

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THỰC PHẨM TP.HCM



ĐỒ ÁN PHÁT TRIỂN PHẦN MỀM VÀ ỨNG DỤNG
THÔNG MINH

Đề tài:

Xây dựng Ứng dụng Game Caro

GVHD: TRẦN ANH DŨNG

Nhóm Sinh Viên Thực Hiện:

Lê Minh Nhựt - 2001180152

Trương Công Hậu - 2001180261

Nguyễn Thành Nhân - 2001180088

Tháng 07 Năm 2021

LỜI CẢM ƠN

Chúng em cũng xin gửi lời cảm ơn thầy Trần Anh Dũng, giảng viên hướng dẫn môn Lập trình ứng dụng thông minh, người đã tận tình giảng dạy cho chúng em những kiến thức cơ bản để thực hiện đề tài này dù đã cố gắng hoàn thành thật tốt đề tài nhưng chắc chắn sẽ có những thiếu sót không thể tránh khỏi. Chúng em mong sẽ nhận được sự thông cảm và sự tận tình chỉ bảo của thầy, các bạn. Nhóm chúng em xin chân thành cảm ơn!

LỜI MỞ ĐẦU

Công nghệ thông tin hiện nay đã có những bước phát triển mạnh mẽ và đạt được nhiều thành tựu. Trong các lĩnh vực công nghệ thông tin, thì lập trình Game đã có một bước nhảy vọt đáng kể về cả công nghệ lẫn nội dung. Ngày nay chiếc máy tính PC hay laptop đã rất phổ biến gắn liền với học tập ,công việc, và giải trí của con người, những giờ phút căng thẳng sẽ được giải tỏa, hay cần cảm giác thú vị, chúng ta đến với Game giải trí sẽ đáp ứng các yêu cầu đó.

Hiện nay đã có nhiều công nghệ để tạo game trên máy tính ,và ở chương trình này chúng em đã chọn Visual Studio môi trường viết với ngôn ngữ sử dụng là C#. Với các kiến thức đã học trong trường và tìm hiểu thêm một số tài liệu liên quan chúng em xây dựng chương trình game này.

Bài toán chúng em đặt ra không có nhu cầu khắt khe nhưng có đầy đủ những yêu cầu cơ bản nhất của một Game trên máy tính. Nhằm mục đích học hỏi, trao đổi kinh nghiệm và làm quen với thực tế.

Em xin chân thành cảm ơn thầy Trần Anh Dũng đã chỉ bảo chúng em tận tình trong suốt quá trình học tập để chúng em có thể hoàn thành đồ án này

Chân thành cảm ơn!

MỤC LỤC

MỤC LỤC HÌNH ẢNH	5
CHƯƠNG 1: GIỚI THIỆU CHUNG VỀ GAME CARO	1
I. Sơ lược lịch sử về dòng game cờ:.....	1
II. Quá trình phát triển game Caro:.....	3
1. Lịch sử game Caro:	3
1.1 Khát quát:	3
1.2 Giới thiệu về game Caro:.....	5
1.3 Luật chơi:	7
CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ GAME.....	8
I. Yêu cầu phần mềm:	8
1. Yêu cầu chơi:	8
2. Giao diện hệ thống:	9
3. Tương Tác :.....	9
II. Phân tích và thiết kế.....	10
1. Xây dựng Module trò chơi:	10
1.1 Sơ đồ chức năng trò chơi:.....	10
1.2 Luồng xử lý của trò chơi:	11
2. Thiết Kế Hệ Thống.....	13
3. Thuật toán xây dựng game Caro:	29
3.1 Vét cạn:.....	29
3.2 Heuristics:	29
3.3 Minimax:	30
3.4 Thuật toán cắt tỉa Alpha-Beta:	33
Giới thiệu:	33
● Mục đích của thuật toán là làm giảm số nhánh trong cây tìm kiếm.	33

● Không làm ảnh hưởng đến sự đánh giá của đỉnh đang xét.	33
● Phương pháp cắt tỉa này cho phép cắt tỉa các nhánh không cần thiết cho sự đánh giá của đỉnh đang xét.....	33
3.5 So sánh Minimax và Alpha-Beta:	35
CHƯƠNG 3: THIẾT KẾ GIAO DIỆN	36
I. Giao diện chương trình:	36
CHƯƠNG 4: KẾT LUẬN	41
Ưu điểm:.....	41
Nhược điểm:.....	41
TÀI LIỆU THAM KHẢO	42
http://docshare02.docshare.tips/files/24124/241249261.pdf	42
https://vallicon.com/post/t%C3%ACm-ki%E1%BA%BFm-v%C3%A9tc%E1%BA%A1n-(complete-search)-DPZNpnDEwoq	42
https://www.howkteam.vn/d/chat-lan-su-dung-socket-trong-c-288	42
https://vi.wikipedia.org/wiki/C%E1%BB%9D_ca-r%C3%B4	42

