TRƯỜNG ĐẠI HỌC NHA TRANG **KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO THỰC TẬP CƠ SỞ**

**SỬ DỤNG CÔNG CỤ VISUAL STUDIO TRONG JAVA MÔ PHỎNG GAME RẮN SĂN MỒI**

**GVHD : Nguyễn Huỳnh Huy**

**SVTH : Nguyễn Tuấn Tú**

**MSSV : 63135967**

**Lớp : 63.CNTT-5**

Khánh Hòa – 2023

TRƯỜNG ĐẠI HỌC NHA TRANG **KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO THỰC TẬP CƠ SỞ**

**SỬ DỤNG CÔNG CỤ VISUAL STUDIO TRONG JAVA MÔ PHỎNG GAME RẮN SĂN MỒI**

GVHD : Nguyễn Huỳnh Huy

SVTH : Nguyễn Tuấn Tú

MSSV : 63135967

Lớp : 63.CNTT-5

Khánh Hòa - 2023

# LỜI CẢM ƠN

Trước tiên, trong quá trình thực hiện đề tài, em xin chân thành cảm ơn đến giảng viên Nguyễn Huỳnh Huy cùng với sự hỗ trợ của các giảng viên trong khoa đã tạo điều kiện hỗ trợ và giúp đỡ em trong quá trình học tập và thực hiện đề tài này. Nhờ đó, em đã hoàn thành được bài báo cáo. Qua quá trình này, em đã có cơ hội nắm vững các phương pháp thuật toán và lập trình hướng đối tượng trên Java, cũng như hiểu rõ quy trình phát triển trò chơi.

Dù đã cố gắng tìm hiểu và thực hiện, em nhận thấy rằng đề tài vẫn còn một số sai sót và chưa hoàn thiện đầy đủ các chức năng. Vì vậy, em rất mong nhận được sự đóng góp ý kiến từ các giảng viên để hoàn thiện đề tài này.

Em xin chân thành cảm ơn sự hỗ trợ và đóng góp của quý thầy cô!

# MỤC LỤC

[LỜI CẢM ƠN…………………………………………………………………………i](#_Toc155349238)

[MỤC LỤC ii](#_Toc155349239)

[DANH MỤC HÌNH iv](#_Toc155349240)

[TÓM TẮT v](#_Toc155349241)

[CHƯƠNG 1: GIỚI THIỆU 1](#_Toc155349242)

[1.1. Tổng quan 1](#_Toc155349243)

[1.2. Ngôn ngữ Java 2](#_Toc155349244)

[1.2.1. Java là gì? 2](#_Toc155349245)

[1.2.2. Ưu điểm ngôn ngữ Java 2](#_Toc155349246)

[1.3. Cài đặt công cụ cần thiết 2](#_Toc155349247)

[1.3.1. Visual Studio Code 2](#_Toc155349248)

[1.3.2. Java 3](#_Toc155349249)

[1.4. Bố cục của trò chơi rắn săn mồi 3](#_Toc155349250)

[1.5. Danh sách các thành phần xây dựng trò chơi rắn săn mồi 5](#_Toc155349251)

[1.5.1. Các thông số sử dụng trong game 5](#_Toc155349252)

[1.5.2. Danh sách các sự kiện trong game 5](#_Toc155349253)

[CHƯƠNG 2: PHƯƠNG PHÁP NGHIÊN CỨU 7](#_Toc155349254)

[2.1. Cách thức rắn di chuyển 7](#_Toc155349255)

[2.2. Cách thức lưu và tải điểm cao nhất 7](#_Toc155349256)

[2.2.1. Lưu điểm cao nhất 7](#_Toc155349257)

[2.2.2. Tải điểm cao nhất 8](#_Toc155349258)

[CHƯƠNG 3: KẾT QUẢ THỰC HIỆN 9](#_Toc155349259)

[3.1. Giao diện menu 9](#_Toc155349260)

