**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**--◄۞►--**

**Logo

Description automatically generated**

**BÁO CÁO BÀI TẬP LỚN LỚP CÔNG NGHỆ JAVA-1-1-22 (N08)**

**Sinh viên thực hiện : Nguyễn Hữu Minh Đức**

**Lớp : CNTT6**

**Mã sinh viên : 211214289**

Mục lục

[**1. Các chức năng trong game:** 1](#_Toc134044567)

[**2. Các class chính trong chương trình:** 1](#_Toc134044568)

[**3. Một số hình ảnh của game:** 11](#_Toc134044569)

[**4. Tài liệu tham khảo:** 12](#_Toc134044570)

# **1. Các chức năng trong game:**

* Rắn sẽ di chuyển theo hướng được nhập từ bàn phím, và sẽ dài ra khi ăn được táo
* Rắn sẽ chết nếu như tự cắn phải chính mình hoặc đâm vào tường
* Điểm của người chơi sẽ được cập nhật liên tục

# **2. Các class chính trong chương trình:**

* GamePanel
* GameFrame
* SnakeGameUTube

Class SnakeGameUTube bao gồm hàm main.

package snakeUtube;

public class SnakeGameUTube {

public static void main(String[] args) {

new GameFrame();

}

}

Class GameFrame để tạo khung hình cho game.

package snakeUtube;

import javax.swing.JFrame;

import javax.swing.\*;

import java.awt.BorderLayout;

import java.awt.event.\*;

public class GameFrame extends JFrame {

GameFrame(){

this.add(new GamePanel());

this.setTitle("Snake");

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

this.setResizable(false);

this.pack();

this.setVisible(true);

this.setLocationRelativeTo(null);

}

}

Class GamePanel chứa các phương thức quan trọng trong game. Rắn sẽ được vẽ theo 2 phần là phần đầu và phần thân, tôi sử dụng graphics để lưu chữ rắn và mỗi khi rắn cơ thể của nó sẽ tăng thêm 1 đơn vị. Phương thức newApple() sẽ sinh ra táo ở 1 vị trí bất kỳ. Phương thức checkCollitions() sẽ kiểm tra xem rắn có cắn chính mình hay va vào các chướng ngại vật hay không nếu có sẽ gọi đến phương thức gameOver().

package snakeUtube;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.util.ArrayList;

import java.util.Collections;

import java.util.List;

import java.util.Random;

import java.io.\*;

public class GamePanel extends JPanel implements ActionListener{

static final int SCREEN\_WIDTH = 1300;

static final int SCREEN\_HEIGHT = 750;

static final int UNIT\_SIZE = 50;

static final int GAME\_UNITS = (SCREEN\_WIDTH\*SCREEN\_HEIGHT)/(UNIT\_SIZE\*UNIT\_SIZE);

static int DELAY = 50;

final int x[] = new int[GAME\_UNITS];

final int y[] = new int[GAME\_UNITS];

int bodyParts = 6;

int applesEaten;

int appleX;

int appleY;

char direction = 'R';

boolean running = false;

Timer timer;

Random random;

GamePanel(){

random = new Random();

this.setPreferredSize(new Dimension(SCREEN\_WIDTH,SCREEN\_HEIGHT));

this.setBackground(Color.black);

this.setFocusable(true);

this.addKeyListener(new MyKeyAdapter());

startGame();

}

public void startGame() {

newApple();

running = true;

timer = new Timer(DELAY,this);

timer.start();

}

public void paintComponent(Graphics g) {

super.paintComponent(g);

draw(g);

}

public void draw(Graphics g) {

if(running) {

g.setColor(Color.red);

g.fillOval(appleX, appleY, UNIT\_SIZE, UNIT\_SIZE);

for(int i = 0; i< bodyParts;i++) {

if(i == 0) {

g.setColor(Color.green);

g.fillRect(x[i], y[i], UNIT\_SIZE, UNIT\_SIZE);

}

else {

g.setColor(new Color(45,180,0));

//g.setColor(new Color(random.nextInt(255),random.nextInt(255),random.nextInt(255)));

g.fillRect(x[i], y[i], UNIT\_SIZE, UNIT\_SIZE);

}

}

g.setColor(Color.red);

g.setFont( new Font("Ink Free",Font.BOLD, 40));

FontMetrics metrics = getFontMetrics(g.getFont());

g.drawString("Score: "+applesEaten, (SCREEN\_WIDTH - metrics.stringWidth("Score: "+applesEaten))/2, g.getFont().getSize());