MỤC LỤC HÌNH ẢNH

Hình 1 Màn hình game Caro online	5
Hình 2 Giao diện Game Caro trên điện thoại.....	6
Hình 3 Sơ đồ Use case Hệ thống	13
Hình 4 Sơ đồ lớp mức Phân tích	15
Hình 5 Sơ đồ lớp mức thiết kế.....	16
Hình 6 Sơ đồ hoạt động chơi game	17
Hình 7 Sơ đồ hoạt động chọn chế độ chơi.....	18
Hình 8 Sơ đồ hoạt động chức năng hỗ trợ	19
Hình 9 Sơ đồ hoạt động thoát game	20
Hình 10 Sơ đồ tuần tự chơi game.....	21
Hình 11 Sơ đồ tuần tự chọn chế độ chơi	22
Hình 12 Sơ đồ tuần tự chức năng hỗ trợ.....	23
Hình 13 Sơ đồ tuần tự chức năng thoát game.	24
Hình 14 Sơ đồ cộng tác chơi game	25
Hình 15 Sơ đồ cộng tác chọn chế độ chơi.....	25
Hình 16 Sơ đồ cộng tác Chọn chức năng hỗ trợ.....	26
Hình 17 Sơ đồ cộng tác thoát Game	27
Hình 18 Sơ đồ trạng thái chơi game	28
Hình 19 Ví dụ 1 Minimax	32
Hình 20 Ví dụ 2 MiniMax.....	32
Hình 21. Ví dụ thuật toán cắt tỉa Alpha-Beta.....	34
Hình 22 Ví dụ cắt tỉa Alpha.....	34
Hình 23 Ví dụ cắt tỉa Beta	35
Hình 24 Menu Game.....	36
Hình 25 Nhập tên khi người chơi với người	37
Hình 26 Nhập tên khi chơi với máy.....	37
Hình 27 Setting	38
Hình 28 Màn hình chính.....	39

Hình 29 Màn hình khi thắng/thua	40
--	----

CHƯƠNG 1: GIỚI THIỆU CHUNG VỀ GAME CARO

I. Sơ lược lịch sử về dòng game cờ:

Vào năm 1950, Alan Turing - một nhà nghiên cứu người Anh đi tiên phong trong lĩnh vực máy tính số, đã viết chương trình chơi cờ đầu tiên. Vào lúc đó, Turing phải viết và chạy chương trình của ông bằng... bút chì và giấy. Chương trình đó, cũng như chủ nhân của nó, chơi cờ rất tồi, nhưng đạt được mục đích: cho thấy máy tính có thể chơi được cờ. Cũng vào năm đó, Claude Shannon đã vạch ra một chiến lược cho máy tính chơi cờ tốt. Nhưng vào những năm 1950 tốc độ máy tính rất chậm nên không ai dám tiên đoán liệu máy tính có thể thắng con người được không, dù trong các trò chơi đơn giản như trò Checker.

Năm 1958, một chương trình chơi cờ đã lần đầu tiên hạ được đối phương là con người. Người thua là một cô thư kí của chính đội lập trình ra nó, cô chưa bao giờ chơi cờ trước đó và được dạy chơi cờ chỉ một giờ trước cuộc đấu. Đối với ngày nay chiến công này thật nhỏ nhoi, nhưng nó cho thấy tri thức có thể được đưa vào trong một chương trình chơi cờ. Lượng tri thức này được đo chính xác bằng một giờ học chơi.

Sau chiến thắng đó, một số người trong nhóm lập trình cờ đầu tiên đã tiên đoán rằng vào những năm 60 sẽ có chương trình chơi cờ được liệt vào hàng ngũ kiện tướng thế giới. Vào những năm cuối của thập kỷ 60, Spassky đã trở thành kiện tướng cờ thế giới và các chương trình chơi cờ đã chiếm được những thứ hạng cao trong hàng ngũ những người chơi cao cấp. Nhưng nhiều người cho rằng máy tính sẽ không bao giờ có thể giải quyết được những nhiệm vụ thông minh, không thể đạt được chức Vô địch cờ thế giới.

Lời tiên đoán này được nhắc lại một lần nữa vào những năm 70, liên quan đến một cuộc đánh cược giữa David Levy, một kiện tướng quốc tế người Anh (theo phân loại của Liên đoàn cờ quốc tế các đẳng cấp cao bao gồm: Kiện tướng quốc tế, Đại kiện tướng và Vô địch thế giới) và John McCarthy, một nhà nghiên cứu trong lĩnh vực trí tuệ nhân tạo. Lời thách đấu được đưa ra vào năm 1978. Trận đấu đã được diễn ra và chương trình cờ tốt nhất thời đó, CHESS 4.7 đã bị Levy hạ trong trận đấu có năm ván tại Toronto với thành tích ba ván người thắng, một hoà và một máy thắng. Levy không chỉ chiến thắng mà còn rút túi số tiền đánh cược 1000 bảng.

Nếu như mục đích của cuộc đánh cược là làm cho những nhà nghiên cứu phải nghĩ kỹ trước khi tiên đoán đến ngày thắng lợi, thì lần đánh cược này cho thấy: mặc dù tiên đoán sai trong những năm 1958-1968 và 1968-1978, các chuyên gia chương trình cờ lại tiếp tục tiên đoán tiếp rằng máy tính sẽ đạt đến vô địch cờ thế giới trong thập kỉ tiếp theo.

Nhưng một lần nữa, vào năm 1988, Vô địch cờ thế giới vẫn là con người. Trong năm tiếp theo, Deep Thought, một chương trình cờ mạnh nhất từ xưa đến nay đã chiến thắng một cách dễ dàng Kiện tướng Quốc tế Levy. Bộ não của Deep Thought có 250 chip và hai bộ xử lí trong một bảng mạch đơn, nó có khả năng xét 750.000 thế cờ trong một giây và tìm trước được đến 10 nước. Cũng trong năm đó, nó là máy tính đầu tiên hạ được một Đại kiện tướng (Bent Larsen). Deep Thought đã trở thành một trong một trăm người chơi cờ mạnh nhất thế giới. Nhưng trong trận đấu diễn ra vào năm 1989 giữa nhà Vô địch thế giới Garry Kasparov và Deep Thought thì nó đã bị nhà vô địch đè bẹp.

