**ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

**KHOA CÔNG NGHỆ THÔNG TIN**



**Báo cáo đồ án môn học**

**Ngôn ngữ lập trình python**

**Tên đề tài: Xây dựng ứng dụng game Tic tac toe**

Sinh viên thực hiện: **Lê Công Duy**

Mã Lớp*:* **DCT1225**

Giảng viên hướng dẫn : **Trịnh Tấn Đạt**

Thành phố Hồ Chí Minh, 2 tháng 5 năm 2024

**LỜI CẢM ƠN**

Em xin gửi lời cảm ơn chân thành và sự tri ân sâu sắc đối với các thầy cô của trường Đại Học Sài Gòn, đặc biệt là các thầy cô ở khoa Công Nghệ Thông Tin của trường đã tạo điều kiện cho em tiếp cận và tìm hiểu để hoàn thành đồ án môn học lần này. Và em cũng xin chân thành cám ơn thầy Trịnh Tấn Đạt giáo viên giảng dạy đã nhiệt tình hướng dẫn chúng em hoàn thành đồ án lần này.

Trong quá trình nghiên cứu và làm bài báo cáo đồ án, do kiến thức cũng như kinh nghiệm thực tế còn nhiều hạn chế nên bài báo cáo không thể tránh khỏi những thiếu sót, chúng em rất mong nhận được ý kiến đóng góp thầy, cô để em học hỏi được nhiều kĩ năng, kinh nghiệm và sẽ hoàn thành tốt hơn cho những bài báo cáo sắp tới.

Em xin chân thành cảm ơn thầy ạ!

**NHẬN XÉT, ĐÁNH GIÁ CỦA GIẢNG VIÊN**

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

...............................................................................................................................................

**MỤC LỤC**

[Phần I. MỞ ĐẦU 6](#_Toc24996)

[**1. Giới thiệu về đề tài** 7](#_Toc24997)

[**2. Lý do chọn đề tài** 8](#_Toc24998)

[**3. Mục đích - mục tiêu của để tài** 8](#_Toc24999)

[PHẦN II. NỘI DUNG 9](#_Toc25000)

[**1. Tổng quan về game Tic Tac Toe** 9](#_Toc25001)

[**1.1 Giới thiệu về game Tic Tac Toe** 10](#_Toc25002)

[**2. Tổng quan về thuật toán minimax** 10](#_Toc25001)

[**2.1 Giới thiệu về thuật toán Minimax** 11](#_Toc25002)

[**2.2 Ứng dụng của thuật toán Minimax trong game Tic tac toe** 11](#_Toc25002)

[**3. Tổng quan về thư viện pygame** 12](#_Toc25001)

[**3.1 Giới thiệu về thư viện pygame** 13](#_Toc25002)

[**4. Yêu cầu đối với đồ án** 13](#_Toc25004)

[**5. Tiến hành xây dựng game Tic tac toe** 14](#_Toc25005)

[**5.1.Cài đặt pygame và các biến cho trò chơi** 15](#_Toc25006)

[**5.2. Thiết kế giao diện và khởi tạo trò chơi**  18](#_Toc25007)

[**5.3 Xử lý lượt chơi của người và máy**  22](#_Toc25009)

[**5.4 Kiểm tra trạng thái và sử lý kết quả** 25](#_Toc25010)

[**6. Kết luận** 28](#_Toc25013)

[**6.1.Tổng kết về báo cáo** 28](#_Toc25006)

## 

## Phần I. MỞ ĐẦU

# 1. Giới thiệu đề tài

Trò chơi Tic Tac Toe, hay còn được biết đến với tên gọi khác là Xs and Os, là một trò chơi giải đố đơn giản nhưng hấp dẫn, thường được chơi giữa hai người. Trong trò chơi này, người chơi thường đại diện cho hai biểu tượng khác nhau, thường là "X" và "O", và lần lượt đặt các biểu tượng này trên một bảng gồm các ô vuông nhỏ để tạo thành một dãy liên tiếp theo chiều ngang, dọc hoặc chéo.

Dự án này nhằm mục đích tạo ra một phiên bản của trò chơi Tic Tac Toe trên máy tính bằng cách sử dụng ngôn ngữ lập trình Python và thư viện Pygame. Trò chơi sẽ cung cấp cả hai chế độ chơi: chế độ chơi với máy và chế độ chơi hai người. Ngoài ra, dự án cũng sẽ áp dụng thuật toán Minimax, một phương pháp trí tuệ nhân tạo, để tạo ra một máy chơi Tic Tac Toe thông minh.

Mục tiêu của dự án là tạo ra một trò chơi Tic Tac Toe có giao diện thân thiện với người dùng, cung cấp trải nghiệm chơi game mượt mà và thú vị, đồng thời cung cấp lựa chọn cho người chơi giữa việc chơi với máy hoặc với một người khác. Bằng cách này, dự án hy vọng mang lại niềm vui và sự giải trí cho người chơi mọi lứa tuổi.