[3.2. Giao diện game play 10](#_Toc155349261)

[CHƯƠNG 4: TỔNG KẾT 12](#_Toc155349262)

[TÀI LIỆU THAM KHẢO 13](#_Toc155349263)

# DANH MỤC HÌNH

[Hình 1.1 Trò chơi Snake trên Nokia 1](#_Toc155351115)

[Hình 1.2 Giao diện demo game Snake 2](#_Toc155351116)

[Hình 1.3 Hướng dẫn tải VS Code 3](#_Toc155351117)

[Hình 1.4 Hướng dẫn tải Java 3](#_Toc155351118)

[Hình 1.5 Kiểm tra phiên bản 3](#_Toc155351119)

[Hình 1.6 Bố cục game 4](#_Toc155351120)

[Hình 2.1 Cách thức rắn di chuyển 8](#_Toc155178692)

[Hình 2.2 Cách thức lưu điểm cao nhất 9](#_Toc155178693)

[Hình 2.3 Cách thức tải điểm cao nhất 9](#_Toc155178694)

[Hình 3.1 Giao diện nhập tên 10](#_Toc155178620)

[Hình 3.2 Bảng xếp hạng sau khi nhập tên 10](#_Toc155178621)

[Hình 3.3 Giao diện khi chơi 11](#_Toc155178622)

[Hình 3.4 Giao diện khi kết thúc trò chơi 11](#_Toc155178623)

[Hình 3.5 BXH được cập nhật sau khi chơi 12](#_Toc155178624)

# TÓM TẮT

Trải qua những thập kỷ phát triển công nghệ, cuộc sống con người đã được nâng cao đáng kể, và giới giải trí không phải là ngoại lệ. Cùng với sự tiến bộ của công nghệ, những trải nghiệm giải trí đa dạng đã xuất hiện, bao gồm xem phim, nghe nhạc và không thể thiếu những trò chơi số.

Ngành công nghiệp game đã có một hành trình phát triển dài, bắt đầu từ sáng chế "Thiết bị giải trí bằng ống phóng tia âm cực" (CRT tivi) vào cuối những năm 1970. Đó là bước khởi đầu của một ngành công nghiệp giải trí mới, với sự xuất hiện của nhiều trò chơi hấp dẫn và đồ họa đỉnh cao để đáp ứng nhu cầu giải trí trong thời đại công nghệ số.

Vào năm 1997, thiết bị di động đình đám Nokia 3210 đã trở thành huyền thoại nhờ giá cả và tính năng tiên tiến của nó. Trong thời kỳ này, trò chơi cổ điển "rắn săn mồi" đã chinh phục người chơi với hình ảnh những ô vuông liên tiếp di chuyển trên màn hình màu xanh. Mặc dù đơn giản nhưng trò chơi này đã góp phần xây dựng danh tiếng và thành công cho ngành công nghiệp game.

Cùng với đam mê lập trình, em quyết định phát triển một phiên bản mới của trò chơi "rắn săn mồi" bằng cách sử dụng ngôn ngữ lập trình Java trên nền tảng PC. Chương trình sẽ được xây dựng bằng công cụ Visual Studio Code kèm theo những cải tiến phù hợp với xu hướng và đòi hỏi của thị trường ngày nay.

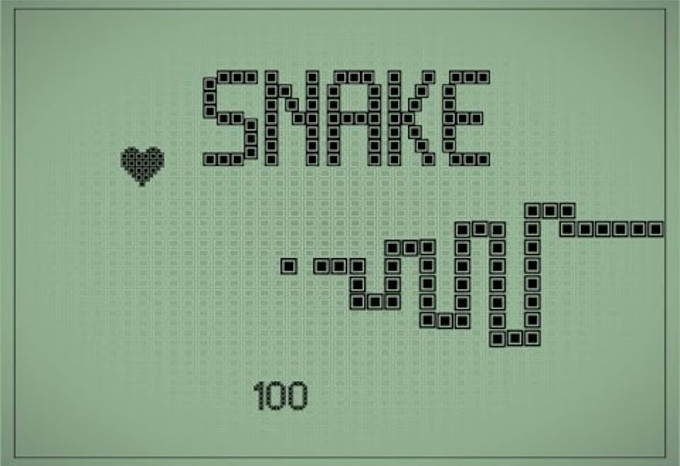
Toàn bộ sản phẩm được đóng gói, mã nguồn được trình bày trong phụ lục và được upload theo địa chỉ: [ngtuantu1202/ThucTapCoSo: code game (github.com)](https://github.com/ngtuantu1202/ThucTapCoSo)