Các lời tiên đoán lại đến như các lần trước. Đã ba lần các nhà nghiên cứu tiên đoán: 'trong thập kỉ tới'. Nhưng lần này họ lại sửa lại là: 'trong 3 năm tới'... Trong năm 1993, Deep Thought đã hạ Judit Polgar - lúc đó là Đại kiện tướng trẻ nhất trong lịch sử và là người phụ nữ chơi hay nhất thế giới, trong trận đấu 2 ván.

Trong năm 1996, Deep Blue (tên mới của Deep Thought và lúc này nó thuộc hãng IBM) là một máy tính song song có 32 bộ xử lí với 256 mạch tích hợp cỡ lớn, khả năng xét từ 2 đến 400 triệu nước đi mỗi giây) đã thắng Gary Kasparov trong ván đầu tiên của trận đấu 6 ván, nhưng lại thua trong toàn trận (với tỉ số máy thắng 1, hoà 2 và thua 3).

Cuối cùng đích mà mọi người chờ đợi đã tới, nhưng sau 9 năm từ lời tiên đoán cuối và 39 năm từ lúc có chương trình chơi cờ đầu tiên, Deep Blue đã chiến thắng nhà đương kim Vô địch thế giới Garry Kasparov vào tháng 5/1997 trong một cuộc chiến dài đầy khó khăn, với tỷ số sát nút 2 thắng, 1 thua và 3 hoà.

II. Quá trình phát triển game Caro:

1. Lịch sử game Caro:

1. 1 Khát quát:

Cờ caro chính là môn cờ logic lâu đời và cổ xưa nhất trên Trái Đất. Cờ caro đã được sáng tạo từ nhiều nền văn minh khác nhau một cách độc lập. Nó bắt đầu xuất hiện từ năm 2000 trước CN ở sông Hoàng Hà, Trung Quốc. Một số nhà khoa học đã tìm thấy bằng chứng chứng minh Caro đã được phát minh ở Hy Lạp cổ đại và ở Châu Mỹ trước thời Colombo.

Môn cờ cổ của Trung Quốc là Wutzu. Cờ Caro du nhập từ Trung Quốc vào Nhật Bản từ khoảng năm 270 trước CN. Nó thường được gọi là Gomoku nhưng cũng có các tên gọi khác tùy theo thời gian và địa phương như Kakugo, gomoku-narabe, Itsutsu-ishi... Người ta đã tìm thấy một trò chơi cổ từ một di tích ở Nhật năm 100 sau CN và thấy nó là một biến thể của Caro. Nó đã lan truyền nhanh chóng với cái tên Kakugo (trò 5 quân). Các nhà sử học nói rằng vào các thế kỷ 17 và 18, mọi người đều chơi trò này- người già cũng như người trẻ. Năm 1858, khi quyển sách đầu tiên về trò chơi này được xuất bản, nó được gọi là Kakugo. Nó tiếp tục được chơi, được gọi với nhiều tên khác nhau như Goren, Goseki, rồi Gomokunarabe, Gomoku và phát triển cho đến ngày nay thành thể loại phức tạp nhất trong họ hàng đông đúc của nó, là Renju (chuối ngọc trai).

Khi trình độ các kỳ thủ Gomoku được nâng cao, họ nhận ra rằng nếu chỉ chơi đơn giản như trong Gomoku thì đó sẽ là một lợi thế quá lớn cho bên tiên tức bên Đen (thực tế chính là ưu thế trắng). Sau đó một số nhà toán học đã chứng minh được rằng nếu chơi với luật Gomoku trên bàn cờ bằng hoặc rộng hơn 15x15 thì Đen chắc chắn thắng (sure win), và sau đó cách đi cụ thể cũng đã được tìm ra, hệ thống và phân loại (theo tôi biết thì cách thắng hoàn toàn không duy nhất như nhiều người chờ đợi, mà thực tế có nhiều nhánh thắng (win branch) cho Đen).

Để các bạn hình dung rõ hơn tôi xin được nêu lại luật của Gomoku cổ:

Bàn cờ: 15x15

- Ai tạo được nước năm (1 hàng 5 quân liền nhau) thì thắng.
- Các nước Overline (≥ 6 quân) không có giá trị với cả hai bên và không bị coi là lỗi (cấm).

Từ đó, Gomoku lâm vào một giai đoạn khủng hoảng. Khả năng đánh thắng 100 phần trăm của Đen đã làm trò chơi này mất đi ý nghĩa của nó. Có

nhiều cải tiến được đề xuất, một số đã bị bỏ qua nhanh chóng, số khác làm xuất hiện các biến thể mới của Gomoku. Ý tưởng chung của các cải tiến là đề ra một số hạn chế cho Đen, nhằm cân bằng ưu thế đi tiên. Dưới đây là một số biến thể phổ biến.

Gomoku. Hiện nay được chơi chính thức với bàn 13x13. Không có hoà. Nếu hết đất thì Trắng thắng. Chưa tìm được chứng minh nào cho thấy Đen chắc chắn thắng. Tuy nhiên Đen vẫn có ưu thế rất lớn.

ProGomoku. Chơi trên bàn 15x15. Nước đầu của Đen đặt sẵn ở trung tâm. Nước thứ ba (nước thứ hai của Đen) phải đặt ngoài hình vuông cấm. Hình vuông cấm là hình vuông trung tâm kích thước 5x5. Không có hạn chế cho Trắng. Đã có chứng minh Đen chắc chắn thắng trong biến thể này.

