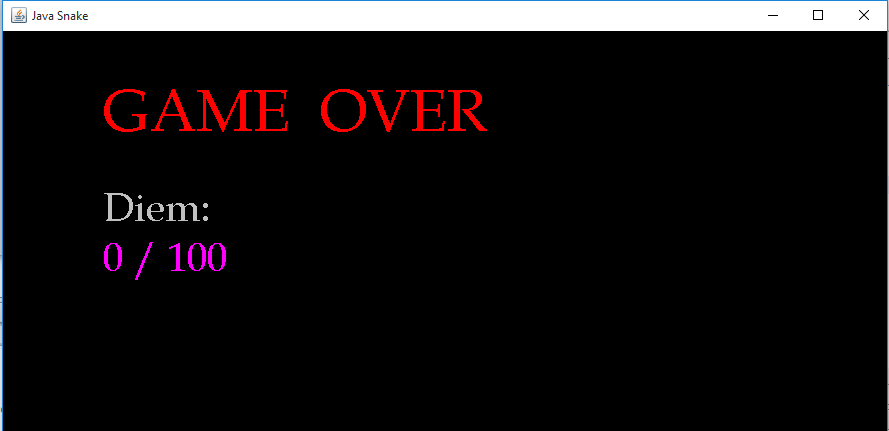
- Ứng dụng game con rắn được xây dựng nhằm mục đích làm quen với ngôn ngữ, các câu lệnh cơ bản và một số công nghệ hướng đối tượng tiên tiến.

- Ứng dụng còn có mục đích chính nhằm giải trí nhẹ nhành cho những ai đang căng thẳng.

**Một số hình ảnh của ưng dụng game con rắn:**







***Phần III.. Các công nghệ hướng đối tượng được sử dụng trong ứng dụng***

**1.. Kế thừa**

Tính kế thừa trong Java là một kỹ thuật mà trong đó một đối tượng thu được tát cả các thuộc tính và hành vi của đối tượng cha. Ý tưởng đằng sau kế thừa trong Java là bạn có thể tạo các lớp mới mà được xây dựng dựa trên các lớp đang tồn tại. Khi bạn kế thừa từ một lớp đang tồn tại, bạn có thể tái sử dụng các phương thức và các trường của lớp cha và bạn cũng có thể bổ sung them các phương thức và các trường khác. Tính kế thừa biểu diễn mối quan hệ IS-A, còn được gọi là quan hệ cha-con.

Khi chúng ta nói về kế thừa , từ khóa thường xuyên nhất được sử dụng là **extends** và **implements**. Những từ này có thể định nghiã một kiểu loại IS-A của loại khác. Sự dụng những từ này chúng ta có thể tạo ra một đối tượng thuộc tính của đối tượng khác. Chúng ta sử dụng từ khóa extends của lớp con để có kế thừa các thuộc tính của lớp cha trừ các thuộc tính private của lớp cha.

Cú pháp kế thừa trong Java:

Class ten\_lop\_con extends ten\_lop\_cha

{

// Các phương thức và các trường

}

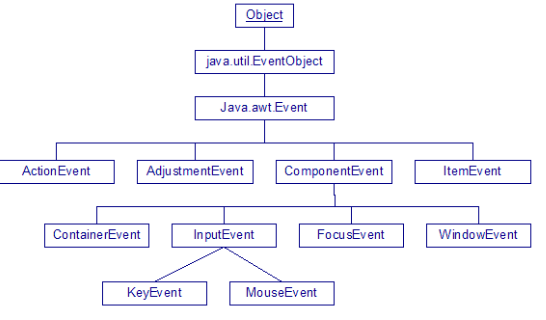
**2.. Bắt sự kiện trong Java**

Các hệ thống GUI xử lí tương tác với người dung với dự hỗ trợ của cá mô hình hướng sự kiện (event\_driven)

Sự kiện trong Java:

+ Sự kiện cso thể được định nghĩa như là một kiểu tín hiệu tới chương trình thông báo một việc gì đó xảy ra.

+ Sự kiện được sinh ra các hành động của người dung, ví dụ như di chuyển hay nhấn phím.



**Các phân cấp các lớp của gói Event**

**3.. Giao diện tương tác với người dung**

Các ứng dụng phần mềm hiện nay rất thân thiện vì được trình bày nhiều màn hình giao diện đồ họa đẹp mắt. Các ngôn ngữ lập trình hiện nay cung cấp các đối tượng đồ họa chúng có thể được điều khiển bởi người lập trình hay người sử dụng. Một trong những kết quả quan trọng đó là ngôn ngữ hiện nay được dựa trên Giao diện dung đồ hạo Graphical User Interface – GUI.