# 2. Lý do chọn đề tài.

Trò chơi Tic Tac Toe đã tồn tại từ lâu và luôn là một trong những trò chơi giải đố đơn giản nhưng vẫn thu hút người chơi bởi tính giải trí và tính thách thức của nó. Lý do chúng tôi chọn đề tài này là vì nó mang lại nhiều lợi ích và cơ hội phát triển trong nhiều mặt:

**Phổ biến và quen thuộc:** Tic Tac Toe là một trò chơi phổ biến và quen thuộc với hầu hết mọi người. Việc phát triển một phiên bản mới của trò chơi này không chỉ tạo ra một trải nghiệm giải trí mà còn tạo ra sự gắn kết với người chơi.

**Thú vị và giáo dục:** Trò chơi này cung cấp một cơ hội tuyệt vời để kích thích trí óc và logic của người chơi. Nó giúp cải thiện kỹ năng tư duy chiến lược, dự đoán và lập kế hoạch, đồng thời mang lại niềm vui và thách thức.

**Học hỏi và thử nghiệm:** Việc phát triển một phiên bản Tic Tac Toe trên máy tính đòi hỏi sự hiểu biết sâu sắc về lập trình Python. Điều này sẽ làm giàu kiến thức và kỹ năng lập trình của chúng tôi thông qua việc thử nghiệm và áp dụng các khái niệm lập trình cụ thể.

**Làm quen với ngôn ngữ Python:** Trong quá trình phát triển trò chơi, chúng tôi sẽ có cơ hội làm quen với ngôn ngữ lập trình Python. Điều này giúp chúng tôi hiểu rõ hơn về cú pháp, cấu trúc và các tính năng của Python, mở ra cơ hội học hỏi và ứng dụng rộng rãi trong tương lai.

Với những lý do trên, chúng tôi tin rằng việc phát triển một phiên bản Tic Tac Toe trên máy tính không chỉ mang lại niềm vui cho người chơi mà còn cung cấp nhiều cơ hội học hỏi và phát triển kỹ năng lập trình.

# 3. Mục đích - mục tiêu của để tài.

**3.1 Mục đích:**

Mục đích chính của dự án này là xây dựng một phiên bản mới của trò chơi Tic Tac Toe trên nền tảng máy tính bằng sử dụng ngôn ngữ lập trình Python và thư viện Pygame. Qua dự án này, chúng tôi muốn:

Tạo ra một sản phẩm giải trí hấp dẫn và thú vị cho người chơi.

Áp dụng và thử nghiệm các kiến thức lập trình Python đã học vào một dự án thực tế.

Nắm vững các khái niệm cơ bản về lập trình game, bao gồm xử lý sự kiện, hiển thị đồ họa, và quản lý trạng thái của trò chơi.

Phát triển kỹ năng làm việc nhóm và quản lý dự án thông qua việc phối hợp làm việc với đồng đội và tuân thủ tiến độ.

**3.2 Mục tiêu:**

Dự án đặt ra một số mục tiêu cụ thể như sau:

Hoàn thiện và triển khai một phiên bản Tic Tac Toe hoạt động mượt mà và ổn định trên nền tảng máy tính.

Phát triển giao diện người dùng thân thiện và dễ sử dụng, bao gồm các tính năng như chơi mới, tùy chỉnh cài đặt, và xem điểm số.

Tối ưu hóa mã nguồn để đảm bảo hiệu suất cao và tính mở rộng của trò chơi.

Hiểu rõ hơn về cách làm việc với thư viện Pygame và áp dụng các nguyên tắc thiết kế game vào dự án.

## PHẦN II. NỘI DUNG

**1. Tổng quan về game Tic Tac Toe**

**1.1 Giới thiệu về game Tic Tac Toe**

Tic Tac Toe, hoặc còn gọi là "X và O" hoặc "Caro", là một trò chơi giải đố đơn giản nhưng phổ biến trên toàn thế giới. Trò chơi thú vị này thường được chơi trên một bảng 3x3, với mục tiêu tạo ra một chuỗi liên tiếp ba dấu X hoặc O trên hàng, cột hoặc đường chéo của bảng.

**1.2. Mục đích và Mục tiêu của Đề tài**

**Mục tiêu:**

Phát triển giao diện đồ họa thân thiện và dễ sử dụng.

Cung cấp chế độ chơi đơn và chơi hai người.

Tích hợp trí tuệ nhân tạo (AI) thông qua thuật toán Minimax.

Kiểm tra và xác định kết quả của trò chơi.

Tối ưu hóa hiệu suất và trải nghiệm người dùng.

**1.3 Yêu Cầu Đối Với Đồ Án**

Đảm bảo mã nguồn dễ đọc và tuân thủ các quy tắc về clean code.

Thực hiện kiểm thử kỹ lưỡng để đảm bảo tính đúng đắn và ổn định của trò chơi.

Phát triển các tính năng mở rộng để cải thiện trải nghiệm người chơi.