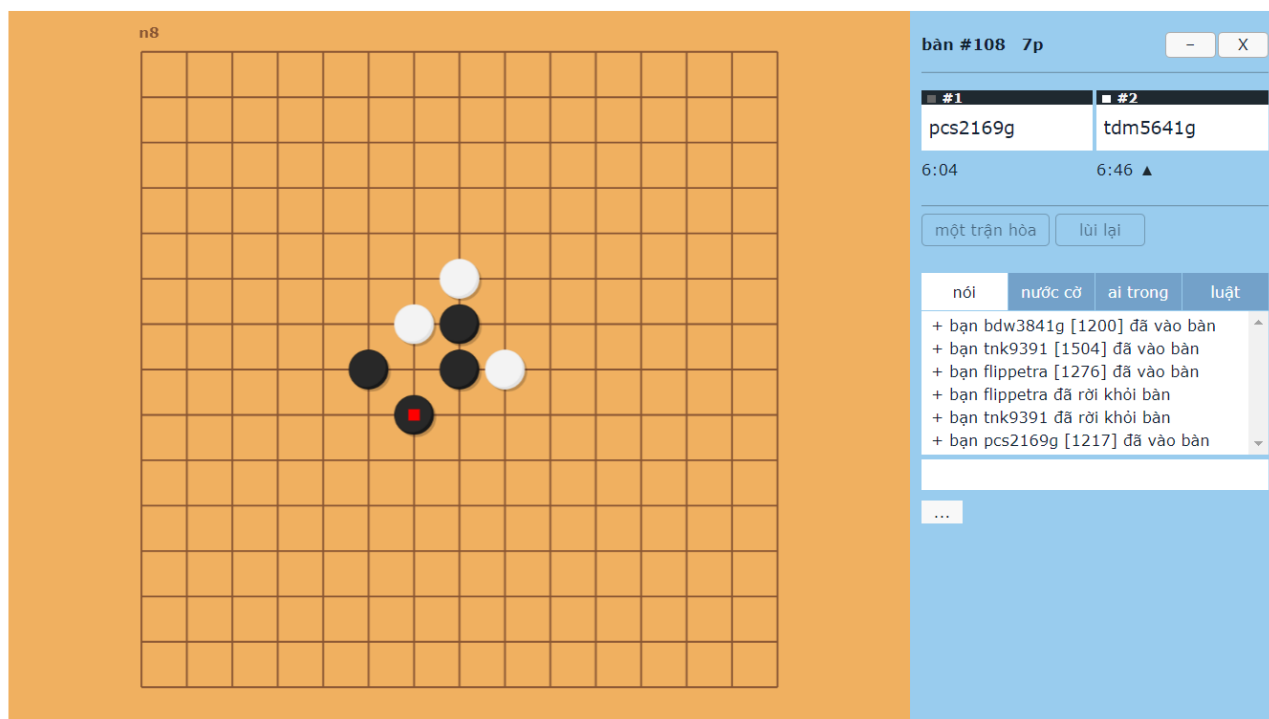
Pente. Biến thể này không còn giống Gomoku. Luật bổ sung là có thể ăn quân đối phương. Nước ăn quân được thực hiện bằng cách chặn hai đầu một nước hai quân đối phương và ăn hai quân đó. Ai tạo được nước năm hoặc ăn được 5 cặp quân trước thì thắng. Rất phổ biến ở Mỹ. Chơi trên bàn 19x19.

Tuy nhiên trò chơi này chỉ thực sự lấy lại được sự hấp dẫn của nó khi phát triển thành thể loại hoàn thiện nhất được chơi ngày nay - Renju. Renju cũng đã mất hàng chục năm để thử nghiệm và phát triển những luật mới. Cụ thể là vào năm 1899, cái tên Renju (chuối ngọc trai) ra đời và đến năm 1966 thì nó đã được hoàn thiện khi Liên đoàn Renju Nhật Bản - Nihon Renju Sha ra đời và công bố luật Renju chính thức!

1.2 Giới thiệu về game Caro:

Đã trải qua bao thập kỷ và cho đến nay Caro vẫn được ưa chuộng trên toàn thế giới, với cách chơi đơn giản, nhưng càng chơi thì sự biến hóa của các nước cờ là vô cùng kì diệu. Sự hấp dẫn game Caro đã lôi cuốn các bạn trẻ Việt Nam rất mạnh, nhất là đối với các học sinh, sinh viên. Sau giờ nghỉ giải lao đều có thể chơi với 1 mảnh giấy nhỏ kẻ ô vuông và 2 cây bút. Và đây chính là 1 game giải trí lành mạnh, phát triển trí tuệ cho các bạn.

Giờ đây với sự phát triển của công nghệ, Game Caro đã được đưa vào PC với rất nhiều các phiên bản cùng nhiều cấp độ cho người chơi với máy, hay hai người chơi với nhau. Cùng với đó Caro cũng chiếm 1 phần trong làng game giải trí online, chủ yếu để 2 người chơi thi đấu và thư giãn khi đánh với máy. Thật tiện ích khi game caro được đưa vào các điện thoại di động có thể chơi được mọi lúc.



Hình 1 Màn hình game Caro online



Hình 2 Giao diện Game Caro trên điện thoại

1. 3 Luật chơi:

a) Tic tac toe:

- Bàn cờ trước là 3x3(3 hàng , 3 cột), có 9 ô vuông ,có 2 bên quân trắng và đen.
- Quân đen đi trước, quân trắng đi sau, khi nào 1 trong 2 quân có 3 quân cùng màu liên nhau thì quân đó thắng.

b) Caro

- Bàn cờ 15x15 (15 hàng, 15 cột), 225 ô vuông, cũng có bên quân trắng, quân đen. Quân đen đi trước, trắng đi sau liên tiếp. Khi nào 1 trong 2 quân có 5 quân cùng màu liên tiếp cạnh nhau (xét theo cả 2 đường chéo, đường dọc và đường ngang) thì quân đó chiến thắng. Ván cờ hòa khi hết ô vuông trống mà chưa phân thắng bại
- Bàn cờ 25 x 25 thì luật chơi tương tự trên.

CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ GAME

I. Yêu cầu phần mềm:

1. Yêu cầu chơi:

Game Caro đã phổ biến với rất nhiều người, với nhiều phiên bản game được đưa ra, ở đây chúng em xây dựng game Caro này phù hợp với nhiều đối tượng Người chơi.

Người chơi có thể chọn 1 trong 2 chế độ chơi:

- Hai Người chơi.
- Chơi với máy.
- Chơi online qua mạng Lan

Ở chế độ “Hai Người chơi”, hai bên lần lượt đi từng nước cờ, chọn loại quân cờ phù hợp, chọn có hoặc không có âm thanh khi chơi.

Ở chế độ “Chơi với máy”, Người chơi và máy lần lượt đi từng nước cờ, NC chọn loại quân cờ phù hợp, có âm thanh khi chơi.

Ở chế độ thông qua mạng LAN thì tương tự chế độ “Hai Người Chơi”.

Ngoài ra còn có:

- Tạo ván chơi mới: trong lúc đang chơi mà không muốn tiếp tục ván chơi đó có thể tạo ván chơi mới khởi động lại từ đầu.
- Nước đi: nhanh, có âm thanh đi kèm (khi Người chơi chọn âm thanh). Sau mỗi nước đi sẽ hiện ra vị trí nước đi đó cho từng bên.
- Thời gian: hiển thị thời gian chơi được tính bắt đầu khi Người chơi đi một quân cờ mới.

2. Giao diện hệ thống:

- Thuận tiện: Các nút chức năng phải dễ nhìn, dễ sử dụng.
- Hình ảnh: Ảnh nền, ảnh bàn cờ, ảnh quân cờ, ảnh nút,...đơn giản, dễ nhìn.
- Âm thanh: Có âm thanh nước đi trong khi chơi.
- Giao diện:

➤ Giao diện chính: Bao gồm bàn cờ 20x20 tổng 400 ô với mỗi hàng dọc và hàng ngang là 20 ô liên tiếp nhau. Bên trái bàn cờ có 1 số Button để giúp người chơi lựa chọn chế độ chơi mà mình mong muốn cũng như tạo ván chơi mới hay kết thúc và thoát khỏi game. Còn có thanh thể hiện thời gian của người chơi khi đi cờ.

3. Tương Tác :

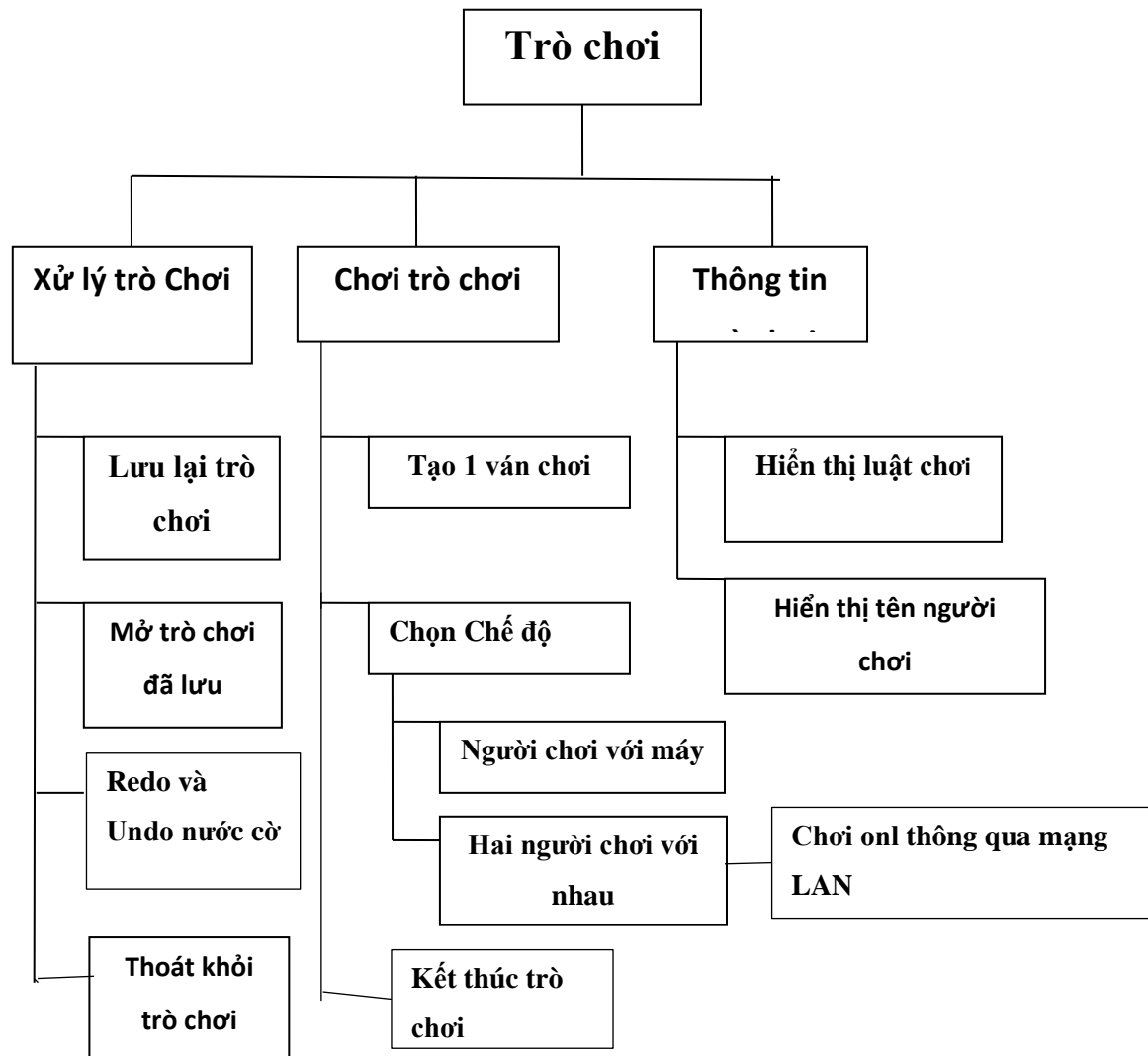
Người chơi có thể tương tác nhanh với chương trình bằng phím hoặc chuột:

- Phím hoặc chuột dùng để điều khiển chương trình.
- Chuột dùng để đi nước cờ.

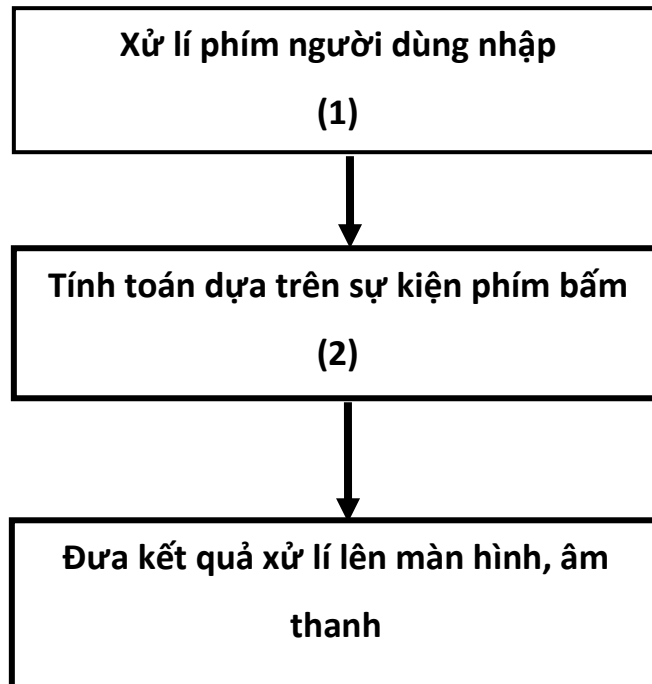
II. Phân tích và thiết kế

1. Xây dựng Module trò chơi:

1.1 Sơ đồ chức năng trò chơi:



1. 2 Luồng xử lí của trò chơi:



(1): nhận phím bấm, xử lí sự kiện bấm phím đó

(kích chuột để đi quân, và lựa chọn một chức năng trong menu...)

(2): với mỗi đầu vào **(1)**, game sẽ thực hiện một chức năng tương ứng(tìm nước đi quân hợp lệ, kiểm tra ván cờ kết thúc ...)

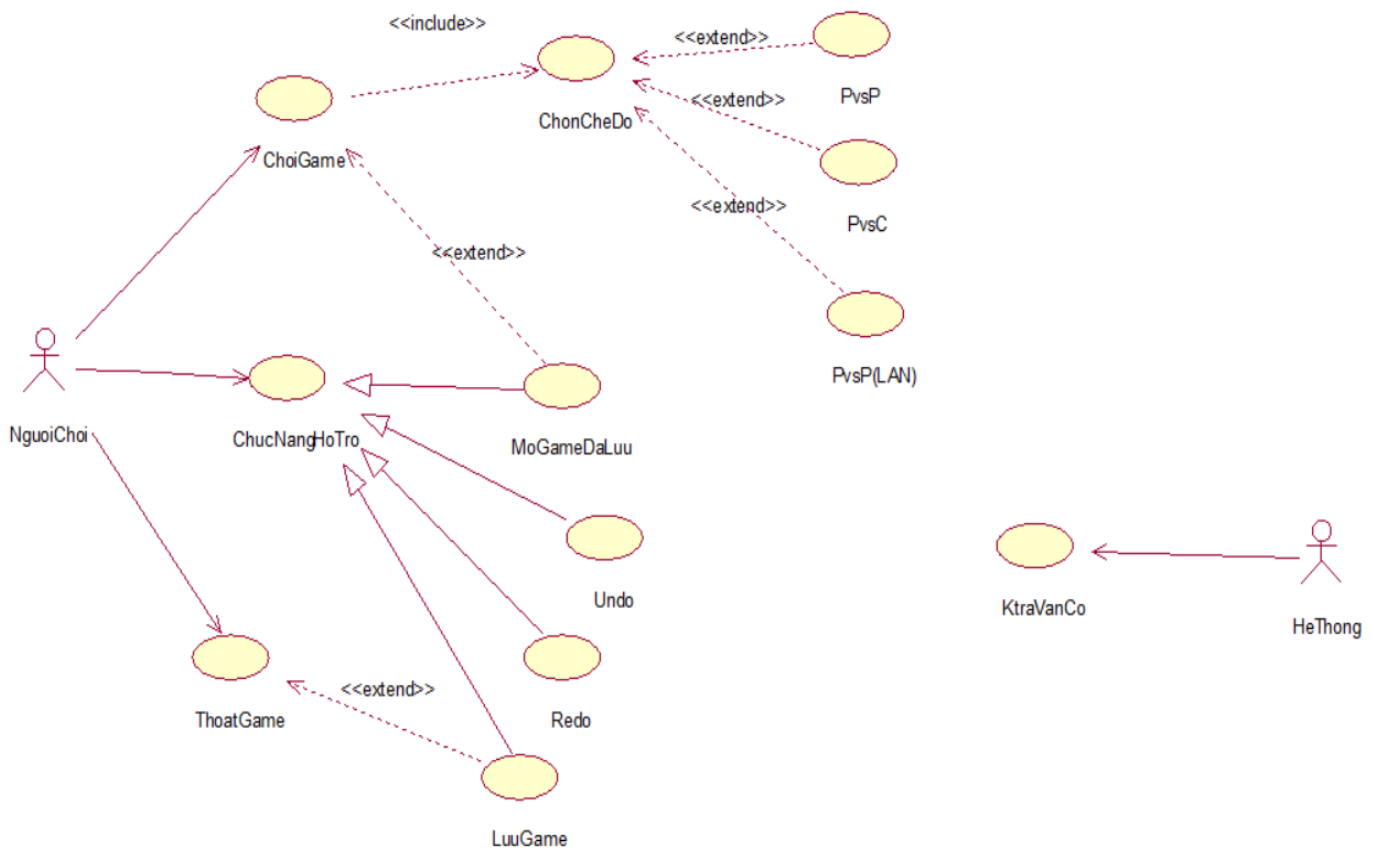
(3): đảm nhiệm chức năng vẽ hình và hiển thị :vẽ quân cờ, hiển thị thông báo,...