***Phần IV.. Soure code***

*package gameconran;*

*import java.awt.Color;*

*import java.awt.Font;*

*import java.awt.Graphics;*

*import java.util.Random;*

*import javax.swing.JPanel;*

*/\*\**

*\**

*\* @author NAM*

*\*/*

*public class MySnakePanel extends JPanel{*

*public MySnakePanel(int map){*

*System.out.println(map);*

*food = false;*

*foodColor = false;*

*isLose = false;*

*scores = 0;*

*this.setBackground(Color.black);*

*matrix = new int[MyStaticVar.Y/MyStaticVar.sizeCell][MyStaticVar.X/MyStaticVar.sizeCell];*

*for(int i = 0; i < matrix.length; i++)*

*{*

*for(int j = 0; j < matrix[i].length; j++)*

*{*

*if(i == 0) {*

*matrix[i][j] = 1;*

*}*

*else if(i == matrix.length - 1) {*

*matrix[i][j] = 1;*

*}*

*else if(j == 0) {*

*matrix[i][j] = 1;*

*}*

*else if(j == matrix[i].length - 1) {*

*matrix[i][j] = 1;*

*}*

*else {*

*matrix[i][j] = 0;*

*}*

*}*

*}*

*if(map == 1){*

*// int a = 1;*

*int b = matrix.length - 20;;*

*for(int j = 11; j < matrix[0].length - 11; j++){*

*// matrix[a][j] = 0;*

*matrix[b][j] = 1;*

*}*

*}*

*else if(map == 2){*

*int a = 10;*

*int b = matrix.length - 10;;*

*for(int j = 12; j < matrix[0].length - 12; j++){*

*matrix[a][j] = 1;*

*matrix[b][j] = 1;*

*}*

*for(int i = a; i <= b; i++){*

*for(int j = 5; j < 6; j++){*

*matrix[i][matrix[0].length - 5] = 1;*

*matrix[i][j] = 1;*

*}*

*}*

*}*

*sn = new Snake();*

*for(int i = 0; i < sn.getLength(); i++){*

*matrix[sn.getPoint(i).getX()][sn.getPoint(i).getY()] = 2;*

*}*

*d = 1;*

*}*

*public void setMatrix(int i, int j, int num){*

*matrix[i][j] = num;*

*}*

*public int getMatrix(int i, int j){*

*return matrix[i][j];*

*}*

*public void setD(int d){*

*this.d = d;*

*}*

*public int getD(){*

*return this.d;*

*}*

*@Override*

*@SuppressWarnings("empty-statement")*

*protected void paintComponent(Graphics g){*

*super.paintComponent(g);*

*if(scores == 100){*

*isLose = true;*

*}*

*if(!isLose){*

*Point p = new Point(0, 0);*

*for(int i = sn.getLength(); i > 0; i--){*

*// point thứ i bằng point thứ i-1*

*p = sn.getPoint(i-1);*

*sn.setPoint(i, p);*

*matrix[sn.getPoint(i).getX()][sn.getPoint(i).getY()] = 2;*

*}*

*// xét point đầu tiên của snake*

*if(d == 1){*

*p.setX(sn.getPoint(0).getX());*

*p.setY(sn.getPoint(0).getY() + 1);*

*sn.setPoint(0, p);*

*}*

*else if(d == 2){*

*p.setX(sn.getPoint(0).getX());*

*p.setY(sn.getPoint(0).getY() - 1);*

*sn.setPoint(0, p);*

*}*

*else if(d == 3){*

*p.setX(sn.getPoint(0).getX() + 1);*

*p.setY(sn.getPoint(0).getY());*

*sn.setPoint(0, p);*

*}*

*else if(d == 4){*

*p.setX(sn.getPoint(0).getX() - 1);*

*p.setY(sn.getPoint(0).getY());*

*sn.setPoint(0, p);*

*}*

*if(matrix[sn.getPoint(0).getX()][sn.getPoint(0).getY()] == 3){*

*// nếu vị trí đầu của snake là thức ăn*

*sn.setLength(sn.getLength() + 1); // tăng chiều dài snake lên 1*

*matrix[sn.getPoint(0).getX()][sn.getPoint(0).getY()] = 2;*

*food = false;*

*scores++;*

*}*

*else if(matrix[sn.getPoint(0).getX()][sn.getPoint(0).getY()] == 0){*

*// nếu là 0 thì cho phép di, xóa vị trí cuối của snake*

*matrix[sn.getPoint(sn.getLength()).getX()][sn.getPoint(sn.getLength()).getY()] = 0;*

*}*

*else if(matrix[sn.getPoint(0).getX()][sn.getPoint(0).getY()] == 2 || matrix[sn.getPoint(0).getX()][sn.getPoint(0).getY()] == 1){*