}

else {

gameOver(g);

}

}

public void newApple(){

appleX = random.nextInt((int)(SCREEN\_WIDTH/UNIT\_SIZE))\*UNIT\_SIZE;

appleY = random.nextInt((int)(SCREEN\_HEIGHT/UNIT\_SIZE))\*UNIT\_SIZE;

}

public void move(){

for(int i = bodyParts;i>0;i--) {

x[i] = x[i-1];

y[i] = y[i-1];

}

switch(direction) {

case 'U':

y[0] = y[0] - UNIT\_SIZE;

break;

case 'D':

y[0] = y[0] + UNIT\_SIZE;

break;

case 'L':

x[0] = x[0] - UNIT\_SIZE;

break;

case 'R':

x[0] = x[0] + UNIT\_SIZE;

break;

}

}

public void checkApple() {

if((x[0] == appleX) && (y[0] == appleY)) {

bodyParts++;

applesEaten++;

newApple();

}

}

public void checkCollisions() {

//checks if head collides with body

for(int i = bodyParts;i>0;i--) {

if((x[0] == x[i])&& (y[0] == y[i])) {

running = false;

}

}

//check if head touches left border

if(x[0] < 0) {

running = false;

}

//check if head touches right border

if(x[0] > SCREEN\_WIDTH) {

running = false;

}

//check if head touches top border

if(y[0] < 0) {

running = false;

}

//check if head touches bottom border

if(y[0] > SCREEN\_HEIGHT) {

running = false;

}

if(!running) {

timer.stop();

}

}

public void gameOver(Graphics g) {

//Score

g.setColor(Color.red);

g.setFont( new Font("Ink Free",Font.BOLD, 40));

FontMetrics metrics1 = getFontMetrics(g.getFont());

g.drawString("Score: "+applesEaten, (SCREEN\_WIDTH - metrics1.stringWidth("Score: "+applesEaten))/2, g.getFont().getSize());

//Game Over text

g.setColor(Color.red);

g.setFont( new Font("Ink Free",Font.BOLD, 75));

FontMetrics metrics2 = getFontMetrics(g.getFont());

g.drawString("Game Over", (SCREEN\_WIDTH - metrics2.stringWidth("Game Over"))/2, SCREEN\_HEIGHT/2);

}

@Override

public void actionPerformed(ActionEvent e) {

if(running) {

move();

checkApple();

checkCollisions();

}

repaint();

}

public class MyKeyAdapter extends KeyAdapter{

@Override

public void keyPressed(KeyEvent e) {

switch(e.getKeyCode()) {

case KeyEvent.VK\_LEFT:

if(direction != 'R') {

direction = 'L';

}

break;

case KeyEvent.VK\_RIGHT:

if(direction != 'L') {

direction = 'R';

}

break;

case KeyEvent.VK\_UP:

if(direction != 'D') {

direction = 'U';

}

break;

case KeyEvent.VK\_DOWN:

if(direction != 'U') {

direction = 'D';

}

break;

