

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

BÁO CÁO MÔN HỌC

HỌC PHẦN: NHẬP MÔN TRÍ TUỆ NHÂN TẠO

# Đề tài :

**XÂY DỰNG GAME CỜ VUA SỬ DỤNG TRÍ TUỆ NHÂN TẠO**

## Sinh viên thực hiện: Lớp: 156725

|  |  |
| --- | --- |
| Lê Nguyên Khải | 20225727 |
| Phạm Lê Quang Minh | 20225887 |
| Nguyễn Trần Hồng Chiến | 20225602 |
| Nguyễn Thành Long | 20225736 |

***Hà Nội, tháng 6 năm 2025***

**Mục lục**

1. [Giới thiệu và mô tả về bài toán 1](#_bookmark0)
   1. [Giới thiệu trò chơi cờ vua 1](#_bookmark1)
   2. [Bài toán cần giải quyết 1](#_bookmark2)
   3. [Mục tiêu, kết quả mong muốn 1](#_bookmark3)
2. [Phương pháp áp dụng 1](#_bookmark4)
3. [Nội dung phương pháp 2](#_bookmark5)
   1. [Thuật toán Minimax 2](#_bookmark6)
   2. [Thuật toán cắt tỉa Alpha - Beta 3](#_bookmark7)
   3. Thuật toán Greedy
   4. [Các hàm heuristic áp dụng 4](#_bookmark8)
      1. [Tổng điểm từng quân cờ 4](#_bookmark9)
      2. [Điểm theo vị trí của từng quân 4](#_bookmark10)
4. [Thực hiện, xây dựng chương trình 6](#_bookmark11)
   1. [Mô tả chương trình 6](#_bookmark12)
   2. [Giao diện chương trình 6](#_bookmark13)
   3. [Tìm hiểu sơ lược về chương trình 8](#_bookmark14)
   4. [Nhận xét 10](#_bookmark15)
   5. [Hướng phát triển 11](#_bookmark16)
5. [Thông tin khác 11](#_bookmark17)
   1. [Phân công công việc 11](#_bookmark18)
   2. [Tham khảo 11](#_bookmark19)

# Giới thiệu và mô tả về bài toán

## Giới thiệu trò chơi cờ vua

Cờ vua là trò chơi 2 người, thể loại **zero-sum**.

Một người chơi sẽ cầm quân cờ màu trắng và người còn lại sẽ cầm quân cờ đen. Trò chơi sử dụng một bàn cờ hình vuông chia thành 64 ô vuông nhỏ hơn với 8 hàng ngang và 8 hàng dọc. Mỗi người chơi sẽ bắt đầu với 16 quân cờ, bao gồm 8 con tốt, 2 mã, 2 tượng, 2 xe, 1 hậu và 1 vua

**Nguồn gốc** : Hình thức chơi hiện tại của cờ vua bắt đầu xuất hiện ở Nam Âu ở nửa sau của thế kỷ 15. Ngày nay, cờ vua là một trong những trò chơi phổ biến nhất thế giới, được hàng triệu người trên toàn thế giới chơi tại nhà, ở câu lạc bộ, trên trực tuyến

Mặc dù chỉ có 64 ô và 32 quân cờ trên bàn cờ nhưng số lượng nước đi có thể được thì còn vượt xa cả số lượng các nguyên tử có trong vũ trụ.

## Bài toán cần giải quyết

Xây dựng game cờ vua có sử dụng giải thuật trí tuệ nhân tạo

## Mục tiêu, kết quả mong muốn

* Trò chơi có chế độ chơi (người vs người) và (người vs máy), trong đó máy có khả năng chơi cờ thông minh
* Áp dụng các thuật toán trong học phần trí tuệ nhân tạo vào trò chơi
* Phần mềm chạy ổn định và đưa ra quyết định các nước đi nhanh chóng

