| **THUYẾT MINH SẢN PHẨM SÁNG TẠO** | **Đơn vị:**  **Thành phố Hồ Chí Minh** |
| --- | --- |

**--------------------**

**ĐĂNG KÝ THÔNG TIN SẢN PHẨM DỰ THI**

| **1** | **Đăng ký Bảng dự thi** | **Bảng D1** | **Bảng D2** | | **Bảng D3** |
| --- | --- | --- | --- | --- | --- |
| **☐** | **☐** | | **✅** |
| **2** | **Sản phẩm dự thi** | **Phần mềm** | | **SP tích hợp phần cứng** | |
| **✅** | | **☐** | |
| **3** | **Tên SPST dự thi** | **COGANH** **-** Game AI rèn luyện lập trình Python & thuật toán dựa trên trò chơi cờ gánh (một trò chơi dân gian của Việt Nam) | | | |
| **4** | **Ngôn ngữ lập trình hoặc nền tảng** | Python, JavaScript, HTML, CSS | | | |
| **5** | **Cấu hình cài đặt** | Website | | | |

**THÔNG TIN TÁC GIẢ (NHÓM TÁC GIẢ)**

| **Số lượng thí sinh trong đội thi** | **1 người ☐** | **2 người ✅** |
| --- | --- | --- |
| **Thí sinh thứ nhất (đội trưởng)** | | |
| **Họ và tên:** | Thông Minh Quân | |
| **Ngày/tháng/năm sinh:** | 06/04/2007 | |
| **Lớp, trường:** | Lớp 11A6, trường THPT Nguyễn Hữu Huân | |
| **Quận, huyện:** | Thành phố Thủ Đức, thành phố Hồ Chí Minh | |
| **Điện thoại:** | 0961560378 | |
| **Email:** | minhquanthong064@gmail.com | |
| **Thí sinh thứ hai** | | |
| **Họ và tên:** | Lê Tuấn Vũ | |
| **Ngày/tháng/năm sinh:** | 27/02/2008 | |
| **Lớp, trường:** | Lớp 10D2, trường THPT Nguyễn Hữu Huân | |
| **Quận, huyện:** | Thành phố Thủ Đức, thành phố Hồ Chí Minh | |
| **Điện thoại:** | 0345613927 | |
| **Email:** | ngovipmo1@gmail.com | |
| **Giáo viên hoặc chuyên gia hướng dẫn** | | |
| **Họ và tên:** | Nguyễn Thị Bích Hảo | |
| **Đơn vị công tác:** | Trường THPT Nguyễn Hữu Huân | |
| **Chức vụ:** | Giáo viên | |
| **Điện thoại:** | 0909372800 | |
| **Email:** | nt.bichhao@gmail.com | |

**GIỚI THIỆU VỀ SẢN PHẨM**

**1. Ý tưởng của sản phẩm**

COGANH là một hệ thống website tổ chức thi đấu lập trình đối kháng lấy ý tưởng từ cờ gánh - một trò chơi dân gian của Việt Nam.

**2. Giới thiệu tổng quan**

Với COGANH, người chơi sẽ được lựa chọn:

+ Người đấu với người (không lập trình)

+ Người đấu với bot AI của hệ thống (không lập trình)

+ Tạo chương trình (bot) để đấu với bot AI của hệ thống

+ Tạo chương trình (bot) để đấu với bot của người chơi khác

**MÔ TẢ SẢN PHẨM**

**1. Chức năng chính của sản phẩm**

- Cờ gánh là ý tưởng mới hoàn toàn, ra đời với mục tiêu là hỗ trợ học sinh phát triển khả năng xây dựng thuật toán và kỹ thuật sử dụng ngôn ngữ lập trình Python.

- Áp dụng phương pháp Learning code by game đang phổ biến trên thế giới kết hợp với ý tưởng mới lạ được lấy từ một trò chơi dân gian mang đậm tính dân tộc, Cờ gánh không chỉ khơi gợi sự tò mò, niềm đam mê của học sinh với lập trình mà qua đó sản phẩm góp phần tái sinh một phần văn hóa dân tộc đến các bạn trẻ qua cách tiếp cận độc đáo.

- Khác với những dạng lập trình thi đấu hiện nay, kiểu học ở COGANH có những điểm mới sau:

o Có mô phỏng trực tiếp nên việc học lập trình không bị nhàm chán, khô cứng

o Có tính hướng đích hơn vì học sinh biết rõ ứng dụng thực tế của sản phẩm, ngoài ra việc chia nhiều cấp độ chơi từ dễ tới khó giúp người dùng dễ tiếp cận hơn.

o Có đánh giá tính chiến thuật của nước đi giúp cải thiện chiến thuật của người chơi và thuật toán của bot.

o Nếu cờ gánh là đề bài thì đây là 1 bài toán ko có đáp án, không giới hạn thuật toán, vẫn có thể tồn tại thuật toán tốt hơn thuật toán tốt nhất hiện tại. Từ đó tăng tính sáng tạo, tăng đam mê học hỏi của học sinh lên mức tối đa.

Không chỉ học sinh, những đối tượng nghiên cứu sâu về AI vẫn có thể sử dụng COGANH để nghiên cứu thuật toán mới.