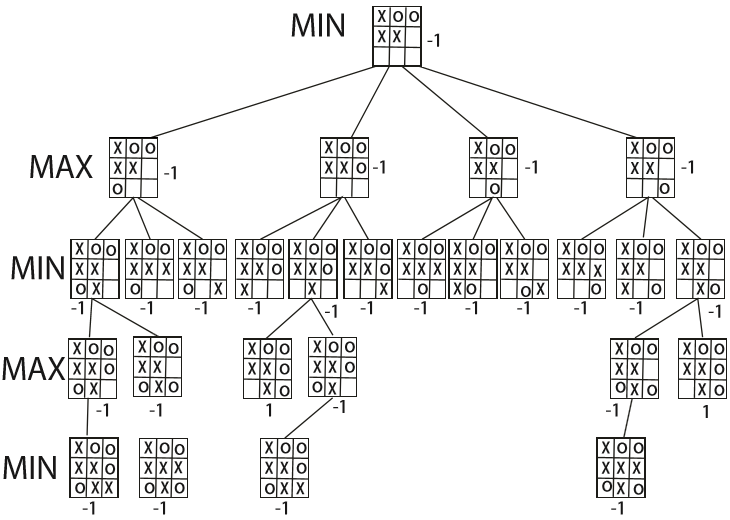
**2. Tổng quan về thuật toán Minimax**

**2.1 .Giới thiệu về thuật toán Minimax**

Thuật toán Minimax là một phương pháp được sử dụng trong trò chơi và trí tuệ nhân tạo để tìm kiếm nước đi tối ưu trong một tình huống có nhiều lựa chọn và các kết quả khả thi.

Minimax hoạt động bằng cách mô phỏng các bước diễn ra của trò chơi từ tương lai, đánh giá giá trị của mỗi nước đi và chọn nước đi tốt nhất cho bản thân (người chơi), trong khi giả định rằng đối thủ sẽ chọn nước đi tốt nhất cho họ (máy tính), dựa trên nguyên tắc tối thiểu hóa tổn thất tối đa.

Mặc dù Minimax có thể giải quyết một số trò chơi đơn giản như Tic Tac Toe một cách hiệu quả, nhưng đối với các trò chơi phức tạp hơn, cần các phương pháp tối ưu hóa như cắt alpha-beta để giảm thời gian tính toán. Minimax là một công cụ mạnh mẽ trong lĩnh vực trò chơi và trí tuệ nhân tạo, được sử dụng rộng rãi trong việc xây dựng các trò chơi và hệ thống thông minh nhân tạo.



**Ví dụ mô phỏng giải thuật Minimax cho trò chơi Tic-Tac-toe**

MAX đại diện quân đi O.

MIN đại diện quân đi X.

Trạng thái kết thúc là trạng thái có 3 ô liên tiếp ngang, dọc, chéo có cùng một quân cờ X hoặc O, nếu là X tức MIN thắng còn O tức MAX thắng còn nếu tất cả các ô cờ đều được đi và trạng thái chưa kết thúc thì bàn cờ hòa. Điểm thắng của X là -1, của O là 1, và bàn cờ hòa là 0.

Áp dụng giải thuật Minimax:

Từ trạng thái bàn cờ hiện tại ta dự đoán nước đi của trạng thái tiếp theo nếu trạng thái tiếp theo ta tiến hành lượng giá cây trò chơi bằng cách ta tiến hành quét cạn tất cả các trạng thái tiếp theo cho đến lúc gặp trạng thái chiến thắng (Node lá) tính điểm cho Node lá bằng cách:

Nếu ở trạng thái mà ta gặp chiến thắng nếu đó là lượt đi của quân X  thì đánh giá điểm trạng thái đó là -1.

Nếu ở trạng thái ta gặp chiến thắng nếu đó là lượt đi của  quân O thì đánh giá điểm trạng thái đó là 1.

Nếu là hòa thì điểm trạng thái đó là 0.

Sau đó tính ngược lại cây trò chơi theo quy tắc:

Nút thuộc lớp MAX thì gán cho nó giá trị lớn nhất của các Node con của Node đó.

Nút thuộc lớp MAX thì gán cho nó giá trị nhỏ nhất của các Node con của Node đó.

Sau khi lượng giá hết cây trò chơi ta tiến hành chọn bước đi tiếp theo nguyên tắc:

Nếu lớp tiếp theo là MAX ta chọn Node con có giá trị lớn nhất.

Nếu lớp tiếp theo là MIN ta chọn Node con có giá trị nhỏ nhất.

[**3. Tổng quan về thư viện pygame**](#_Toc25001)

**3.1 Giới thiệu về thư viện pygame**

Pygame là một thư viện mã nguồn mở được sử dụng để phát triển trò chơi và ứng dụng đa phương tiện trong ngôn ngữ lập trình Python. Nó cung cấp các công cụ và thư viện cho việc xây dựng các ứng dụng đồ họa tương tác, từ trò chơi đơn giản đến các ứng dụng phức tạp hơn.

**Đặc điểm chính:**

Dễ dàng học và sử dụng: Pygame được thiết kế để dễ dàng tiếp cận cho cả người mới bắt đầu và những người có kinh nghiệm trong lập trình.

Đa nền tảng: Pygame hoạt động trên nhiều hệ điều hành, bao gồm Windows, macOS và Linux, làm cho việc phát triển ứng dụng đa nền tảng trở nên dễ dàng.

Hỗ trợ đa phương tiện: Thư viện cung cấp các chức năng để xử lý âm thanh, hình ảnh và video, giúp người phát triển tạo ra các trải nghiệm đa phương tiện phong phú.