# Phương pháp áp dụng

### Các phương pháp áp dụng:

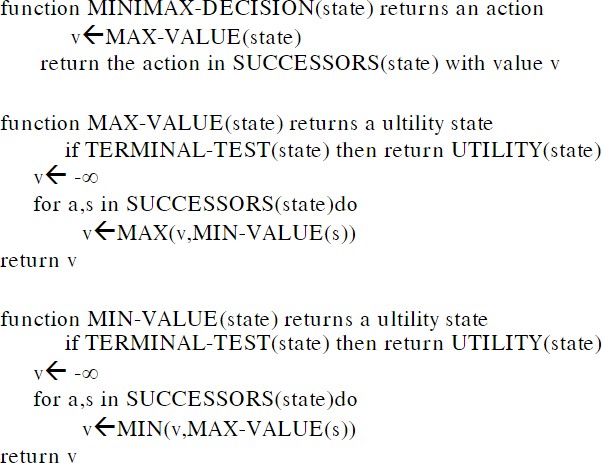
* + Các thuật toán tìm kiếm nước đi : Alpha-beta prunning và Minimax
  + Các hàm đánh giá heuristic
* **Lý do lựa chọn :** Bởi vì lựa chọn hay đoán các nước đi tốt ở trạng thái hiện tại thì rất khó đạt được, chương trình cờ vua sẽ dựa trên việc **tìm kiếm** các nước đi và **đánh giá** bàn cờ sau khi thực hiện nước đi

# Nội dung phương pháp

### Thuật toán Minimax

Minimax là giải thuật là một thuật toán đệ quy lựa chọn bước đi kế tiếp trong một trò chơi có hai người bằng cách so sánh các Node trên cây trò chơi

Hai đối thủ trong trò chơi được gọi là MIN và MAX luân phiên thay thế nhau đi. MAX đại diện cho người quyết dành thắng lợi và cố gắng tối đa hóa ưu thế của mình, ngược lại người chơi đại diện cho MIN lại cố gắng giảm điểm số của MAX và cố gắng làm cho điểm số của mình càng âm càng tốt. Giả thiết đưa ra MIN và MAX có kiến thức như nhau về không gian trạng thái trò chơi và cả hai đối thủ đều cố gắng như nhau.



***Thuật toán Minimax***

* **Ưu điểm** : Tìm kiếm được mọi nước đi tiếp theo sau đó lựa chọn nước đi tốt nhất, vì giải thuật có tính chất vét cạn nên không bỏ sót trạng thái.
* **Khuyết điểm** : Đối với các trò chơi có không gian trạng thái lớn như cờ vua việc chỉ áp dụng giải thuật Minimax có lẽ không còn hiệu quả nữa do sự bùng nổ tổ hợp quá lớn. Giải thuật áp dụng nguyên lý vét cạn không tận dụng được thông tin của trạng thái hiện tại để lựa chọn nước đi, vì duyệt hết các trạng thái nên tốn thời gian.

### Thuật toán cắt tỉa Alpha - Beta

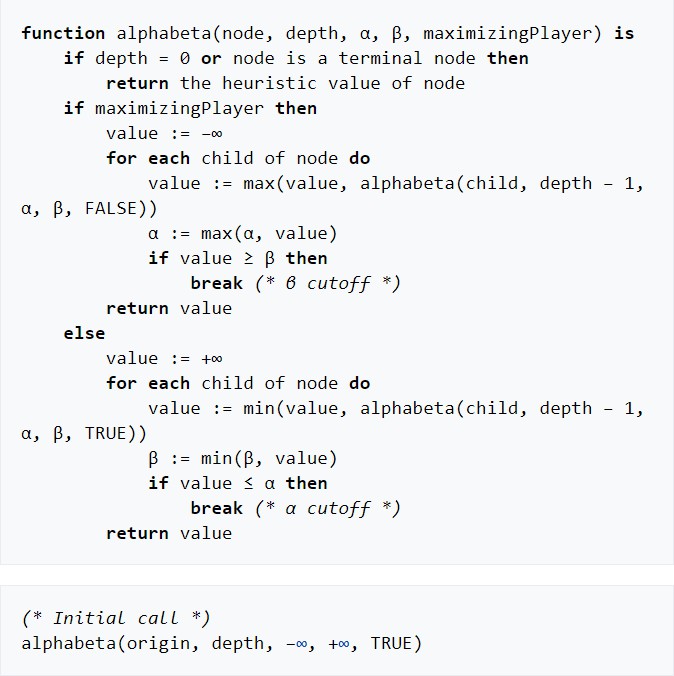
Thuật toán alpha-beta ra đời và dần thay thế thuật toán Minimax. Một trong số những ưu điểm đáng chú ý của alpha-beta đó là tinh giản được quá trình tìm nghiệm, loại trừ được một số trường hợp không thích hợp phát sinh nhưng vẫn đảm bảo không ảnh hưởng đến kết quả sau cùng.