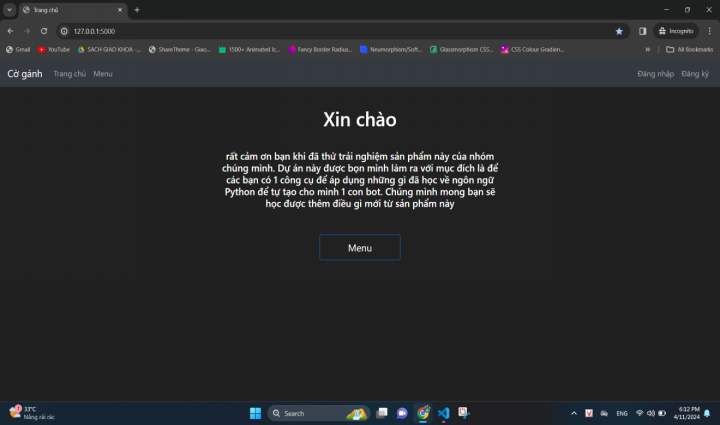
***1.2. Mô tả nền tảng phát triển của sản phẩm***

**1.2.1** **Ngôn ngữ lập trình:** Python, JavaScript, HTML, CSS

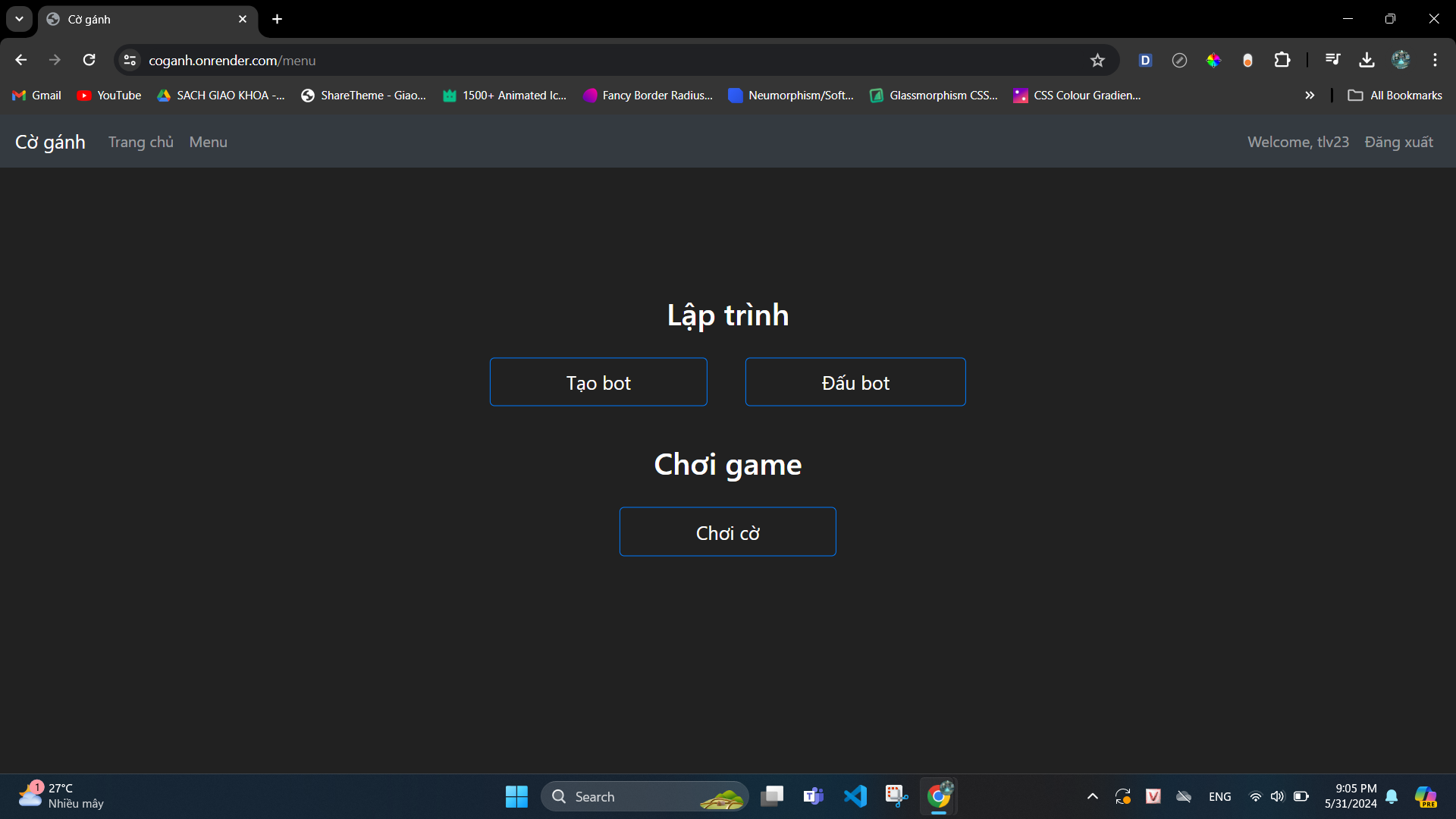
**1.2.2** **Nền tảng để phát triển các ứng dụng:** Website