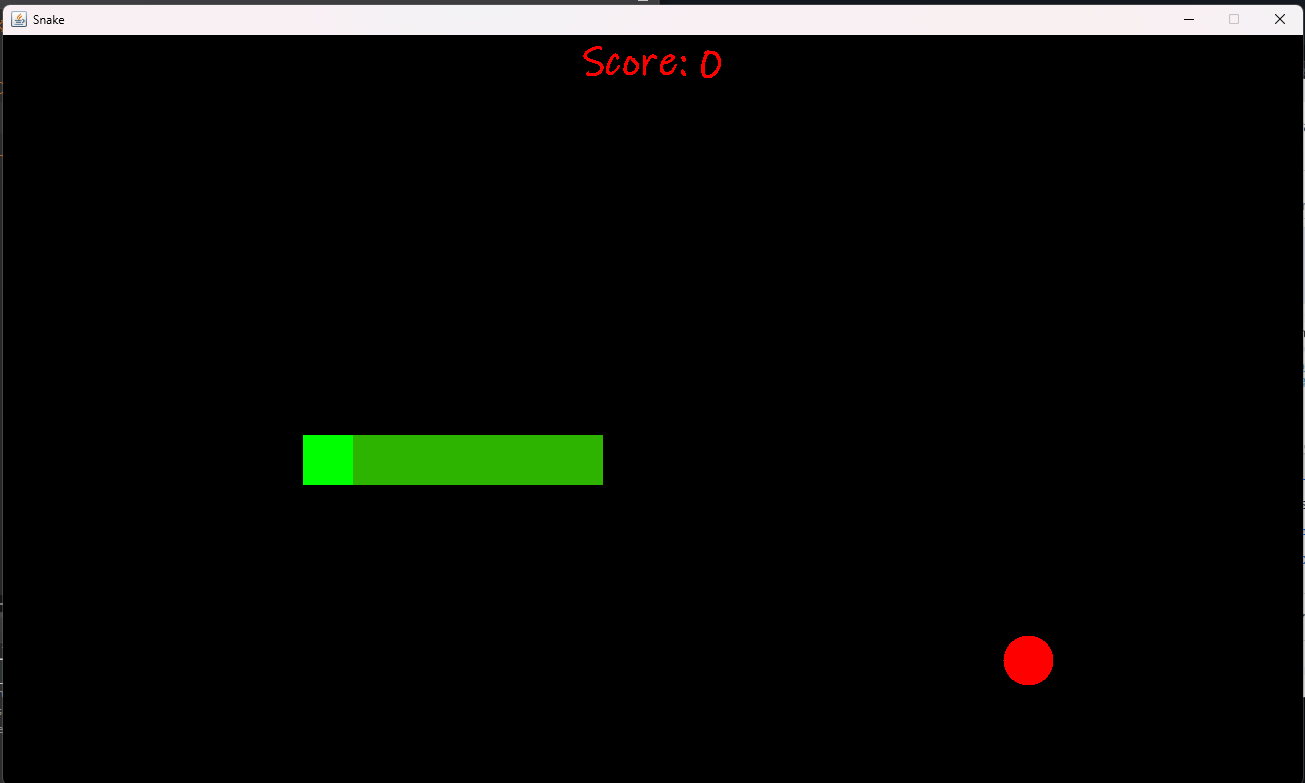
}

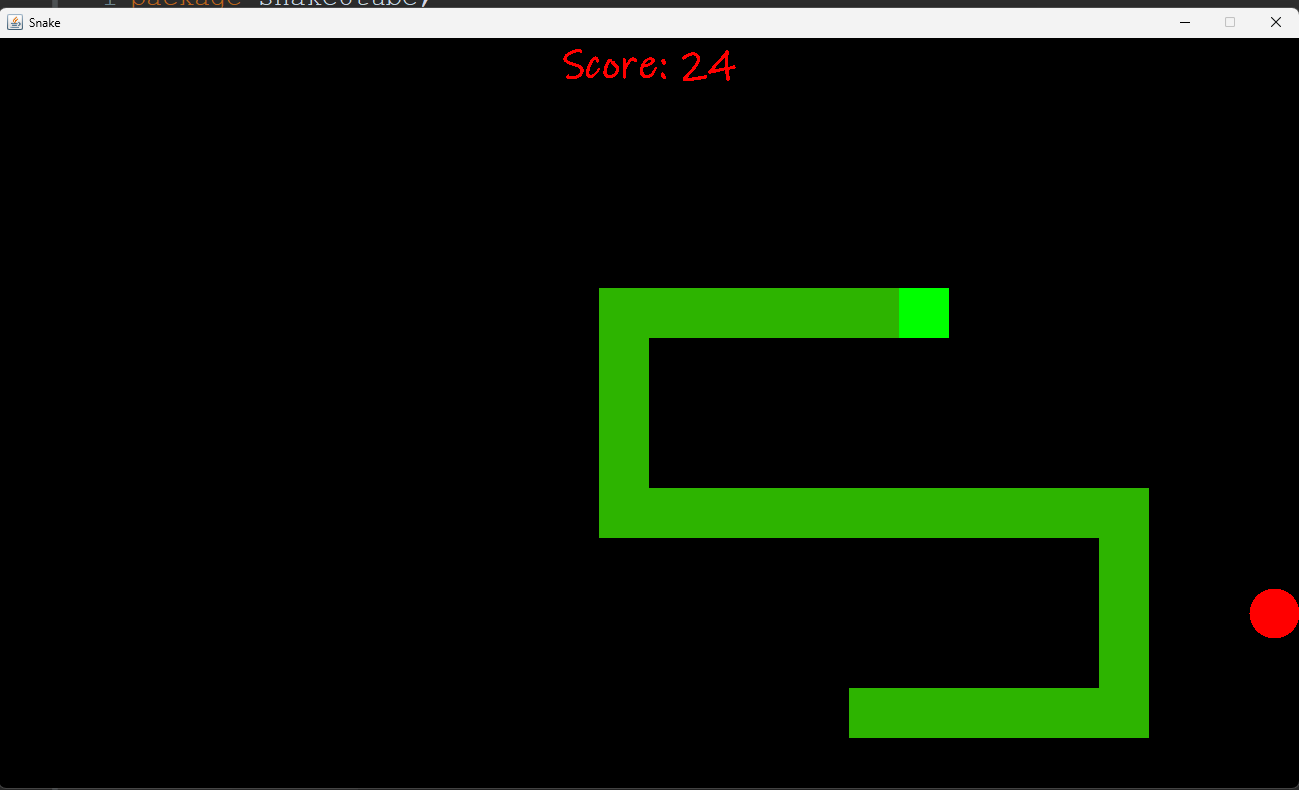
}

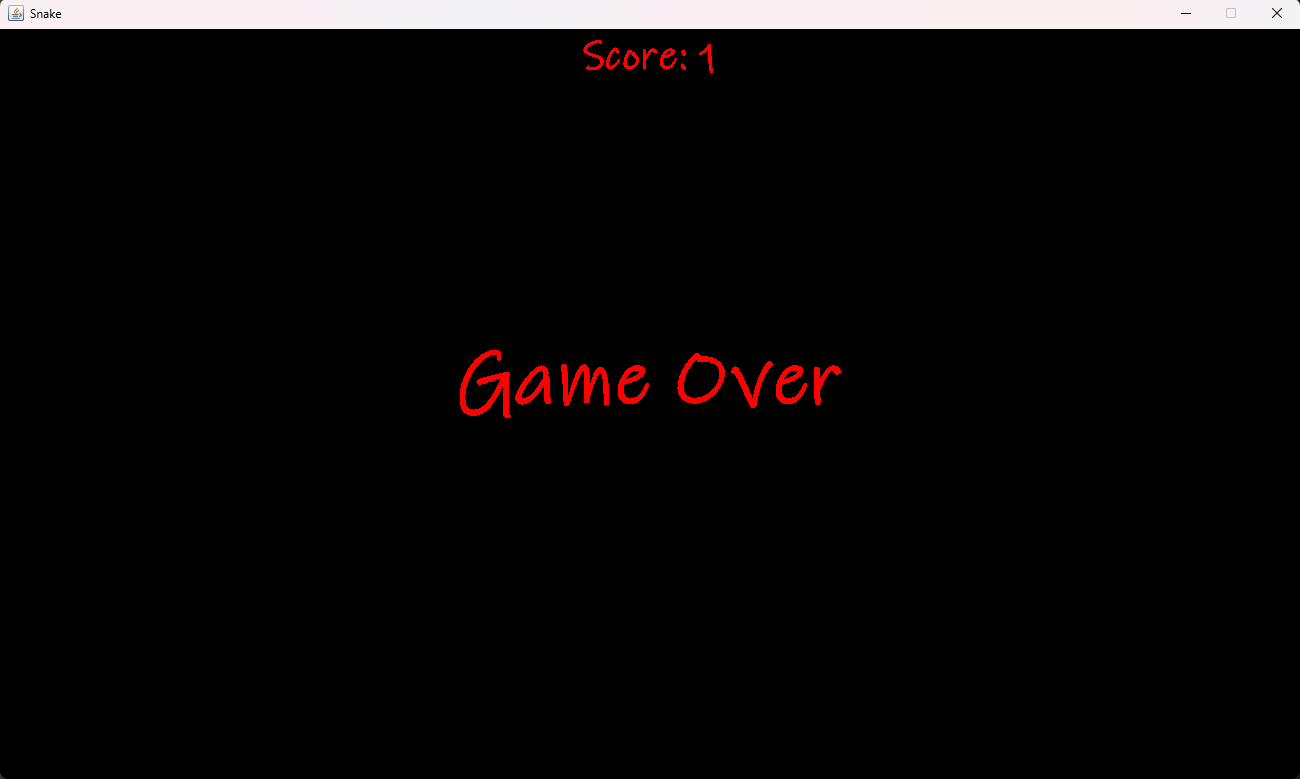
}

}

# **3. Một số hình ảnh của game:**







# **4. Tài liệu tham khảo:**

[Design Snake Game - GeeksforGeeks](https://www.geeksforgeeks.org/design-snake-game/)

https://youtu.be/bI6e6qjJ8JQd