Chiến lược cắt tỉa:

* Nút Max có một giá trị 𝑎𝑙𝑝ℎ𝑎 (luôn tăng), nút Min có một giá trị 𝑏𝑒𝑡𝑎 (luôn giảm).
* Khi chưa có 𝑎𝑙𝑝ℎ𝑎 và 𝑏𝑒𝑡𝑎 xác định thì thực hiện tìm kiếm sâu (depth-first) để xác định được 𝑎𝑙𝑝ℎ𝑎, 𝑏𝑒𝑡𝑎 và truyền ngược lên các nút cha.

Phương pháp cắt tỉa alpha-beta:

* Nếu một nhánh tìm kiếm nào đó không thể cải thiện đối với giá trị (hàm tiện ích) mà chúng ta đã có, thì không cần xét đến nhánh tìm kiếm đó nữa!
* Việc cắt tỉa các nhánh tìm kiếm (“tồi”) không ảnh hưởng đến kết quả cuối cùng



***Thuật toán Alpha-Beta pruning***

### Thuật toán Greedy

Thuật toán Greedy là một trong những thuật toán đơn giản nhất trong trí tuệ nhân tạo. Đây là phương pháp tìm kiếm theo chiến lược "tham lam", luôn chọn lựa giải pháp tốt nhất tại mỗi bước mà không quan tâm đến hậu quả trong tương lai.**Chiến lược tìm kiếm:**

* Tại mỗi bước, thuật toán sẽ xem xét tất cả các nước đi có thể
* Đánh giá điểm số cho từng nước đi thông qua hàm đánh giá
* Chọn nước đi có điểm số cao nhất để thực hiện

### Các hàm heuristic áp dụng

### Tổng điểm từng quân cờ

Giá trị này có ảnh hưởng nhất trong các hàm đánh giá. Mỗi quân cờ sẽ được gắn điểm riêng

Trong chương trình cờ vua hiện tại, sử dụng bảng điểm sau:

|  |  |
| --- | --- |
| **Quân cờ** | **Giá trị** |
| Tốt | 100 |
| Mã | 320 |
| Tượng | 330 |
| Xe | 500 |
| Hậu | 900 |
| **Vua** | **20000** |

*Source:* [Simplified Evaluation Function - Chessprogramming wiki](https://www.chessprogramming.org/Simplified_Evaluation_Function)

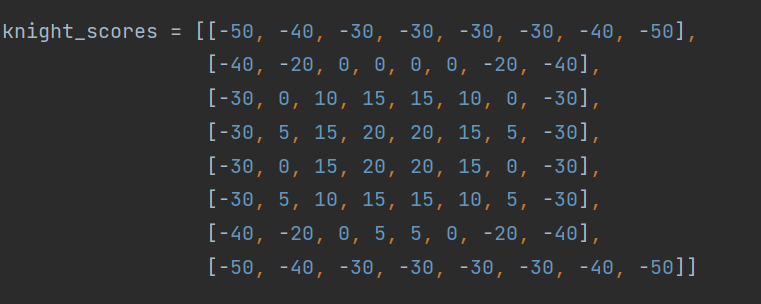
Điểm tối đa của 1 bên dựa vào tổng điểm từng quân cờ là:

20000 + 9 ∗ 900 + 2 ∗ 500 + 4 ∗ 300 = 30300

### Điểm theo vị trí của từng quân

Vì mỗi quân cờ sẽ phát huy hết điểm mạnh dựa vào vị trí của nó trên bàn cờ. Do đó mỗi quân cờ sẽ có một ma trận điểm thưởng 8 × 8