**1.2.3** **Chi tiết các trang**

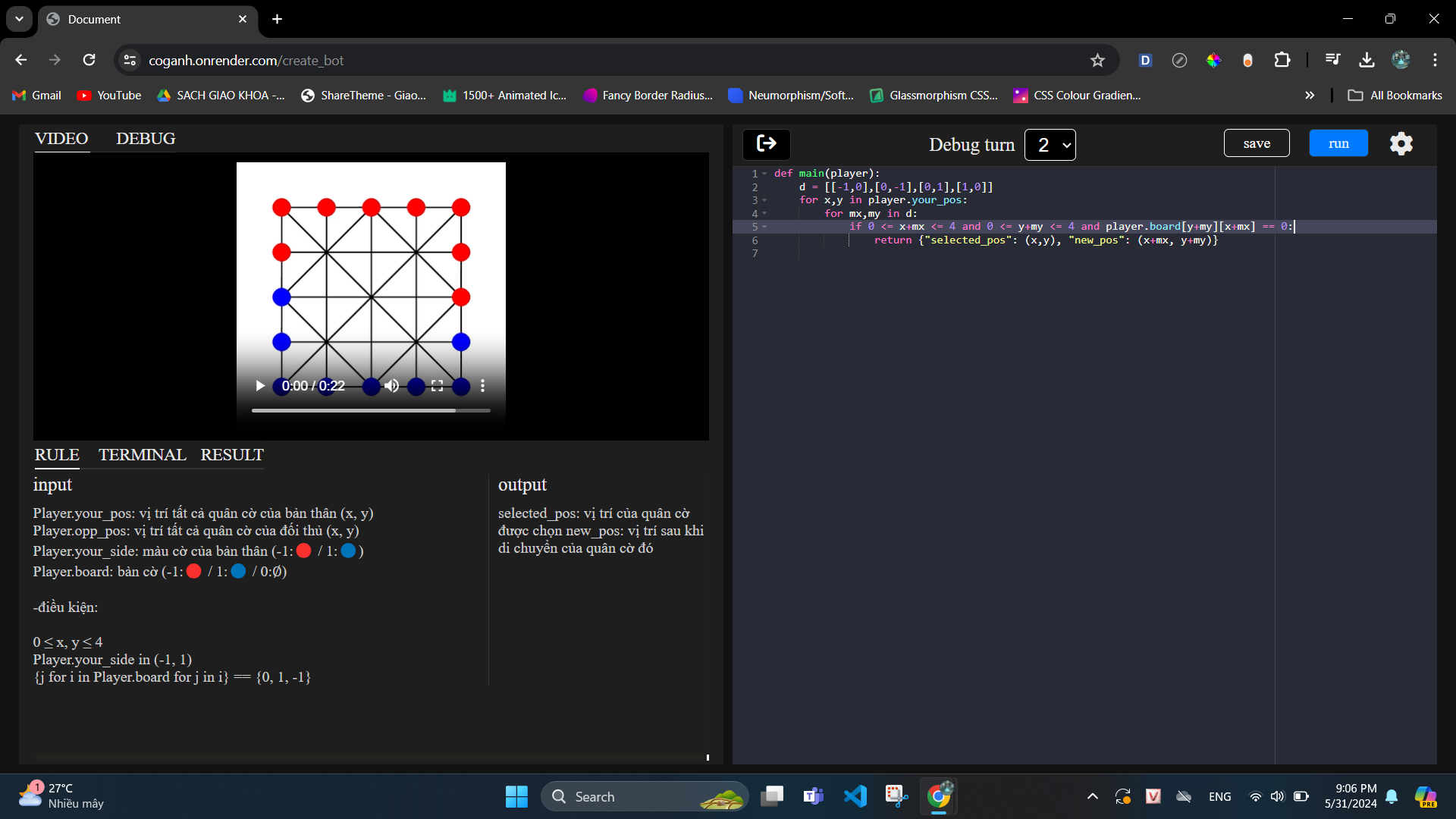
* home\_page



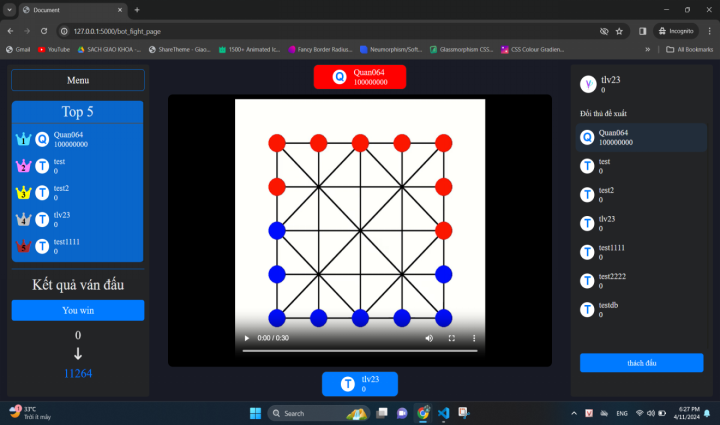
* menu



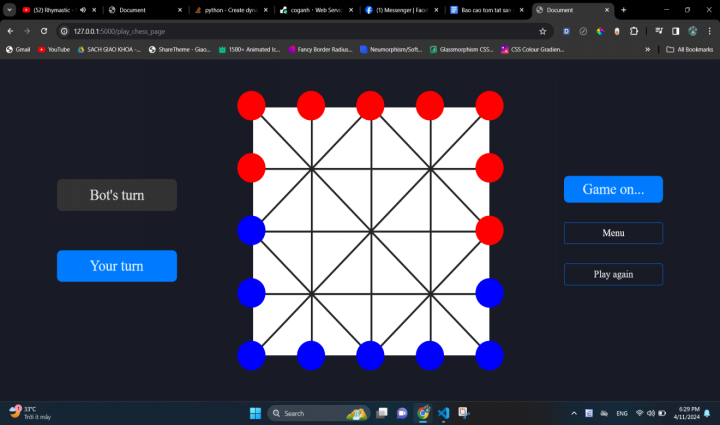
* create\_bot



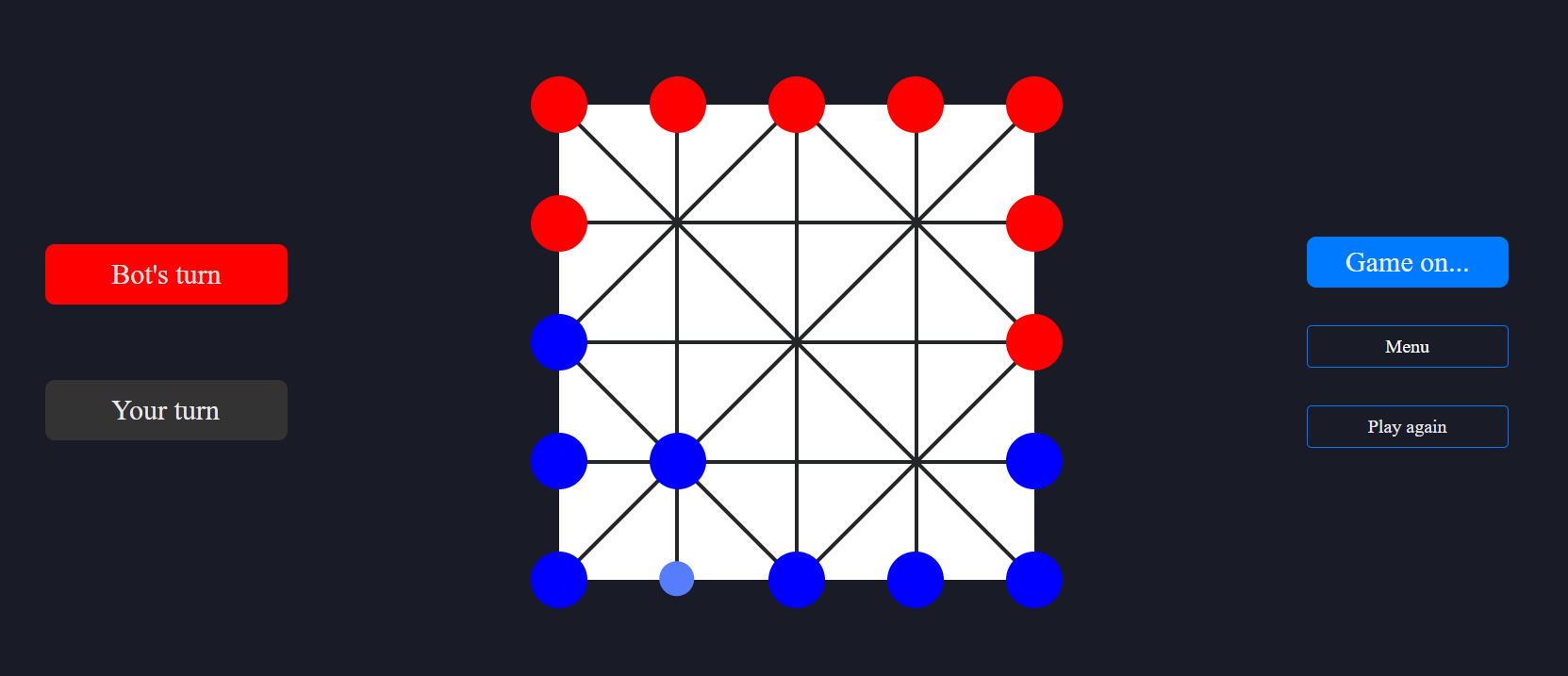
* bot\_bot\_page (nơi để bot của người chơi đấu với nhau)



* human\_bot\_page (nơi để người chơi trải nghiệm chơi cờ gánh với bot của hệ thống)



* human\_human\_page (nơi để hai người cùng chơi cờ gánh không phân biệt khoảng cách địa lý)