Cộng đồng lớn: Có một cộng đồng lớn và năng động xung quanh Pygame, với nhiều tài liệu hướng dẫn, ví dụ và hỗ trợ từ các nhà phát triển khác.

Linh hoạt: Pygame cho phép người dùng tận dụng sức mạnh của Python để tạo ra các trò chơi và ứng dụng độc đáo và sáng tạo.

Các thành phần chính của Pygame bao gồm:

**pygame.display**: Quản lý cửa sổ và hiển thị đồ họa.

**pygame.event:** Xử lý sự kiện như nhấn phím và di chuyển chuột.

**pygame.sprite:** Quản lý và vẽ các đối tượng trong trò chơi.

**pygame.image và pygame.surface:** Xử lý hình ảnh và bề mặt đồ họa.

**pygame.mixer và pygame.mixer.music:** Điều khiển âm thanh và nhạc.

Với các tính năng và công cụ mạnh mẽ, Pygame là một lựa chọn lý tưởng cho việc phát triển các ứng dụng đa phương tiện và trò chơi trong Python.

# 4. Yêu cầu đối với đồ án

**Mục tiêu:** Xây dựng một trò chơi Tic Tac Toe đơn giản sử dụng ngôn ngữ lập trình Python.

**Yêu cầu cụ thể:**

Giao diện trò chơi: Hiển thị bảng 3x3 để chơi Tic Tac Toe.

Người chơi: Có thể là hai người chơi hoặc một người chơi đấu với máy.

Luật chơi: Người chơi lần lượt đặt "X" hoặc "O" trên bảng cho đến khi có người chiến thắng hoặc hòa.

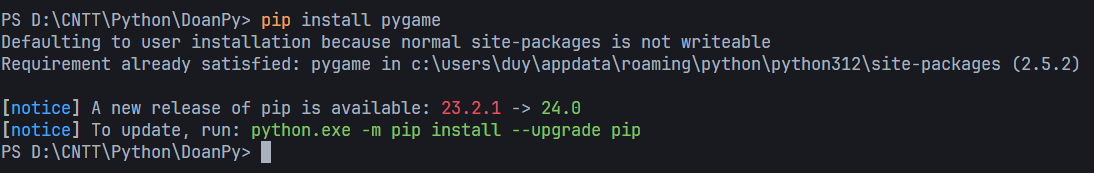
Xử lý kết quả: Phát hiện người chiến thắng hoặc hòa và kết thúc trò chơi.

Giao diện người dùng: Hiển thị thông báo kết quả và cho phép chơi lại.

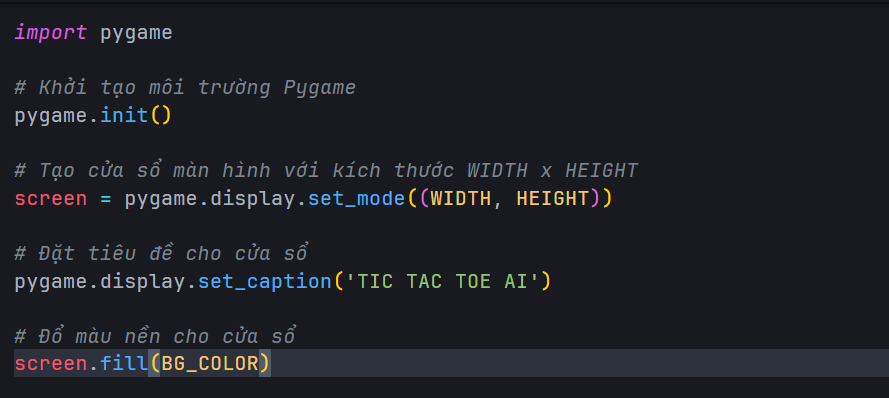
**Hướng dẫn thực hiện:** Sử dụng Python để xây dựng logic của trò chơi và tạo giao diện đồ họa đơn giản hoặc giao diện dòng lệnh để người chơi tương tác.

[**5. Tiến hành xây dựng game Tic tac toe**](#_Toc25005)

**[5.1.Cài đặt pygame và các biến cho trò chơi](#_Toc25006)**



Đầu tiên, bạn cần cài đặt thư viện Pygame trên máy tính của mình bằng cách sử dụng trình quản lý gói pip. Bạn có thể sử dụng câu lệnh sau trên dòng lệnh: **pip install pygame**



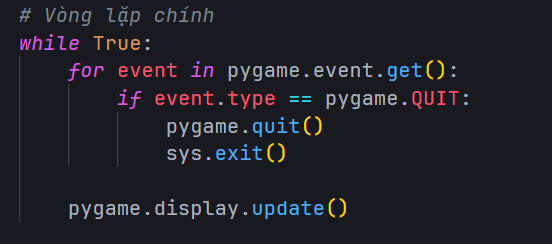
Trong phần này, chúng ta đã thực hiện các chức năng sau:

**pygame.init():** Khởi tạo môi trường Pygame.

**pygame.display.set\_mode((WIDTH, HEIGHT)):** Tạo cửa sổ màn hình với kích thước được chỉ định bởi các hằng số WIDTH và HEIGHT.