# CHƯƠNG 1: GIỚI THIỆU

## 1.1. Tổng quan

Trò chơi "Con rắn săn mồi" (Snake) xuất hiện lần đầu vào cuối những năm 1970 với sự phát triển của ngành công nghiệp game. Trò chơi đầu tiên được chơi trên một máy arcade và có tên gọi là "Blockade." Nó được tạo ra bởi nhóm nhà phát triển game Atari và trở thành một trong những trò chơi đầu tiên sử dụng màn hình đa người chơi.

Trong thập kỷ sau, trò chơi "Con rắn săn mồi" lan rộng trên nhiều nền tảng khác nhau, bao gồm điện thoại di động, máy tính cá nhân và các thiết bị chơi game cầm tay. Sự đơn giản và tính gây nghiện của trò chơi đã giữ cho nó tồn tại và trở thành một biểu tượng trong thế giới game.



**Hình 1.1 Trò chơi Snake trên Nokia**

Bài báo cáo này tập trung vào việc xây dựng trò chơi "Snake" bằng ngôn ngữ lập trình Java và các phương pháp giải quyết các vấn đề trong quá trình phát triển trò chơi. Nó cung cấp các bước thực hiện chi tiết và cách tiếp cận để giải quyết các vấn đề, cùng với những kết quả đã đạt được dựa trên một số ví dụ thử nghiệm trong quá trình lập trình.

Trò chơi "Snake" sẽ có các tính năng như điều khiển con rắn bằng các phím mũi tên hoặc các phím tương tự, cập nhật điểm số, kiểm tra va chạm và xử lý kết thúc trò chơi khi con rắn va chạm. Ngoài ra, có thể thêm các tính năng bổ sung như mức độ khó, âm thanh và giao diện người dùng đẹp mắt để tăng thêm sự thú vị cho trò chơi.

A screenshot of a computer

Description automatically generated

**Hình 1.2 Giao diện demo game Snake**

## 1.2. Ngôn ngữ Java

### 1.2.1. Java là gì?

Java là một ngôn ngữ lập trình phổ biến, được tạo ra vào năm 1995. Java thuộc sở hữu của Oracle, và hơn 3 tỷ thiết bị chạy Java. Nó được sử dụng cho nhiều mục đích như ứng dụng di động (đặc biệt là ứng dụng Android), ứng dụng desktop, ứng dụng web, máy chủ web và máy chủ ứng dụng, trò chơi, kết nối cơ sở dữ liệu, và rất nhiều ứng dụng khác nữa! Java đã trở thành một trong những công cụ quan trọng nhất trong lĩnh vực phát triển phần mềm, đóng vai trò quan trọng trong việc xây dựng và triển khai nhiều loại ứng dụng khác nhau trên các nền tảng khác nhau.

### 1.2.2. Ưu điểm ngôn ngữ Java

* Java hoạt động trên nhiều nền tảng khác nhau (Windows, Mac, Linux, Raspberry Pi, v.v.).
* Nó là một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới.
* Java có nhu cầu lớn trên thị trường lao động hiện nay.
* Nó dễ học và sử dụng, phù hợp cho cả người mới học lập trình.
* Java là mã nguồn mở và miễn phí.
* Nó đảm bảo tính bảo mật, tốc độ và sức mạnh.
* Java có sự hỗ trợ mạnh mẽ từ cộng đồng (tens of millions of developers).
* Java là một ngôn ngữ hướng đối tượng, mang lại cấu trúc rõ ràng cho các chương trình và cho phép việc sử dụng mã nguồn lại, giảm chi phí phát triển.
* Do Java gần gũi với C++ và C#, nó giúp các lập trình viên dễ dàng chuyển đổi giữa Java và các ngôn ngữ khác hoặc ngược lại.