## 1.2.4. Trình bày mã nguồn theo từng Modun chương trình

### 1.2.4.1 Cấu trúc file/ [folder]

[Code github](https://github.com/Quan064/Co_ganh)

*Công dụng các file*:

main.py: xây dựng Web Application

game\_manager.py: vận hành ván đấu

[instance]/database.db: cơ sở dữ liệu người dùng

[static]: nguồn tổng

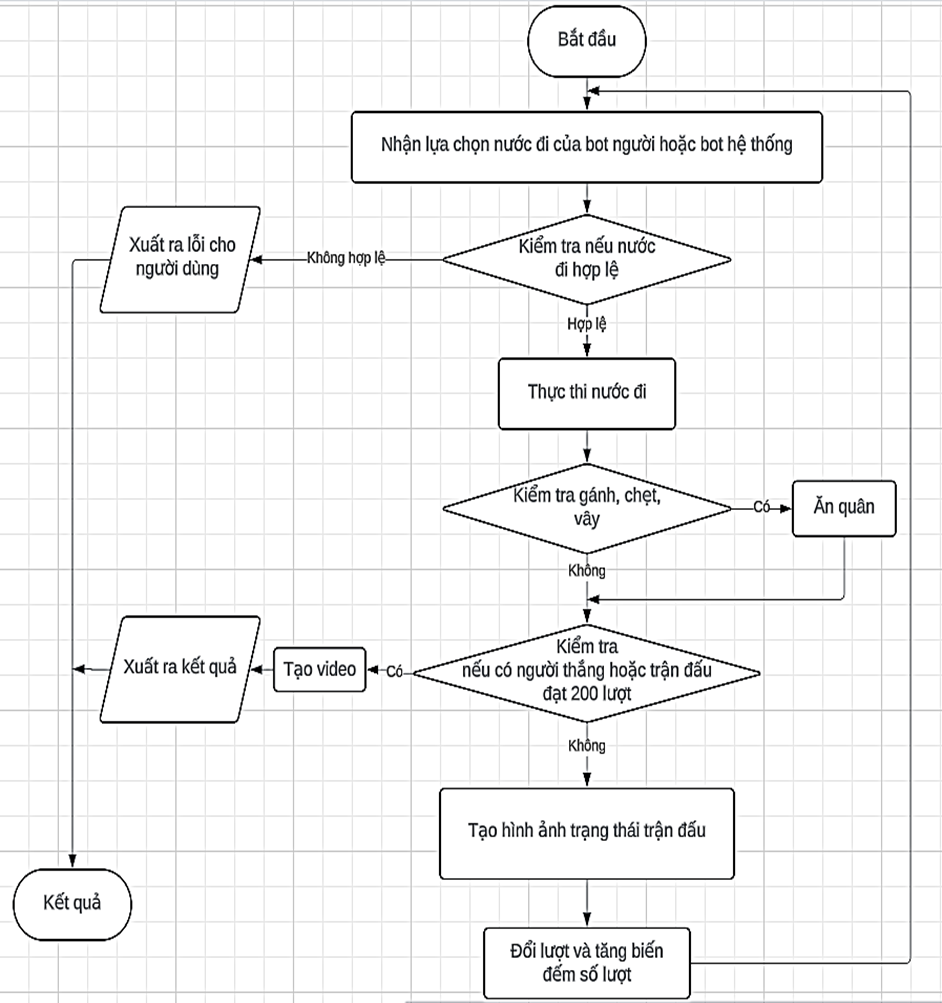
[static]/[script]: xử lý tương tác website

[templates]: lưu các file HTML quy định giao diện website

[trainAI]/Master.py: bot hệ thống (đưa ra nước đi tối ưu mỗi lượt)