**pygame.display.set\_caption('TIC TAC TOE AI'):** Đặt tiêu đề cho cửa sổ.

**screen.fill(BG\_COLOR):** Đổ màu nền cho cửa sổ.



**while True:** Đây là một vòng lặp vô hạn, nghĩa là nó sẽ tiếp tục chạy mã bên trong mà không có điều kiện dừng rõ ràng.

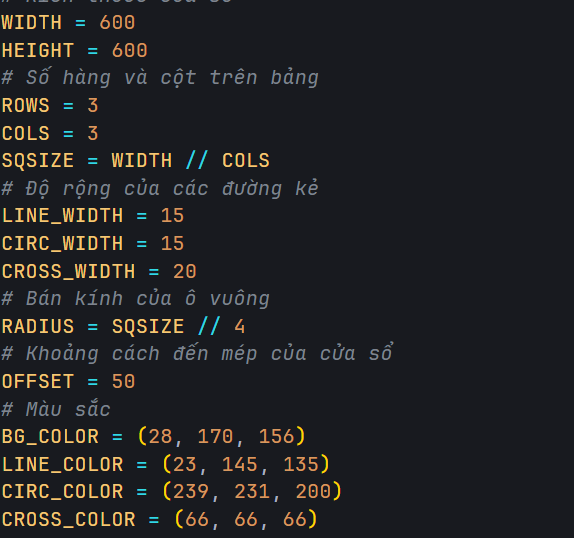
**for event in pygame.event.get():** Vòng lặp này lặp qua tất cả các sự kiện từ hàng đợi sự kiện của Pygame. Mỗi lần lặp, nó kiểm tra xem người dùng đã thực hiện hành động gì (như nhấn phím hoặc nhấp chuột).

**if event.type == pygame.QUIT:** Kiểm tra nếu sự kiện là nhấn nút thoát (QUIT), tức là người dùng muốn đóng cửa sổ trò chơi. Nếu có, Pygame sẽ được kết thúc và chương trình sẽ thoát.

**pygame.display.update():** Cập nhật màn hình. Mọi thay đổi về hình ảnh sẽ được vẽ lên màn hình.

**pygame.display.update():** Cập nhật màn hình sau mỗi vòng lặp để hiển thị các thay đổi mới.

Chức năng chính của đoạn mã là duy trì việc hiển thị và tương tác với giao diện của trò chơi .



**WIDTH**: Chiều rộng của cửa sổ màn hình.

**HEIGHT**: Chiều cao của cửa sổ màn hình.

**ROWS**: Số hàng trên bảng trò chơi.

**COLS**: Số cột trên bảng trò chơi.

**SQSIZE**: Kích thước của mỗi ô trên bảng (chiều rộng hoặc chiều cao).

**LINE\_WIDTH**: Độ dày của các đường kẻ trên bảng.

**CIRC\_WIDTH**: Độ dày của đường viền cho hình tròn (đánh dấu "O").

**CROSS\_WIDTH**: Độ dày của đường viền cho hình chữ nhật (đánh dấu "X").

**RADIUS**: Bán kính của hình tròn (đánh dấu "O").

**OFFSET**: Khoảng cách từ mép của bảng đến các ô.

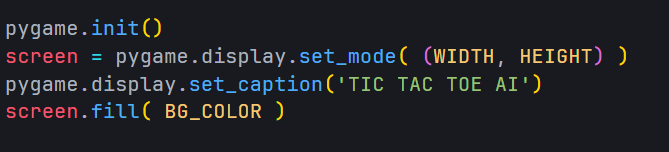
**BG\_COLOR**: Màu nền của cửa sổ.

**LINE\_COLOR**: Màu của các đường kẻ trên bảng.

**CIRC\_COLOR**: Màu của hình tròn (đánh dấu "O").

**CROSS\_COLOR**: Màu của hình chữ nhật (đánh dấu "X").

[**5.2. Thiết kế giao diện và khởi tạo trò chơi**](#_Toc25007)

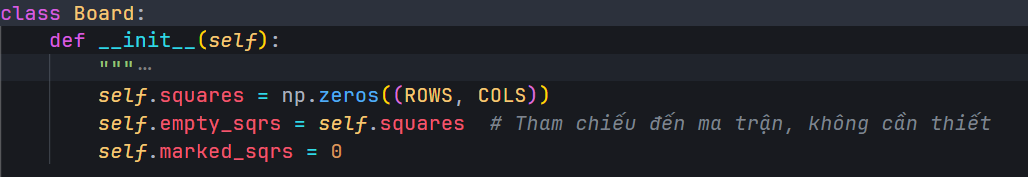
**5.2.1** **Khởi tạo trò chơi** 

Khởi tạo Pygame và cửa sổ trò chơi:

**5.2.2** **Thiết kế giao diện**



Khởi tạo class Board

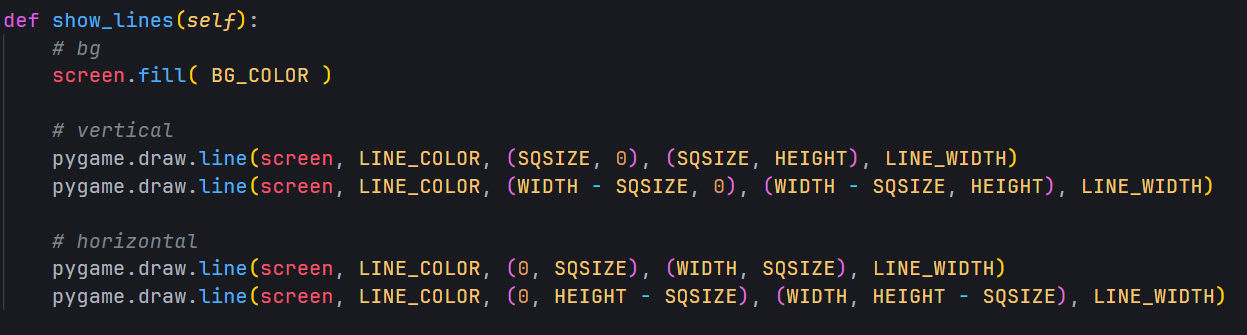


        Khởi tạo một bảng trống.

**- self.squares:** Ma trận các ô trên bảng.

**- self.empty\_sqrs:** Tham chiếu tới ma trận các ô trống (không cần thiết).

**- self.marked\_sqrs:** Số lượng ô đã được đánh dấu.



Mục đích của phương thức này là vẽ các đường kẻ ngang và dọc trên bảng của trò chơi Tic Tac Toe, cũng như làm sạch màn hình bằng cách điền màu nền.

**screen.fill(BG\_COLOR):** Đoạn mã này điền màu nền (được xác định bởi BG\_COLOR) vào màn hình, xóa bất kỳ nội dung nào đã được vẽ trước đó.

**pygame.draw.line(screen, LINE\_COLOR, (SQSIZE, 0), (SQSIZE, HEIGHT), LINE\_WIDTH):** Đoạn mã này vẽ đường kẻ dọc ở vị trí SQSIZE (chiều rộng của một ô) từ đầu màn hình đến cuối màn hình. Màu sắc của đường được xác định bởi LINE\_COLOR, và độ dày của đường là LINE\_WIDTH.

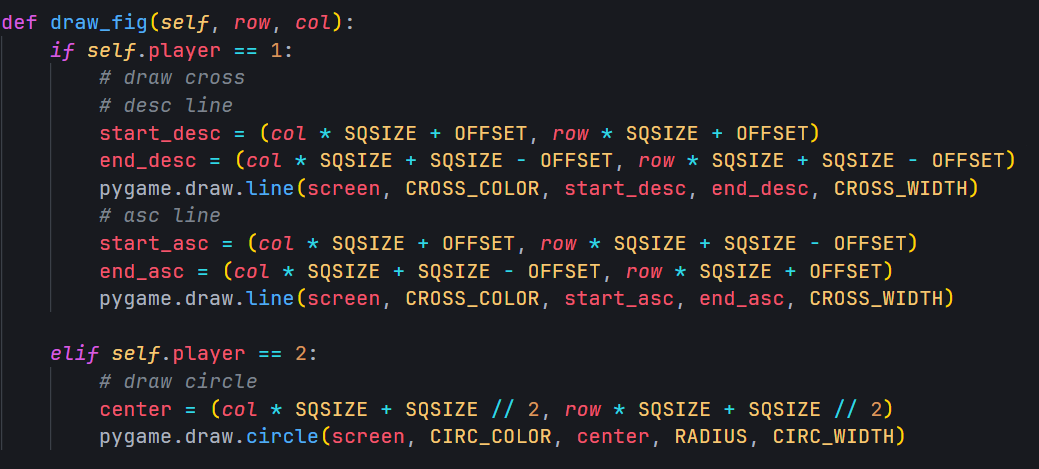
**pygame.draw.line(screen, LINE\_COLOR, (WIDTH - SQSIZE, 0), (WIDTH - SQSIZE, HEIGHT), LINE\_WIDTH):** Tương tự, đoạn mã này vẽ một đường kẻ dọc khác ở vị trí WIDTH - SQSIZE, từ đầu màn hình đến cuối màn hình.

**pygame.draw.line(screen, LINE\_COLOR, (0, SQSIZE), (WIDTH, SQSIZE), LINE\_WIDTH):** Đoạn mã này vẽ đường kẻ ngang tại vị trí SQSIZE (chiều cao của một ô), từ cạnh trái của màn hình đến cạnh phải của màn hình.

**pygame.draw.line(screen, LINE\_COLOR, (0, HEIGHT - SQSIZE), (WIDTH, HEIGHT - SQSIZE), LINE\_WIDTH):** Tương tự, đoạn mã này vẽ một đường kẻ ngang khác tại vị trí HEIGHT - SQSIZE, từ cạnh trái của màn hình đến cạnh phải của

màn hình. A blue grid with black lines

Description automatically generated with medium confidence



Mục đích của phương thức này là vẽ các biểu tượng (cờ) của người chơi trên bảng theo hàng và cột được chỉ định.

Kiểm tra người chơi hiện tại (self.player):

**Nếu người chơi là 1:**

Vẽ ký hiệu 'cross' (dấu chéo):

Tạo điểm bắt đầu (start\_desc) ở góc trên bên trái của ô, tính từ vị trí hàng và cột của ô và kích thước ô.