Các sự kiện xử lý chức năng trong chương trình:

- Tạo ván chơi mới:
 - ✓ Khởi tạo bàn cờ.
 - ✓ Đưa bàn cờ về trạng thái chưa kết thúc.(trạng thái kết thúc là khi ván chơi có 1 trong 2 bên đối thủ giành chiến thắng hoặc không còn ô để đi)
 - ✓ Thông báo về chiến thắng là rỗng.
 - ✓ Đếm nước đi khởi tạo giá trị 0.
 - ✓ Thời gian chơi khởi tạo về giá trị 0.
 - ✓ Có phát ra âm thanh khi chơi(mặc định là có âm thanh).
 - ✓ Hiện thị ảnh quân cờ).
- Chơi với máy:
 - ✓ Đưa ra máy tính
 - ✓ Đưa bàn cờ về trạng thái chưa kết thúc
 - ✓ Khởi tạo giá trị cho biến về số ván thắng của người chơi và máy=0.
 - ✓ Thông báo chiến thắng khởi tạo về rỗng.
 - ✓ Hiện thị thông báo người thắng
 - ✓ Gọi lại chức năng “tạo ván chơi mới”.
- Hai người chơi:
 - ✓ Đưa ra máy tính
 - ✓ Đưa bàn cờ về trạng thái chưa kết thúc.
 - ✓ Thông báo chiến thắng khởi tạo về rỗng.
 - ✓ Gọi lại chức năng “tạo ván chơi mới”.
- Lưu ván chơi:

2. Thiết Kế Hệ Thống

2.1. Use Case Hệ Thống.



Hình 3 Sơ đồ Use case Hệ thống

2.2. Đặc tả hệ thống.

- **Chơi Game** : Khi người chơi mở game và muốn chơi game, người chơi phải chọn chế độ chơi của game như : PvsP, PvsC và PvsP(LAN). Sau khi chọn chế độ người chơi sẽ chơi game bằng cách click chuột vào bàn cờ (panelBanCo) để đánh quân cờ ở ô cờ mà người chơi muốn.

- **Chức năng hỗ trợ**: Trong game người chơi có thể chọn thực hiện các chức năng như: Lưu game, mở game đã lưu, Undo và Redo.

+**Lưu game**: Trong quá trình chơi game nếu người chơi muốn lưu lại ván cờ của mình để chơi lại từ những nước cờ đã đi đó.

+**Mở game**: Chức năng này hỗ trợ mở lại những ván cờ mà người chơi đã lưu trước đó để tiếp tục chơi.

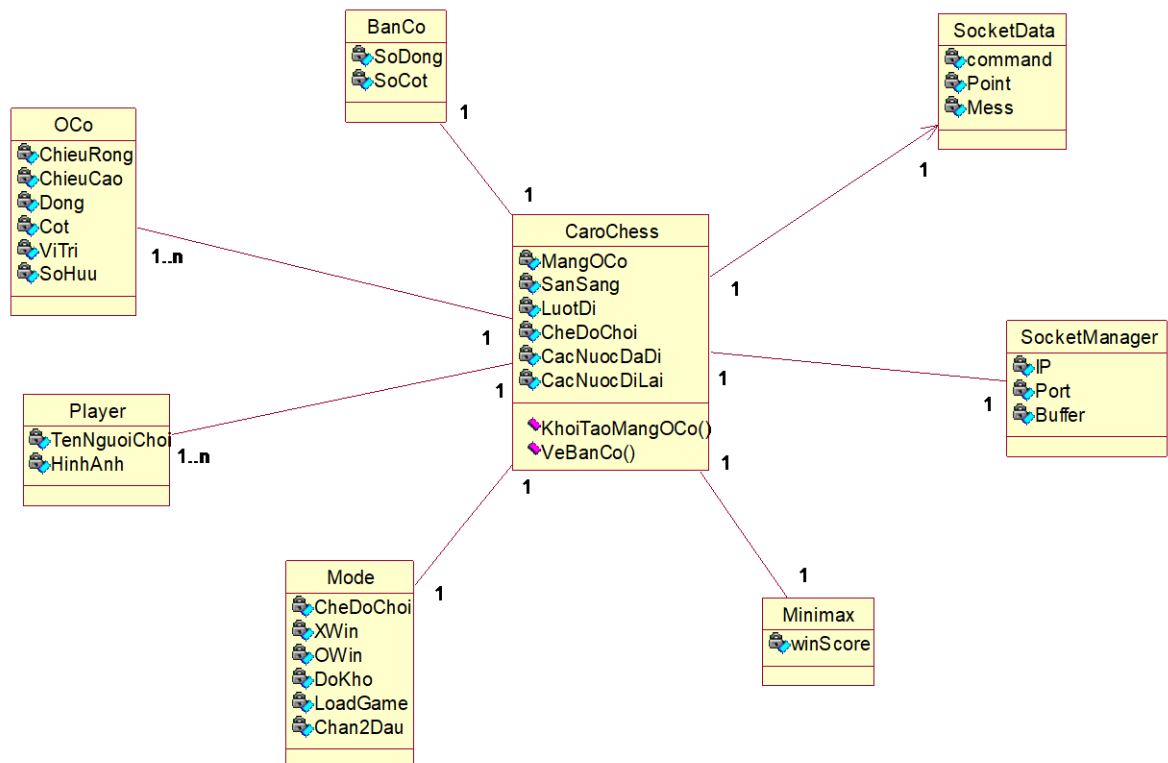
+ **Undo và Redo:** Trong quá trình đang chơi game , người chơi có thể thu hồi 1 nước đã đi nhằm để đi lại thì người chơi có thể sử dụng chức năng Undo của game, hoặc trong trường hợp người chơi lỡ Undo nhầm 1 nước cờ thì người chơi có thể sử dụng chức năng Redo để phục hồi nước vừa Undo nhầm. Lưu ý: Khi người chơi thực hiện Undo và đánh nước mới rồi thì những nước cờ Undo không được Redo lại.