## 1.3. Cài đặt công cụ cần thiết

### 1.3.1. Visual Studio Code

B1: Truy cập trang chủ và tải về Visual Studio Code ([Visual Studio Code - Code Editing. Redefined](https://code.visualstudio.com/)).

A screenshot of a computer program

Description automatically generated

**Hình 1.3 Hướng dẫn tải VS Code**

B2: Chấp nhận điều khoảng sử dụng và bấm Next.

B3: Chọn thư mục và tuỳ chọn rồi Install.

### 1.3.2 Java

B1: Truy cập trang chủ và tải về Java ([Getting Started with Java in Visual Studio Code](https://code.visualstudio.com/docs/java/java-tutorial)).

**A purple rectangle with white text

Description automatically generated**

**Hình 1.4 Hướng dẫn tải Java**

B2: Chấp nhận điều khoảng sử dụng và bấm Next.

B3: Install.

B4: Kiểm tra bằng cmd.

A computer screen with white text

Description automatically generated

**Hình 1.5 Kiểm tra phiên bản**

## 1.4. Bố cục của trò chơi rắn săn mồi

Trò chơi được hiển thị trên màn hình, các vị trí vật thể được thể hiện dưới dạng trục (X,Y) như sau:

A white rectangle with black lines and numbers

Description automatically generated

**Hình 1.6 Bố cục game**

Một Panel chính (''Game'' class):

* Nơi hiển thị chính cho việc chơi trò chơi rắn săn mồi.
* Được thêm vào ''JFrame'' trong lớp ''App'' thông qua ''frame.add(game); ''.
* Sử dụng ''Timer'' để triển khai game loop và cập nhật trạng thái của trò chơi.
* Xử lý sự kiện từ bàn phím để điều khiển rắn.

Một Panel phụ (''App'' class):

* Được sử dụng để hiển thị hộp thoại thông báo và nhập tên người chơi.
* Được sử dụng trong phương thức "getPlayerName()" của lớp "App" thông qua "JOptionPane.showInputDialog("Nhập tên của bạn:");".
* Cũng được sử dụng để hiển thị top 3 điểm cao nhất.

Bản thân Panel chính (''Game'' class) chứa các thành phần như:

* Đầu rắn ("snakeHead"): Được vẽ bằng "fill3DRect" trong phương thức "draw".
* Thân rắn ("snakeBody"): Duyệt qua danh sách và vẽ từng phần của thân rắn.
* Thức ăn ("food"): Được vẽ bằng "fill3DRect" trong phương thức "draw".
* Thông tin điểm số, tên người chơi, điểm số cao nhất:
* Được vẽ và hiển thị thông qua "g.drawString" trong phương thức "draw".
* Được hiển thị dựa trên trạng thái của trò chơi (gameOver).

Cấu tạo rắn:

* Con rắn được khởi tạo với diện tích 5\*5px
* Xuất phát điểm rắn ở vị trí tọa độ (5,5)
* Rắn được điều khiển bởi hai tham số velocityX, velocityY trong đó nếu velocityX =1 có nghĩa là rắn đang đi về hướng phải, velocityX =-1 là rắn đang đi về hướng trái và velocityX =0 có nghĩa là hiện tại đang đi thẳng lên hoặc xuống dưới, tương tự với velocityY =1 có nghĩa là rắn đang hướng xuống, velocityY =-1 là rắn đang hướng lên, velocityY =0 là rắn đang hướng phải hoặc hướng trái.
* Rắn có 2 phần: Đầu rắn, thân rắn được tạo nên từ các thân rắn khác nối đuôi nhau tạo thành một thân rắn dài.