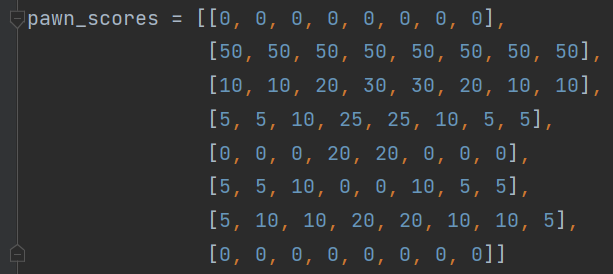
* + - * Ma trận điểm của quân mã:



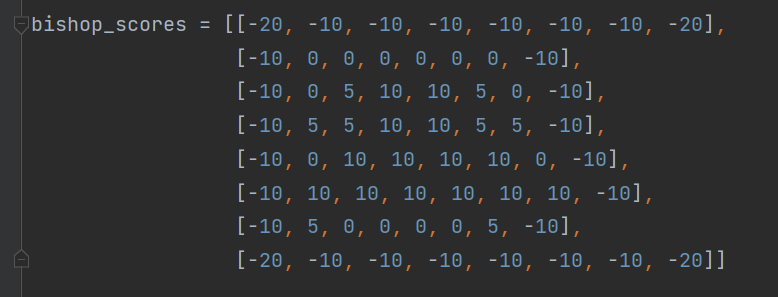
*Source:* [Simplified Evaluation Function - Chessprogramming wiki](https://www.chessprogramming.org/Simplified_Evaluation_Function)

Ma trận được tính dựa trên tính chất của quân mã. Mã đi theo hình chữ L nên rất yếu khi ở biên với góc bàn cờ và ngược lại càng mạnh khi ở gần trung tâm

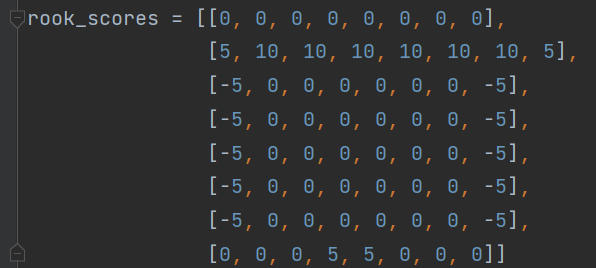
* + - * Ma trận điểm của quân tốt:



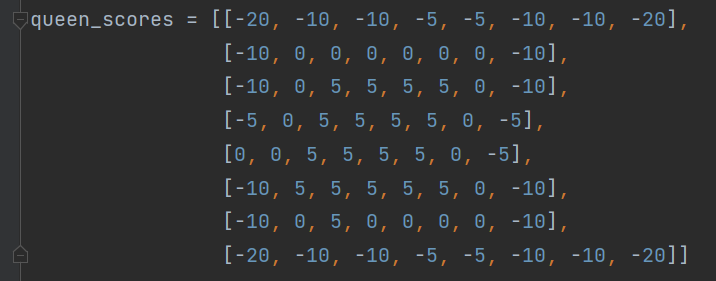
* + - * Ma trận điểm của quân tượng:



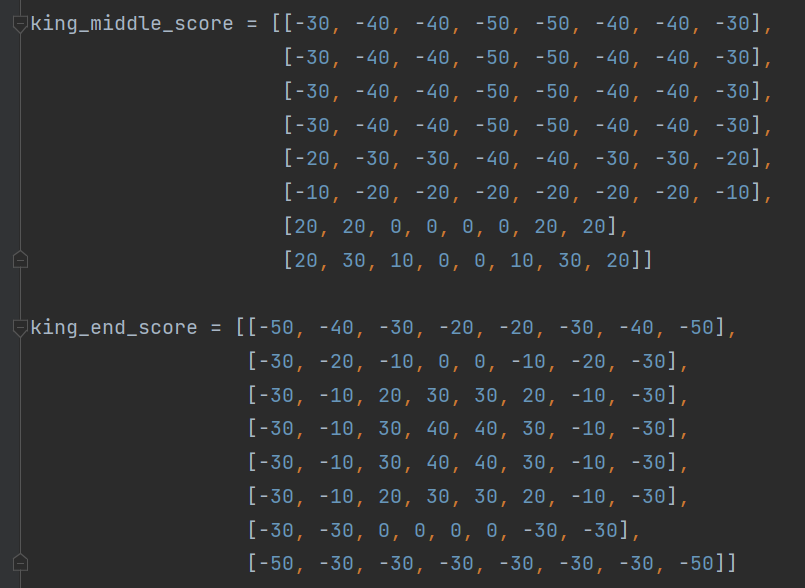
* + - * Ma trận điểm của quân xe:



* + - * Ma trận điểm của quân hậu:



Ma trận điểm của quân vua:



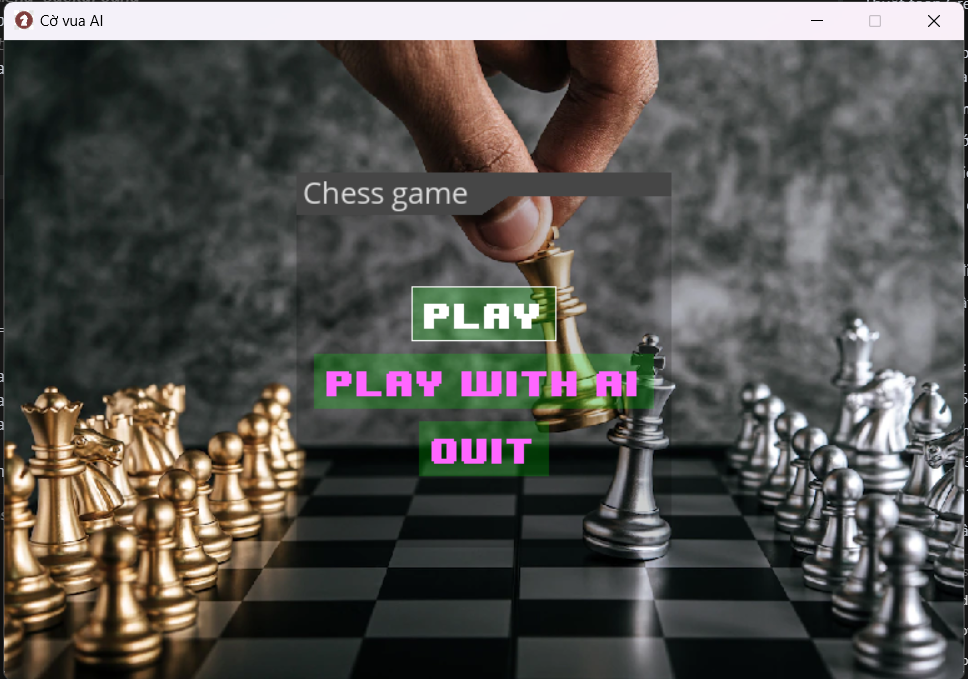
## Thực hiện, xây dựng chương trình

### Mô tả chương trình

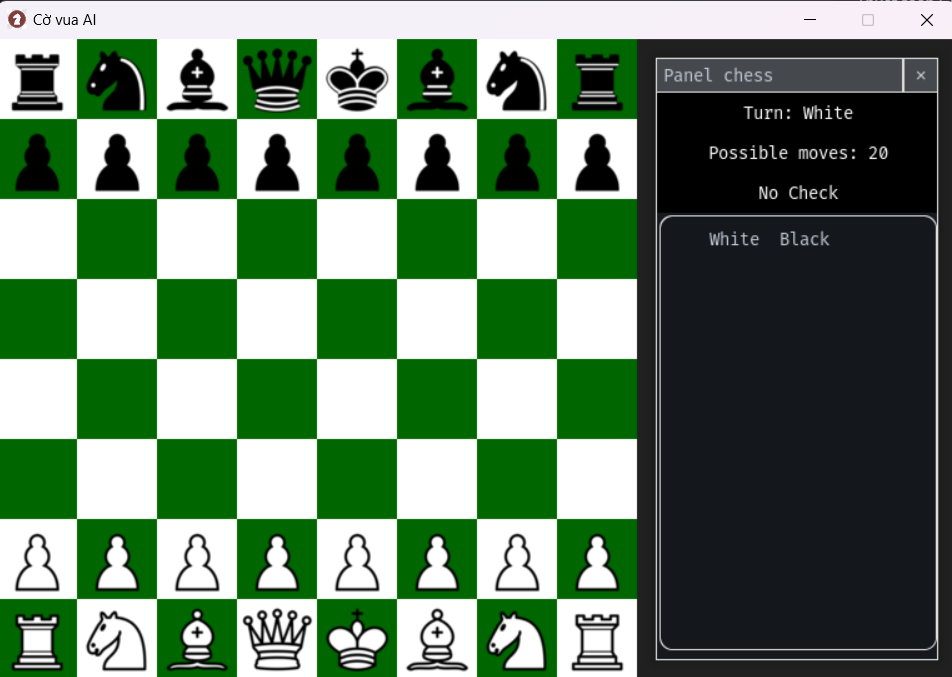
* Chương trình được xây dựng trên môi trường Window
* Ngôn ngữ lập trình sử dụng : Python
* Môi trường phát triển: PyCharm Community Edition
* Hướng dẫn cách chạy chương trình ở file README.md
* Các thư viện sử dụng: pygame, pygame-gui, pygame-menu