[trainAI]/[source\_code]/pos\_point.txt: nguồn của bot hệ thống

### 1.2.4.2. Sơ đồ vận hành ván đấu



### 1.2.4.3 Các thuật toán chính

#### a. Kiểm tra nước đi hợp lệ

*Input:*

move: vị trí quân cờ được chọn sau khi di chuyển → *dict*

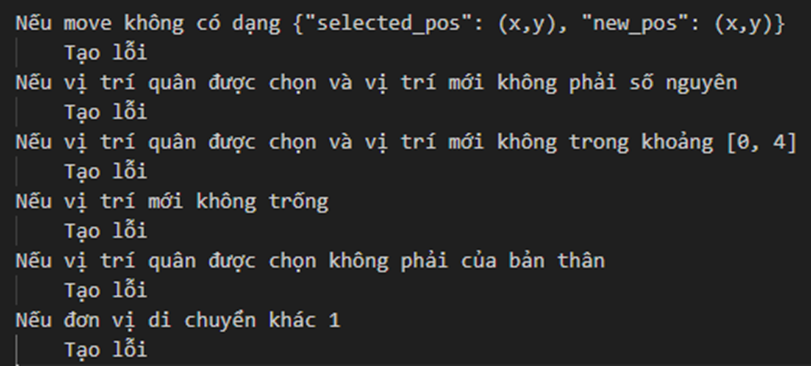
current\_turn: màu cờ của lượt hiện tại (1:🔵/ -1:🔴) → *int*

game\_state["board"]: ma trận bàn cờ → *list*

*Output:*

None

*Thuật toán:*

**

*Funtion Raise\_exception*



#### b. Kiểm tra Gánh/ Chẹt:

*Input*:

move: vị trí quân cờ được chọn sau khi di chuyển → *dict*

opp\_pos: tổng hợp vị trí quân cờ đối thủ → *list*

side: màu cờ của bản thân → *int*

opp\_side: màu cờ của đối thủ → *int*

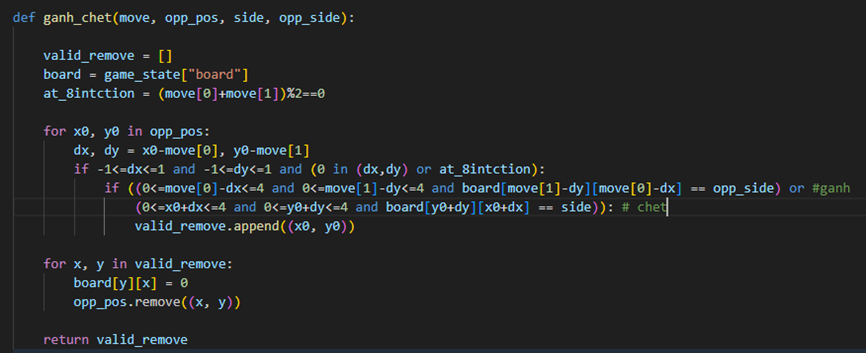
*Output*:

valid\_remove: vị trí quân cờ bị ăn → *list*

*Thuật toán:*

**

*Funtion ganh\_chet*

**

#### c. Kiểm tra Vây:

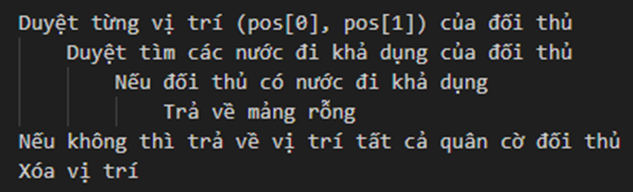
*Input*:

opp\_pos: tổng hợp vị trí quân cờ đối thủ → *list*

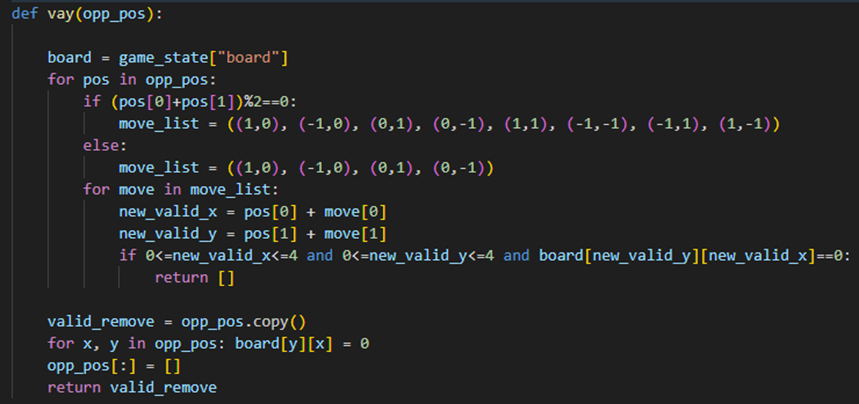
*Output*:

valid\_remove: vị trí quân cờ bị ăn → *list*

*Thuật toán:*

**

*Funtion vay*



#### d. Thuật toán AI đánh cờ (MiniMax): tìm nước đi tốt nhất – Dành cho bot hệ thống

*Input*:

player: trạng thái bàn cờ từ file game\_manager.py → *class*

*Output*:

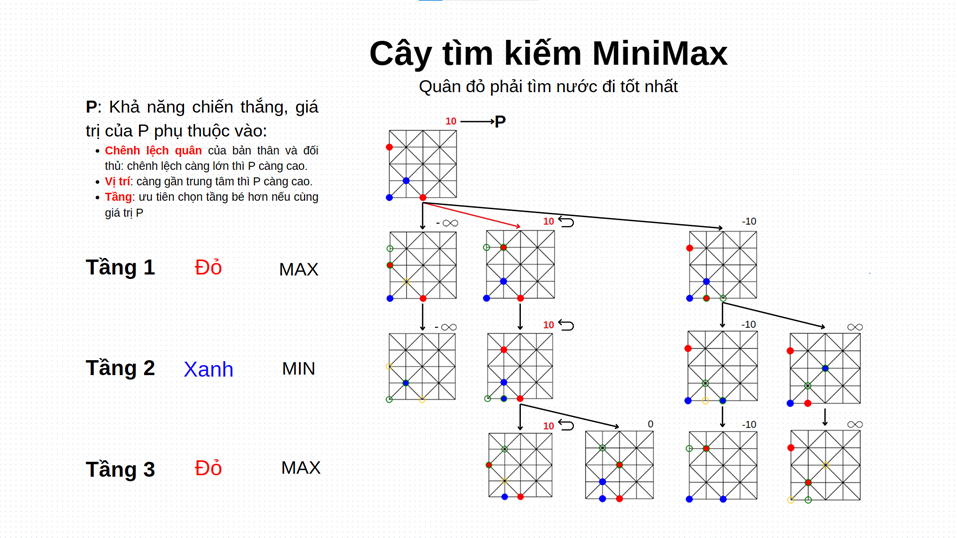
{"selected\_pos": (x, y), "new\_pos": (x, y)}: nước đi → *dict*