Tạo điểm kết thúc (end\_desc) ở góc dưới bên phải của ô, tính từ vị trí hàng và cột của ô và kích thước ô.

Vẽ đường chéo xuống từ start\_desc đến end\_desc với màu và độ dày xác định bởi CROSS\_COLOR và CROSS\_WIDTH.

Tạo điểm bắt đầu (start\_asc) ở góc dưới bên trái của ô, tính từ vị trí hàng và cột của ô và kích thước ô.

Tạo điểm kết thúc (end\_asc) ở góc trên bên phải của ô, tính từ vị trí hàng và cột của ô và kích thước ô.

Vẽ đường chéo lên từ start\_asc đến end\_asc với màu và độ dày xác định bởi CROSS\_COLOR và CROSS\_WIDTH.

A black x on a blue background

Description automatically generated

**Nếu người chơi là 2:**

Vẽ ký hiệu 'circle' (hình tròn):

Tính toán tâm của hình tròn (center) tại trung tâm của ô, tính từ vị trí hàng và cột của ô và kích thước ô.

Vẽ hình tròn tại tâm được tính toán với bán kính RADIUS, màu và độ dày xác định bởi CIRC\_COLOR và CIRC\_WIDTH.

A white circle on a blue background

Description automatically generated

**5.2.3** **Sử lí sự kiện và tương tác**

**1.** Khởi tạo các đối tượng Game, Board và AI:

A black background with blue and yellow text

Description automatically generated

Tạo các đối tượng cần thiết để bắt đầu trò chơi.

**2.** Vòng lặp chính (while True):

Xử lý sự kiện Pygame:

Sự kiện thoát (pygame.QUIT):

A computer screen shot of a black background with colorful text

Description automatically generated with medium confidence

Khi người dùng đóng cửa sổ trò chơi, sự kiện này được kích hoạt và trò chơi sẽ thoát

Sự kiện nhấn phím (pygame.KEYDOWN):

Chuyển đổi chế độ chơi (K\_g)

A black background with white text

Description automatically generated

Chuyển đổi giữa chế độ chơi PvP (Player vs Player) và AI (Player vs AI)

Khởi động lại trò chơi (K\_r)

A screen shot of a computer program

Description automatically generated

Đặt lại trò chơi về trạng thái ban đầu.

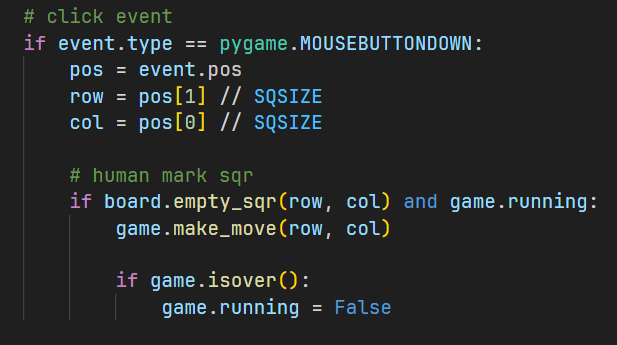
Chọn mức độ AI (K\_0 và K\_1):

A screen shot of a computer program

Description automatically generated

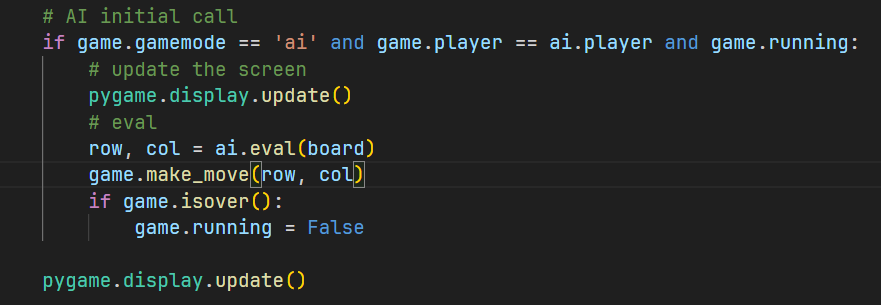
Chọn mức độ AI (0 là ngẫu nhiên, 1 là sử dụng thuật toán minimax).

Sự kiện nhấn chuột (pygame.MOUSEBUTTONDOWN):



Khi người dùng nhấn chuột, xác định vị trí nhấn và nếu ô đó trống, thực hiện đánh dấu ô và kiểm tra xem trò chơi đã kết thúc chưa.

Xử lý nước đi của AI:



Nếu đang ở chế độ AI và đến lượt của AI, AI sẽ tính toán nước đi tiếp theo và thực hiện nước đi đó.

Cập nhật màn hình:



Cập nhật màn hình sau mỗi vòng lặp.

[**5.3 Xử lý lượt chơi của người và máy**](#_Toc25009)

Thực hiện nước đi của người chơi:

A computer code on a black background

Description automatically generated

Hàm **make\_move** thực hiện nước đi của người chơi:

Đánh dấu ô trên bảng với ký hiệu của người chơi hiện tại (**self.board.mark\_sqr(row, col, self.player)**).