### Giao diện chương trình

Menu trước khi vào game



Giao diện ở chế độ người với người:



Giao diện ở chế độ người với máy:



Ở bên phải của bàn cờ có các bảng trạng thái:

|  |  |
| --- | --- |
|  |  |
| *Bảng 1* | *Bảng 2* |

*Bảng 1*: Hiển thị các thông tin của bên đang có quyền đi: Lượt đi, các nước đi có thể, có bị chiếu tướng hay không, và bảng hiển thị các nước đi của 2 bên tính từ lúc bắt đầu trận

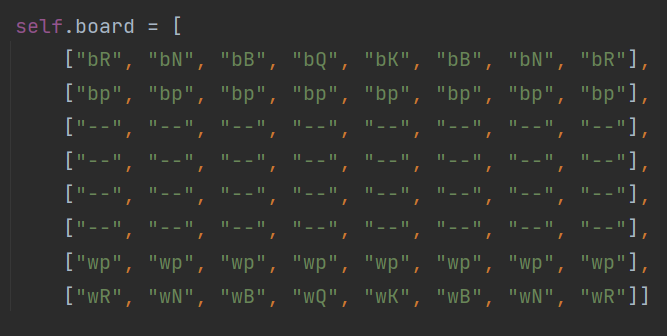
Bảng 2: Hiển thị thông tin liên quan đến đến máy (AI): giải thuật, độ sâu tìm kiếm, điểm sau khi thực hiện tìm kiếm, và bảng chứa thông tin về số node tìm được, nhánh được cắt bỏ, số node lá (hay là số lần gọi hàm đánh giá) và thời gian chạy, ở bảng này có thể chọn các thuật toán khi chạy game.

### Các phím tắt hỗ trợ trong quá trình chơi game:

Phím Z : Để undo lại nước đi Phím R: Để reset lại game

### Tìm hiểu sơ lược về chương trình

* **Biểu diễn bàn cờ (Board Representation)** :
* Lưu trị vị trí các quân cờ để phục vụ cho việc **tìm kiếm, đánh giá** và vẽ giao diện
* Sử dụng mảng 8 × 8 với mỗi phần tử trong mảng là kiểu xâu với 2 kí tự đại diện mỗi quân cờ.

Nhận xét:

Cách biểu diễn trực quan do đó giảm bớt độ phức tạp khi xây dựng chương trình Nhược điểm: Hiệu năng kém so với nhiều phương pháp biểu diễn khác như **bitboards**

### Sinh các nước đi hợp lệ (Legal Move Generation):

* Các nước đi được gọi là hợp lệ nếu nước đi đó không để tướng bị chiếu
* Là 1 phần quan trọng việc xây dựng chương trình, phụ thuộc lớn vào việc **biểu diễn bàn cờ**.

Tham khảo: [Generating Legal Chess Moves Efficiently • Peter Ellis Jones](https://peterellisjones.com/posts/generating-legal-chess-moves-efficiently/) Ý tưởng khi sinh các nước đi hợp lệ:

* Kiểm tra xem vua có bị chiếu không?
* Nếu vua bị chiếu thì việc sinh ra các nước đi sẽ như thế nào?