*Thuật toán:*

Định nghĩa: nước đi tốt nhất của bản thân là nước đi tối đa hóa được khả năng chiến thắng của bản thân đồng thời hạn chế được tối đa khả năng chiến thắng của đối thủ.

Khả năng chiến thắng (P) được tính dựa tổng số lượng quân của bản thân cộng cho giá trị mức chiến lược vị trí quân cờ bản thân trừ cho lượng quân cờ của đối thủ trừ cho giá trị mức độ chiến lược vị trí quân cờ đối thủ. Vậy P càng lớn thì khả năng chiến thắng của bản thân càng lớn, ngược lại nếu P càng nhỏ thì khả năng chiến thắng của đối thủ càng lớn.

Ta có cây trò trường hợp thể hiện các biến cố của trạng thái bàn cờ ở lượt của quân xanh:



Vì đây là lượt của quân xanh nên P càng lớn thì xanh càng có lợi.

Ta tính được P của tầng đáy từ biến thiên trạng thái bàn cờ.

Lại áp dụng định nghĩa có được nước đi tốt nhất của quân xanh là nước đi tối đa hóa được P và nước đi tốt nhất của quân đỏ là được đi hạn chế tối đa được P. Vậy:

P hàng 3 = min hàng 4

P hàng 2 = max hàng 3

P hàng 1 = min hàng 2

P hàng 0 = max hàng 1

→ Nước đi tốt nhất: mũi tên màu xanh

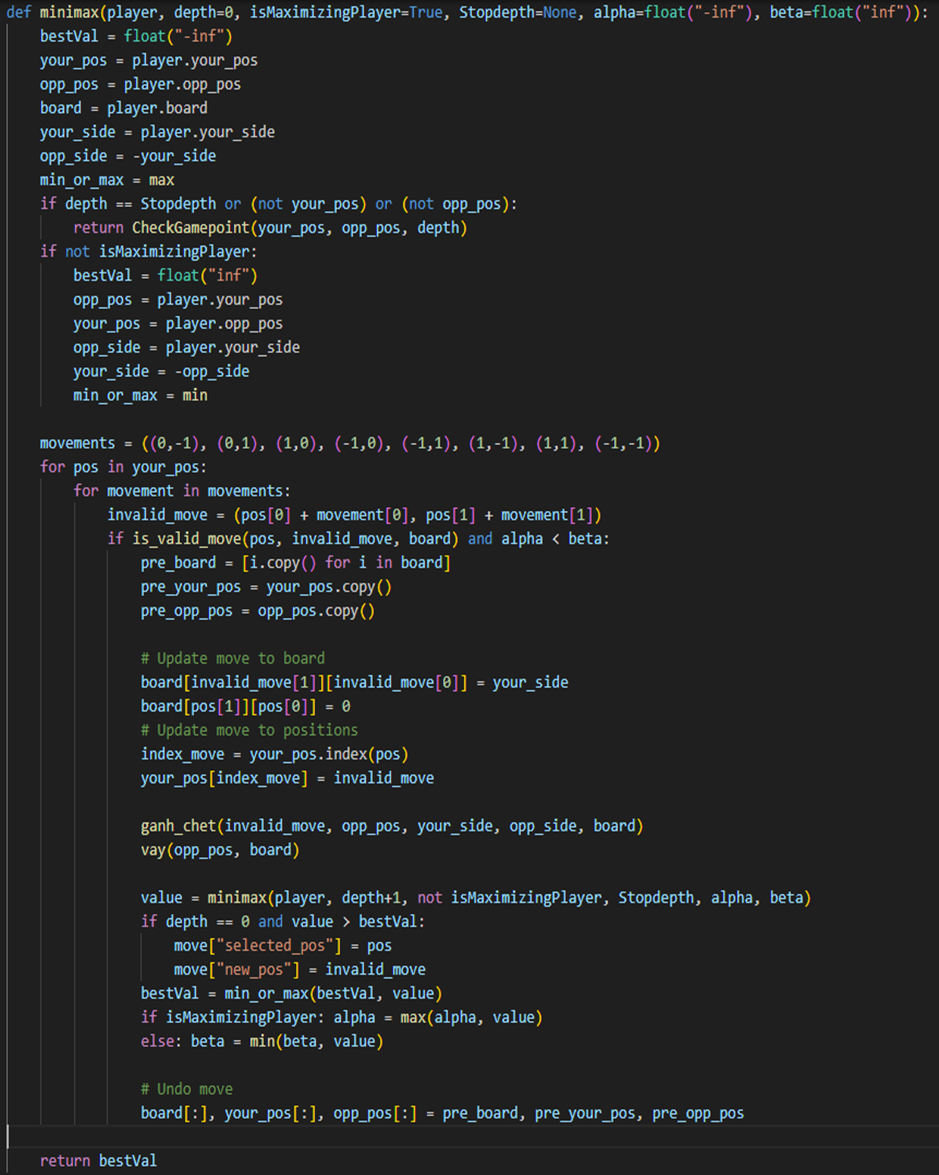
Vì số lượng biến cố quá lớn nên chương trình chỉ tính 6 tầng đầu.