Vẽ ký hiệu của người chơi trên ô đã đánh dấu (**self.draw\_fig(row, col)**).

Chuyển đến lượt tiếp theo (**self.next\_turn()**).

Xử lý lượt chơi của AI trong vòng lặp chính :

A screen shot of a computer

Description automatically generated

Kiểm tra xem chế độ trò chơi là 'ai', người chơi hiện tại là AI (**game.player == ai.player**), và trò chơi đang chạy (**game.running**).

Cập nhật màn hình (**pygame.display.update()**).

AI đánh giá nước đi tốt nhất bằng cách gọi hàm **eval** (**row, col = ai.eval(board)**).

Thực hiện nước đi của AI (**game.make\_move(row, col)**).

Kiểm tra xem trò chơi đã kết thúc chưa (**game.isover()**). Nếu kết thúc, đặt **game.running** thành False.

Đánh giá nước đi của AI:

A computer code on a black background

Description automatically generated

Hàm **eval** quyết định nước đi của AI dựa trên cấp độ của AI:

Nếu cấp độ là 0, AI chọn ngẫu nhiên một ô trống (**move = self.rnd(main\_board)**).

Nếu cấp độ khác, AI sử dụng thuật toán Minimax để chọn nước đi tốt nhất (**eval, move = self.minimax(main\_board, False)**).

In ra nước đi đã chọn và giá trị đánh giá (**print(f'AI has chosen to mark the square in pos {move} with an eval of: {eval}')**).

Trả về nước đi (hàng và cột).

Thuật toán Minimax của AI:

A screen shot of a computer program

Description automatically generated

Thuật toán Minimax đánh giá tất cả các nước đi có thể và chọn nước đi tối ưu dựa trên việc tối đa hóa hoặc tối thiểu hóa điểm số.

Nếu trạng thái cuối cùng là thắng, thua hoặc hòa, trả về giá trị đánh giá và không có nước đi (**return 1, None**, **return -1, None**, **return 0, None**).

Nếu đang ở trạng thái tối đa hóa, tìm nước đi với giá trị đánh giá lớn nhất (**max\_eval**).

Nếu đang ở trạng thái tối thiểu hóa, tìm nước đi với giá trị đánh giá nhỏ nhất (**min\_eval**).

Trả về giá trị đánh giá và nước đi tốt nhất.

[**5.4 Kiểm tra trạng thái và sử lý kết quả**](#_Toc25010)

Kiểm tra trạng thái cuối cùng của trò chơi:

A computer screen with text and images

Description automatically generated

Hàm **final\_state** kiểm tra xem có ai chiến thắng hay không bằng cách duyệt qua các hàng, cột và đường chéo của bảng.

Nếu một hàng, cột hoặc đường chéo có cùng ký hiệu và khác 0, tức là có người chiến thắng.

Nếu tham số **show** là True, vẽ đường kẻ nối các ô chiến thắng trên màn hình.

Trả về 0 nếu không có ai chiến thắng, 1 nếu người chơi 1 chiến thắng, và 2 nếu người chơi 2 chiến thắng.

Kiểm tra xem bảng đã đầy chưa:

A screenshot of a computer

Description automatically generated

Hàm **isfull** kiểm tra xem tất cả các ô trên bảng đã được đánh dấu hay chưa.

Nếu số lượng ô đã được đánh dấu là 9, tức là bảng đã đầy, trả về True. Ngược lại, trả về False.

Kiểm tra xem bảng có trống không:

A screenshot of a computer

Description automatically generated

Hàm **isempty** kiểm tra xem bảng có trống không.

Nếu số lượng ô đã được đánh dấu là 0, tức là bảng trống, trả về True. Ngược lại, trả về False.

Kiểm tra và xử lý kết quả trong vòng lặp chính:



Hàm **isover** kiểm tra xem trò chơi đã kết thúc chưa bằng cách gọi hàm **final\_state** của bảng và kiểm tra xem bảng đã đầy chưa (**self.board.isfull()**).

Nếu có người chiến thắng hoặc bảng đã đầy, trò chơi kết thúc, trả về True. Ngược lại, trả về False.

**6 .Tổng kết**

Quá trình phát triển trò chơi Tic Tac Toe này không chỉ cung cấp một ứng dụng giải trí đơn giản mà còn là một bài tập thực hành tuyệt vời về lập trình hướng đối tượng, xử lý sự kiện, và áp dụng thuật toán trong trí tuệ nhân tạo. Qua việc phân tích từng phần của mã nguồn, chúng ta đã hiểu rõ hơn về cách xây dựng và tối ưu hóa một trò chơi điện tử cơ bản. Dự án này cũng mở ra nhiều hướng phát triển và cải tiến trong tương lai, chẳng hạn như nâng cao giao diện người dùng, tối ưu hóa thuật toán AI, và mở rộng trò chơi với các tính năng mới.

Hy vọng rằng báo cáo này đã cung cấp một cái nhìn tổng quan và chi tiết về quá trình xây dựng và hoạt động của trò chơi Tic Tac Toe với AI, giúp ích cho những ai quan tâm đến lập trình trò chơi và trí tuệ nhân tạo.