### Các chiến lược tìm kiếm

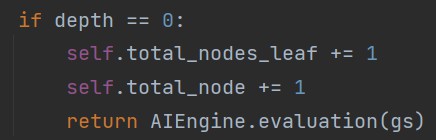
Như đã giới thiệu ở phần trên, chương trình sử dụng 3 thuật toán tìm kiếm là Minimax ,Alpha-Beta Prunning và Greedy

Sau đây là phương thức gọi hàm sử dụng Minimax và Alpha-Beta Prunning (nằm trong file

./engnie/AIEngine)





Nếu depth == 0 khi đó gọi trả về giá trị của hàm đánh giá

Nếu depth != 0 thì khi đó sinh các nước đi hợp lệ từ bàn cờ, duyệt các nước đi hợp lệ và gọi đệ quy:

### Text Description automatically generatedCác hàm đánh giá trạng thái bàn cờ

Như đã đề cập ở phần trên, chương trình sử dụng 2 hàm đánh giá cơ bản là: **Tổng điểm từng quân cờ** và **Điểm theo vị trí của từng quân**

Phương thức gọi hàm đánh giá được sử dụng trong chương trình:

|  |  |
| --- | --- |
| Text  Description automatically generated |  |
| ***Tổng điểm từng quân cờ*** | ***Điểm theo vị trí của từng quân*** |

### Kiểm thử và Nhận xét

* **Kiểm thử :**

Khi đặt độ sâu là 4, AI mất trung bình 4s để thực hiện nước đi Dựa vào bảng trên, thời gian trung bình là 12000 nodes/s



Khi đặt độ sâu là 3, AI mất trung bình 0.2s để thực hiện nước đi

* **Kết quả** : Một chương trình trò chơi cờ vua ứng dụng trí tuệ nhân tạo trong việc tính toán các nước đi của máy, dễ sử dụng và gần gũi với người dùng

### Hạn chế :

* + Máy thực hiện các nước đi chậm chạp khi tăng độ sâu tìm kiếm lên 5
  + Giao diện chưa được hấp dẫn
  + Các hàm đánh giá còn đơn giản nên chưa đánh giá đủ tốt thế trận



khi sử dụng thuật toán minimax, đặt độ sâu là 4, thời gian trung bình của ai suy nghĩ là 15 giây, thời gian này là quá lâu, do minimax phải duyệt hết tất cả các trạng thái của bàn cờ với độ sâu là 4.



Khi sử dụng thuật toán greedy, thời gian chạy gần như bằng 0, do greedy chỉ xem xét nước đi tốt nhất ở thời điểm hiện tại, chỉ ưu tiên ăn quân để tăng điểm

### Hướng phát triển

* Cải thiện hiệu năng của chương trình để máy thực hiện nhanh ở độ sâu lớn hơn
* Tìm hiểu thêm các hàm đánh giá để cải thiện chất lượng chơi cờ của máy
* Cải tiến phiên bản cờ vua mới với đầy đủ các chức năng như: Vẽ các hiệu ứng khi di chuyển quân cờ, lựa chọn độ khó của máy, thay đổi màu nền bàn cờ trực tiếp,…

## Thông tin khác

### Phân công công việc

* Lê Nguyên Khải : Nghiên cứu thuật toán, nghiên cứu thư viện, tìm hiểu các nước đi, làm slide
* Phạm Lê Quang Minh : Nghiên cứu thuật toán, nghiên cứu thư viện, kiểm thử, làm báo cáo
* Nguyễn Trần Hông Chiến :Nghiên cứu thuật toán, nghiên cứu thư viện, giao diện, làm báo cáo
* Nguyễn Thành Long :Nghiên cứu thuật toán, nghiên cứu thư viện, tìm hiểu các nước đi, làm slide

### Tham khảo

1. [Chessprogramming wiki](https://www.chessprogramming.org/Main_Page)
2. “Artificial Intelligence A Modern Approach (3rd Edition)” - Stuart J. Russell and Peter Norvig
3. [Generating Legal Chess Moves Efficiently • Peter Ellis Jones](https://peterellisjones.com/posts/generating-legal-chess-moves-efficiently/)