Gomoku

Chươnng trình game caro sử dụng Turtle trong python và engine AI dựa trên thuật toán minimax, cắt tỉa alpha-beta cùng với heuristic điểm số 4 hướng.

**Thuật toán**

* Phân tích: +Trò chơi có thể biếu diễn như một cây gốc, những nút, những lá và nhánh:

. Gốc là trạng thái ban đầu của trò chơi. Với mỗi trò chơi cụ thể thì trạng thái( ở mỗi điểm) lại đưuọc đặc trung bởi những thông số riêng

. Các nút(Node) của cây thể hiện tình trạng hiện tại của trò chơi, gồm nút cha và nút con.

. Các nhánh nối giữa các nút thể hiện nước đi, tức là cho biết từ một tình huống của trò chơi chuyển sang tình huống khác thông qua chỉ một nước đi nào đó.

. Các lá hay còn gọi là nút lá, thể hiện thời điểm kết thúc khi mà kết quả của trờ chơi đã rõ.

. Độ sâu của cây: Số tầng của cây

* + Mỗi vị trí kết thúc trò chơi(nút lá) sẽ gán một trọng số. vd: 1 win, 0 hòa, -1 lose. Tại mỗi nút cũng có một trọng số tương ứng được xác định bằng một cách nào đó. Dựa vào cây trò chơi này, ta có thể tìm nước đi tốt để giành chiến thắng.
  + Cứ sau mỗi nước đi số ô trống sẽ giảm. Vì vậy việc tìm kiếm nước đi tiếp theo là việc tìm kiếm trong không gian các ô trống còn lại, sau mỗi lượt đi thì không gian tìm kiếm sẽ giảm dần. Chiến lược tìm kiếm nước đi là chọn 1 nút trên cây sao cho nước đi là tốt. Và để đánh giá được nút đó thì phải nhìn đến độ sâu của cây. Vì không gian tìm kiếm là quá lớn nên chúng ta giới hạn cho máy tính chỉ tìm kiếm ở một độ sâu nhất định ==> Chương trình có độ sâu càng lớn thì càng chơi giỏi nhưng sẽ phải trả giá về mặt thời gian

1.Thuật toán Minimax

* Minimax là một thuật toán đệ quy lựa chọn bước đi kế tiếp trong một trò chơi 2 người bằng cách tính các giá trị cho các Node trên cây trò chơi sau đó tìm Node có giá trị phù hợp để đi bước tiếp theo.
* 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. Mỗi Node biểu diễn cho một trạng thái trên cây trò chơi. Node lá là Node chứa trạng thái kết thúc của trò chơi.

Giải thuật Minimax thể hiện bằng cách định trị các Node trên cây trò chơi:

* Node thuộc lớp MAX thì gán cho nó giá trị lớn nhất của con Node đó.
* Node thuộc lớp MIN thì gán cho nó giá trị nhỏ nhất của con Node đó. Từ các giá trị này người chơi sẽ lựa chọn cho mình nước đi tiếp theo hợp lý nhất.
* Nếu như đạt đến giới hạn tìm kiếm (đến tầng dưới cùng của cây tìm kiếm tức là trạng thái kết thúc của trò chơi).
* Tính giá trị của thế cờ hiện tại ứng với người chơi ở đó. Ghi nhớ kết quả.
* Nếu như mức đang xét là của người chơi cực tiểu (nút MIN), áp dụng thủ tục Minimax này cho các con của nó. Ghi nhớ kết quả nhỏ nhất.
* Nếu như mức đang xét là của người chơi cực đại (nút MAX), áp dụng thủ tục Minimax này cho các con của nó. Ghi nhớ kết quả lớn nhất.

1. Thuật toán cắt tỉa alpha-beta.

* Vì sự bùng nổ cây trò chơi trong minimax nên ta sẽ bỏ những nút không tối ưu bằng cách cắt tỉa alpha-beta.
* Tư tưởng: + Nếu một nhánh tìm kiếm nào đó không thể cải thiện với giá trị mà chúng ta đã có, thì không cần xét đến hàm đó nữa -> tiết kiệm chi phí thời gian, bộ nhớ cho cây tìm kiếm + Dùng hai cận Anpha và Beta để so sánh và loại bỏ các trường hợp sẽ không cần xét đến trong thuật toán minimax.
* Mô tả: + Anpha lưu nướcc đi tốt nhất của máy, Beta lưu giá trị tốt nhất của Người chơi
  + Nếu bất cứ khi nào anpha >= beta, thì người chơi chắc chắn sẽ chọn nước đi tốt nhất cho họ và cưỡng bức nước đi tồi hơn anpha cho máy, vì vậy mà không cần xét thêm bước nào nữa.