Thức ăn:

* Thức ăn được hiển thị ngẫu nhiên trên sau khi load game, không nằm trên người rắn.
* Kích thước thức ăn: 5\*5 px
* Vị trí các thức ăn được đặt lại ngẫu nhiên sau mỗi lần ăn bằng: placefood()

## 1.5. Danh sách các thành phần xây dựng trò chơi rắn săn mồi

### 1.5.1. Các thông số sử dụng trong game

* **boardWidth, boardHeight**: xác định chiều cao và rộng của bảng trò chơi.
* **tileSize**: kích thước mỗi ô trên bảng trò chơi.
* **velocityX**, **velocityY**: thay đổi toạ độ di chuyển của rắn theo trục x và y.
* **gameLoop**: tốc độ di chuyển của rắn.
* **snakeHead**, **snakeBody**: đối tượng đầu rắn và thân rắn.
* **food**, **placeFood**(): đối tượng và phương thức đặt thức ăn.
* **gameOver**: biến đánh dấu trạng thái kết thúc trò chơi.
* **highScore**: Số điểm cao nhất trong trò chơi.
* **playerName**: tên của người chơi.
* **scoreManager**: đối tượng quản lý điểm số, dùng để lưu và tải điểm số cao nhất.
* **startGame**(): phương thức bắt đầu trò chơi, được gọi từ "main()" của "App".
* **move():** phương thức xử lý logic di chuyển của rắn.
* "**showTopScores()":** Phương thức hiển thị top 3 điểm cao nhất.
* **restartGame():** Phương thức khởi động lại trò chơi sau khi kết thúc.
* **updateHighScore():** Phương thức cập nhật điểm số cao nhất sau khi kết thúc trò chơi.

### 1.5.2. Danh sách các sự kiện trong game

Có 2 loại sự kiện chính được xử lý:

**Sự kiện Di chuyển (ActionEvent):**

* Được xử lý bởi phương thức "actionPerformed(ActionEvent e) "xử lý các bước di chuyển, logic kiểm tra va chạm và cập nhật giao diện.
* Được kích hoạt bởi đối tượng "Timer (gameLoop) " với một khoảng thời gian cố định (100 milliseconds).

**Sự kiện Phím (KeyEvent):**

* Được xử lý bởi các phương thức keyPressed(KeyEvent e), keyTyped(KeyEvent e), và keyReleased(KeyEvent e).
* Xác định hành động cụ thể dựa trên phím được nhấn.

**1.5.3. Cách thức hoạt động**

**Khởi Tạo:**

* Khi chương trình bắt đầu, main phương thức trong App.java được gọi.
* Một cửa sổ trò chơi được tạo bằng JFrame.
* Người chơi nhập tên của mình thông qua hộp thoại JOptionPane.
* Một đối tượng Game mới được tạo với kích thước bảng trò chơi và tên người chơi.

**Vòng Lặp Trò Chơi:**

* Mỗi 100 milliseconds, sự kiện ActionEvent được kích hoạt trong Game (qua phương thức actionPerformed).
* Trong mỗi chu kỳ này, move() phương thức được gọi để cập nhật trạng thái của rắn và kiểm tra các điều kiện kết thúc trò chơi.

**Di Chuyển Rắn:**

* Khi rắn ăn một mồi (collision(snakeHead, food)), một phần thân mới được thêm vào cuối rắn và mồi mới được đặt lại ở một vị trí ngẫu nhiên.
* Phần thân của rắn được di chuyển theo đuổi đầu rắn. Các phần thân này được duyệt qua và cập nhật vị trí của mình dựa trên vị trí của phần thân trước đó.