*isLose = true;*

*}*

*matrix[sn.getPoint(0).getX()][sn.getPoint(0).getY()] = 2; // vị trí đầu tiên của snake*

*if(!food){*

*// tạo vị trí ngẫu nhiên của thức ăn*

*Random rd = new Random();*

*int randomX = rd.nextInt(matrix.length);*

*int randomY = rd.nextInt(matrix[0].length);*

*if(matrix[randomX][randomY] != 0){*

*outerloop:*

*for(int i = 0; i < matrix.length; i++){*

*for(int j = 0; j < matrix[i].length; j++){*

*if(matrix[i][j] == 0){*

*matrix[i][j] = 3;*

*break outerloop;*

*}*

*}*

*}*

*}*

*else{*

*matrix[randomX][randomY] = 3;*

*}*

*food = true;*

*}*

*// vẽ trên panel dựa trên ma trận*

*for(int i = 0; i < matrix.length; i++){*

*for(int j = 0; j < matrix[i].length; j++){*

*if(matrix[i][j] == 1){*

*g.setColor(Color.red);*

*g.fill3DRect(j \* MyStaticVar.sizeCell, i \* MyStaticVar.sizeCell, MyStaticVar.sizeCell, MyStaticVar.sizeCell, true);*

*}*

*else if(matrix[i][j] == 2){*

*g.setColor(Color.GREEN);*

*g.fill3DRect(j \* MyStaticVar.sizeCell, i \* MyStaticVar.sizeCell, MyStaticVar.sizeCell, MyStaticVar.sizeCell, true);*

*}*

*else if(matrix[i][j] == 3){*

*if(foodColor){*

*g.setColor(Color.CYAN);*

*}*

*else{*

*g.setColor(Color.blue);*

*}*

*foodColor = !foodColor;*

*g.fill3DRect(j \* MyStaticVar.sizeCell, i \* MyStaticVar.sizeCell, MyStaticVar.sizeCell, MyStaticVar.sizeCell, true);*

*}*

*}*

*}*

*g.setColor(Color.LIGHT\_GRAY);*

*Font font = new Font("Book Antiqua", Font.PLAIN, 35);*

*g.setFont(font);*

*char [] data = {'D', 'i', 'e', 'm'};*

*g.drawChars(data, 0, data.length, MyStaticVar.X + 30, 40);*

*String s = scores + " / " + MyStaticVar.maxLengthSnake;*

*data = s.toCharArray();*

*g.setColor(Color.MAGENTA);*

*g.drawChars(data, 0, data.length, MyStaticVar.X + 30, 80);*

*g.setColor(Color.green);*

*Font f = new Font("Book Antiqua", Font.PLAIN, 30);*

*g.setFont(f);*

*char [] a = {'N', 'a', 'm', ' ',':','1','0','0'};*

*g.drawChars(a, 0, a.length, MyStaticVar.X + 30, 150);*

*g.setColor(Color.green);*

*Font fa = new Font("Book Antiqua", Font.PLAIN, 30);*

*g.setFont(f);*

*char [] b = {'T', 'h', 'a', 'i', ' ', ' ',':','9','3'};*

*g.drawChars(b, 0, b.length, MyStaticVar.X + 30, 200);*

*g.setColor(Color.green);*

*Font fb = new Font("Book Antiqua", Font.PLAIN, 30);*

*g.setFont(f);*

*char [] c = {'T', 'h', 'a','o', ' ', ' ', ' ',':','9','3',};*

*g.drawChars(c, 0, c.length, MyStaticVar.X + 30, 250);*

*}*

*else{*

*g.setColor(Color.red);*

*Font font = new Font("Book Antiqua", Font.PLAIN, 60);*

*g.setFont(font);*

*char [] data = {'G', 'A', 'M', 'E', ' ', ' ', 'O', 'V', 'E', 'R'};*

*g.drawChars(data, 0, data.length, 100, 100);*

*g.setColor(Color.LIGHT\_GRAY);*

*font = new Font("Book Antiqua", Font.PLAIN, 40);*

*g.setFont(font);*

*String s = "Diem: ";*

*data = s.toCharArray();*

*g.drawChars(data, 0, data.length, 100, 190);*

*s = scores + " / " + MyStaticVar.maxLengthSnake;*

*data = s.toCharArray();*

*g.setColor(Color.MAGENTA);*

*g.drawChars(data, 0, data.length, 100, 240);*

*}*

*}*

*private int matrix[][];*

*private int d; // 1 là từ trái qua phải, 2 là từ phải qua trái, 3 là từ trên xuống dưới, 4 là từ dưới lên trên*

*private Snake sn;*

*private boolean food; // true là có thức ăn, false là không có*

*private boolean foodColor; // dùng để thay đổi màu sắc cho thức ăn*

*private boolean isLose; // true là thua*

*private int scores;*

*}*