Link tham khảo: [Thuật toán Minimax (AI trong Game) (viblo.asia)](https://viblo.asia/p/thuat-toan-minimax-ai-trong-game-APqzeaVVzVe)

*Funtion minimax*

*Cải tiến thuật toán:*

* **Thuật toán cắt tỉa alpha-beta:** ý tưởng chính ở đây là giảm số lượng nút trong cây trò chơi bằng cách cắt chúng đi.
* **Bitboard**: Bitboards là một cách đại diện cho bảng trò chơi bằng cách sử dụng bit. Trong trường hợp này, mỗi bit đại diện cho một vị trí trên bảng. Hơn nữa, cả hai người chơi đều có bảng riêng.
* **Bảng chuyển vị:** Bảng chuyển vị được sử dụng để lưu trữ kết quả của các bảng đã được phân tích. Điều này ngăn thuật toán đánh giá cùng một bảng nhiều lần và đọc nó từ bộ nhớ.

**

#### 3.3.3 Kiểm tra kết quả trận đấu

*Input:*

positions[1]: tổng hợp vị trí quân cờ xanh → *list*

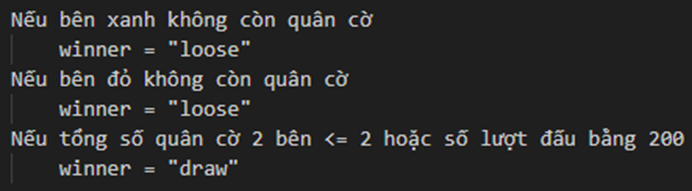
positions[2]: tổng hợp vị trí quân cờ đỏ → *list*

move\_counter: số lượt đấu → *int*

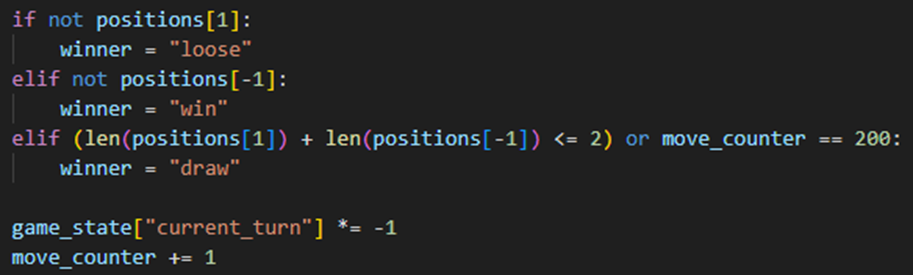
*Output:*

winner: kết quả trận đấu → *str*: (“win”/ “loose”/ “draw”)

*Thuật toán:*

**

*Funtion*



***1.3. Kết luận:***

* Những chế độ đã làm được so với yêu cầu đề ra:
* Human\_human: 100%
* Bot\_bot: 100%
* Bot\_human: 100%
* Trong mỗi chế độ, đều được chia nhiều cấp độ từ dễ tới khó, có nhận xét nước đi của người chơi và bot (Tệ/ Bình thường/ Tốt/ Thiên tài), điều này sẽ làm tăng trãi nghiệm học của người chơi.
* Ưu điểm:
* Việc sử dụng website khiến COGANH không bị cản trở bởi khoảng cách địa lý và các thao tác cài đặt rườm rà.
* Giao diện thân thiện, trẻ trung, dễ sử dụng.
* Sử dụng cờ gánh - một loại cờ tuy đơn giản nhưng lại mang tính chiến thuật cao.
* COGANH là sản phẩm có ý tưởng hoàn toàn mới và có cơ hội phát triển cao, ít cạnh tranh.
* Dễ tiếp cận mọi đối tượng người dùng, từ tiểu học, học sinh trung học cơ sở đến học sinh trung học phổ thông, sinh viên, nhà nghiên cứu.
* Nhược điểm:
* Hiện tại vẫn có nhiều người chưa biết tới sự tồn tại của cờ gánh nên cần kế hoạch giới thiệu rõ ràng
* Vì là ý tưởng hoàn toàn mới nên sẽ gặp khó khăn trong việc định hướng cũng như thiếu các tài liệu hỗ trợ.
* Người dùng chỉ được sử dụng một ngôn ngữ lập trình là Python

**2. Đánh giá sản phẩm**

***2.1. Tiềm năng ứng dụng***

- COGANH có thể là học liệu lập trình phù hợp với mọi đối tượng học sinh. Học sinh tiểu học hoặc THCS có thể chọn chế độ người đấu với bot hệ thống. Học sinh THPT hoặc sinh viên có thể tham gia lập trình bot để đấu với bot của hệ thống hay bot của người chơi khác.

- Câu lạc bộ hoặc Tổ bộ môn Tin Học hoặc các Trung tâm dạy lập trình có thể sử dụng COGANH để tổ chức ngoại khóa dưới dạng các cuộc thi.

- Chuyên viên nghiên cứu có thể áp dụng cờ gánh để nghiên cứu thuật toán AI mới.

**3. Yêu cầu đối với cơ sở hạ tầng cần thiết để triển khai ứng dụng sản phẩm:**

* Wifi
* Ổ điện

**4. Sản phẩm được phát triển ước tính trong khoảng thời gian:**

Số tháng: 6 *(Từ tháng 11/2023 đến tháng 5/2024)*

**5. Hướng dẫn sử dụng sản phẩm**

### 5.1 Hướng dẫn luật chơi cờ gánh trong dân gian

[Video hướng dẫn luật chơi cờ gánh](https://1drv.ms/v/s!Akgf6UPdX_vghJMXnFHCj1nBFSvOyw) (Nguồn: internet)

### 5.2 Hướng dẫn chung về các chế độ chơi trên website

- **Mô tả:** Một ván cờ được chia thành nhiều lượt đi, mỗi bên đi một lượt. Sau mỗi lượt chơi, trạng thái bàn cờ sẽ thay đổi, mỗi bên sẽ tự cập nhật trạng thái mới để suy nghĩ nước cờ tiếp theo

- **Các chế độ chơi:**

* **Chế độ người đấu với người (không lập trình) – human\_human**:

Phù hợp cho hai người chơi không phân biệt khoảng cách địa lý.