**Kiểm Tra Va Chạm:**

* Khi đầu rắn va chạm với phần thân của mình hoặc chạm vào tường, trò chơi kết thúc (gameOver được đặt là true).
* Khi trò chơi kết thúc, điểm số được so sánh với điểm cao nhất trước đó. Nếu lớn hơn, nó được cập nhật.

**Giao Diện Đồ Họa:**

* Giao diện đồ họa được vẽ lại trong mỗi chu kỳ với các đối tượng rắn và thức ăn mới.
* Thông tin như tên người chơi, điểm số, và điểm số cao nhất cũng được hiển thị.

**Tương Tác Bàn Phím:**

* Khi người chơi nhấn một phím, sự kiện KeyEvent được kích hoạt và xử lý trong keyPressed phương thức.
* Hướng di chuyển của rắn được cập nhật tùy thuộc vào phím được nhấn.
* Nếu trò chơi kết thúc và người chơi nhấn SPACE, trò chơi sẽ khởi động lại.

**Điểm Số Cao Nhất:**

* Điểm số cao nhất và tên người chơi được lưu và hiển thị trong hộp thoại JOptionPane thông báo.

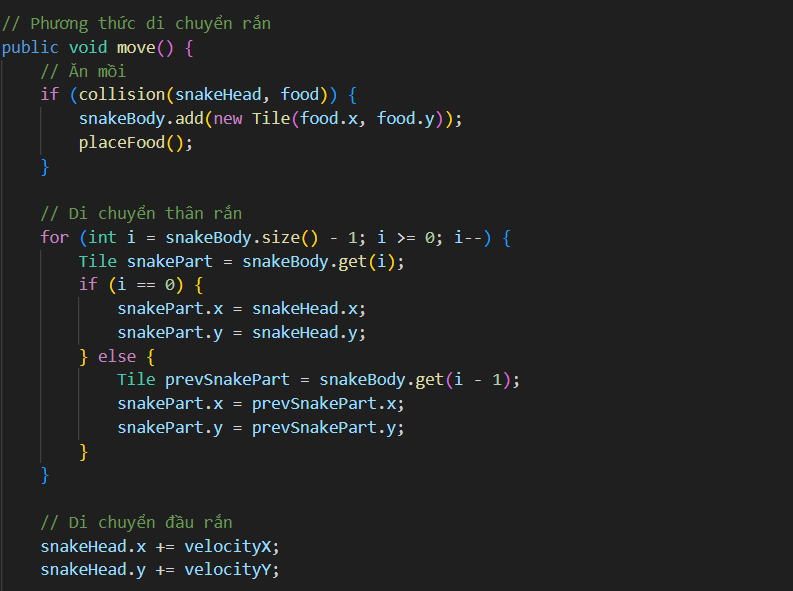
**Chế Độ Chơi Lại:**

* Nếu người chơi nhấn SPACE sau khi trò chơi kết thúc, trò chơi sẽ khởi động lại với đầu rắn và điểm số reset.

# CHƯƠNG 2: PHƯƠNG PHÁP NGHIÊN CỨU

## 2.1. Cách thức rắn di chuyển

Cách rắn di chuyển trong game của bạn được quản lý thông qua việc cập nhật tọa độ của đầu rắn và thân rắn. Dưới đây là cách thức di chuyển của rắn được triển khai trong phương thức move():



**Hình 2.1 Cách thức rắn di chuyển**

Nếu đầu rắn va chạm vào thức ăn, tạo một phần mới của rắn tại vị trí thức ăn và đặt lại vị trí của thức ăn bằng cách gọi placeFood().

Sử dụng vòng lặp để di chuyển mỗi phần của thân rắn. Nếu đang xử lý phần đầu tiên, lấy vị trí từ đầu rắn; ngược lại, lấy vị trí từ phần trước đó trong danh sách thân rắn.

Di chuyển đầu rắn theo hướng được xác định bởi velocityX và velocityY.