* **Chế độ người đấu với bot của hệ thống (không lập trình) – human\_bot**:

Phù hợp cho những người muốn học chiến thuật.

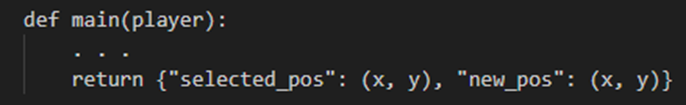
* **Chế độ tạo chương trình (bot) đấu với bot của hệ thống hoặc bot của người chơi khác – bot\_bot:**

Phù hợp với những người đã biết về lập trình Python từ cơ bản đến nâng cao. Tại một lượt chơi, bot người dùng sẽ nhận Input từ hệ thống, từ đó chạy các mã lệnh để trả về 1 nước đi của lượt đấu đó.

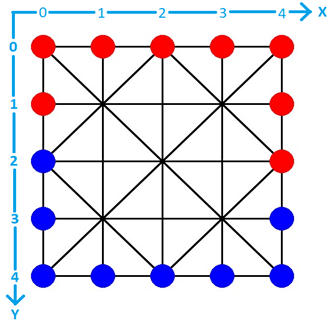
***Mỗi chế độ chơi đều có nhiều cấp độ từ dễ tới khó.***

### 5.3 Hướng dẫn chi tiết về cách viết 1 bot

- Hàm thực thi chính của chương trình:



- Input → *class* player



● player.your\_pos: vị trí tất cả quân cờ của bản thân [(x, y), . . .]

● player.opp\_pos: vị trí tất cả quân cờ của đối thủ [(x, y), . . .]

● player.your\_side: màu quân cờ của bản thân (1:🔵)

● player.board: bàn cờ (-1:🔴 / 1:🔵 / 0:∅)

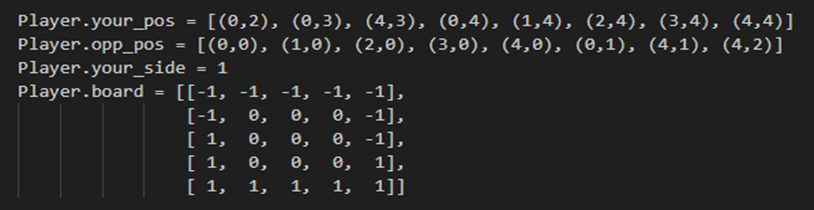
- Ràng buộc

● 0 ≤ x, y ≤ 4

● player.your\_side = 1

● {j for i in player.board for j in i} = {0, 1, -1}

- Khởi đầu ván đấu:



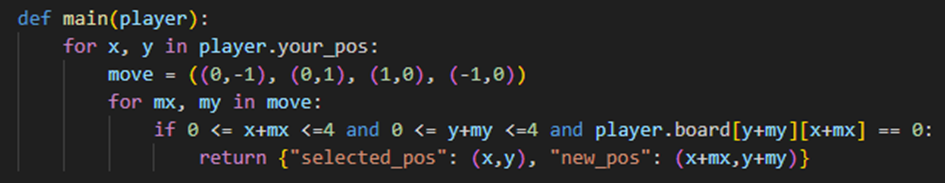
- Output → *dictionary*

● selected\_pos: vị trí của quân cờ được chọn → *tuple*(x, y)

● new\_pos: vị trí sau khi di chuyển của quân cờ đó → *tuple*(x, y)

Ví dụ: {"selected\_pos": (1, 0), "new\_pos": (1, 1)}

- Code mẫu: thực hiện nước đi hợp lệ đầu tiên tìm thấy



**6. Tự đánh giá về những mặt còn tồn tại chưa giải quyết được của sản phẩm để khắc phục.**

- Xử lý mã độc trong bot được người chơi nạp vào hệ thống.

- Tối ưu hiệu suất tạo mô phỏng các ván cờ.

**KẾT LUẬN**

**1. Hướng phát triển của sản phẩm trong tương lai**

- Cờ gánh có thể là học liệu lập trình phù hợp với mọi đối tượng học sinh. Học sinh tiểu học hoặc THCS có thể chọn chế độ người đấu với bot hệ thống. Học sinh THPT hoặc sinh viên có thể tham gia lập trình bot để đấu với bot của hệ thống hay bot của người chơi khác.

- Tổ bộ môn tin học, các câu lạc bộ hoặc các Trung tâm dạy lập trình có thể sử dụng Cờ gánh để tổ chức ngoại khóa dưới dạng các cuộc thi.

- Trong tương lai gần, nhóm thực hiện sẽ tiếp tục mở rộng chức năng của hệ thống theo hướng Huấn luyện cho Người chơi và bot để ngày càng nâng cao khả năng xây dựng chiến thuật chơi cờ. Ngoài ra, một số công việc cần được chú ý khi đưa hệ thống vào triển khai là việc phát hiện và xử lý mã độc trong bot được người chơi nạp vào hệ thống

- Trong tương lai xa, nhóm thực hiện mong muốn phát triển hệ thống tương thích với nhiều ngôn ngữ lập trình phổ biến để tạo thêm cơ hội trải nghiệm cho người học

**2. Nguyện vọng trong tương lai**

**-** Đưa COGANH trở thành 1 công cụ chính thức trong việc dạy học lập trình tại cái trường học.

- Mở nhiều giải đấu giữa bot của người chơi với nhau, từ đó có thể tìm ra được những thuật toán mới, chưa từng được biết tới.

- Có thêm kinh phí để duy trì website và đưa COGANH trở nên phổ biến tới mọi đối tượng người học.

**TÀI LIỆU THAM KHẢO**

[Tổng hợp tài liệu](https://dev.to/larswaechter/improving-minimax-performance-1924)

[Video hướng dẫn luật chơi cờ gánh](https://1drv.ms/v/s!Akgf6UPdX_vghJMXnFHCj1nBFSvOyw): (Nguồn: internet)

*ngày 31 tháng 5 năm 2024*

**Chữ ký của tác giả/nhóm tác giả**



Thông Minh Quân