## 2.2. Cách thức lưu và tải điểm cao nhất

Cách thức lưu và tải điểm cao nhất được thực hiện thông qua lớp "ScoreManager":

### 2.2.1. Lưu điểm cao nhất

Khi trò chơi kết thúc, điểm số mới sẽ được lưu bằng cách gọi phương thức "saveScore(playerName, score)" của lớp "ScoreManager".

Danh sách điểm số được tải từ tệp tin, và điểm số mới được thêm vào danh sách.

Danh sách sau đó được sắp xếp giảm dần và giữ lại chỉ TOP\_SCORES\_COUNT phần tử.

A computer screen shot of a program code

Description automatically generated

**Hình 2.2 Cách thức lưu điểm cao nhất**

### 2.2.2. Tải điểm cao nhất

Khi cần hiển thị điểm số cao nhất, phương thức "loadScores()" của "ScoreManager" được gọi để tải danh sách điểm số từ tệp tin.

Danh sách sau đó được sắp xếp giảm dần và trả về để hiển thị.

A screen shot of a computer program

Description automatically generated

**Hình 2.3 Cách thức tải điểm cao nhất**

# CHƯƠNG 3: KẾT QUẢ THỰC HIỆN

## 3.1. Giao diện menu

A screenshot of a computer

Description automatically generated

**Hình 3.1 Giao diện nhập tên**

Menu giao diện trò chơi cho phép chúng ta nhập tên người chơi khi vừa mở

A screenshot of a computer

Description automatically generated

**Hình 3.2 Bảng xếp hạng sau khi nhập tên**

Bảng xếp hạng top 3 người điểm cao nhất sẽ xuất hiện sau khi bạn đặt tên.

## 3.2. Giao diện game play

A screenshot of a computer

Description automatically generated

**Hình 3.3 Giao diện khi chơi**

A screenshot of a computer

Description automatically generated

**Hình 3.4 Giao diện khi kết thúc trò chơi**

**A screenshot of a computer

Description automatically generated**

**Hình 3.5 BXH được cập nhật sau khi chơi**

Màn hình trò chơi khi điểm của bạn cao hơn 1 trong 3 người cao nhất.

# CHƯƠNG 4: TỔNG KẾT

Trong đề tài này, em đã chọn phát triển trò chơi Rắn săn mồi sử dụng ngôn ngữ lập trình Java và môi trường phát triển Visual Studio Code (VS Code). Trò chơi này mang tính giải trí, nơi người chơi có thể điều khiển con rắn để ăn thức ăn và mở rộng chiều dài của nó mà không chạm vào tường hoặc cơ thể của mình..

Các công việc cụ thể bao gồm việc cài đặt và cấu hình môi trường phát triển trên VS Code, xây dựng nền tảng cho trò chơi Rắn săn mồi, và triển khai các tính năng như logic di chuyển của rắn, xử lý va chạm, điều khiển, âm thanh và đồ họa. Qua quá trình này, tôi đã áp dụng lập trình hướng đối tượng vào phát triển trò chơi.

Qua quá trình thực tập cơ sở, em đã đạt được kỹ năng lập trình Java trên VS Code, từ quy trình lập trình đến biên dịch và debug. Đồng thời, thành thạo trong việc quản lý mã nguồn sử dụng Git và GitHub, giúp theo dõi và ghi lại sự thay đổi. Trong phần này, tôi cũng đã chia sẻ trải nghiệm và học hỏi từ bạn bè và người hướng dẫn.

# TÀI LIỆU THAM KHẢO

1. Hướng dẫn cơ bản đến nâng cao: [Java Tutorial (w3schools.com)](https://www.w3schools.com/java/default.asp)

2. Source Code: [Kenny Yip Coding](https://www.kennyyipcoding.com/)

3. Thư viện, API, cú pháp trong Java: [Java Documentation - Get Started (oracle.com)](https://docs.oracle.com/en/java/)

4. Hướng dẫn sử dụng môi trường VS Code: [Documentation for Visual Studio Code](https://code.visualstudio.com/docs)