-**Thoát game:** Khi chơi game xong hoặc người chơi không muốn chơi game nữa người chơi có thể thoát game. Khi thoát game Hệ thống sẽ kiểm tra nếu bàn cờ đã kết thúc thì cho game thoát còn nếu bàn cờ vẫn còn sẵn sàng thì tiếp tục kiểm tra Game đã được người chơi lưu lại hay chơi nếu đã lưu thì thoát game, ngược lại sẽ hiện thông báo cho người chơi muốn lưu game hay không? Nếu lưu thì thực hiện chức năng lưu rồi thoát game, còn người chơi không muốn lưu thì tiến hành thoát game.

-**Kiểm tra ván cờ :** Trong quá trình chơi game hệ thống luôn kiểm tra trạng thái của bàn cờ để xử lý thắng thua cho người chơi bằng các phương thức được cài đặt trong chương trình và thông báo kết quả cho người chơi.

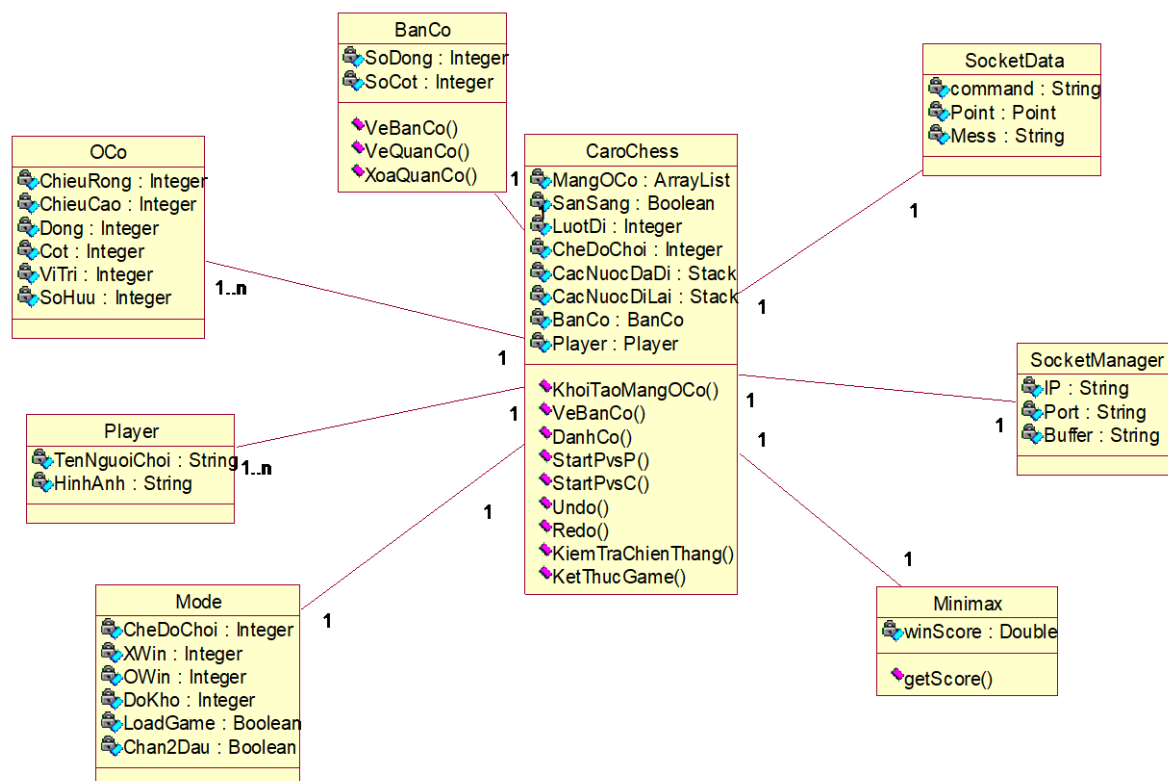
2.3. Xây dựng sơ đồ lớp phân tích và thiết kế.

2.3.1. Sơ đồ lớp mức phân tích.



Hình 4 Sơ đồ lớp mức Phân tích

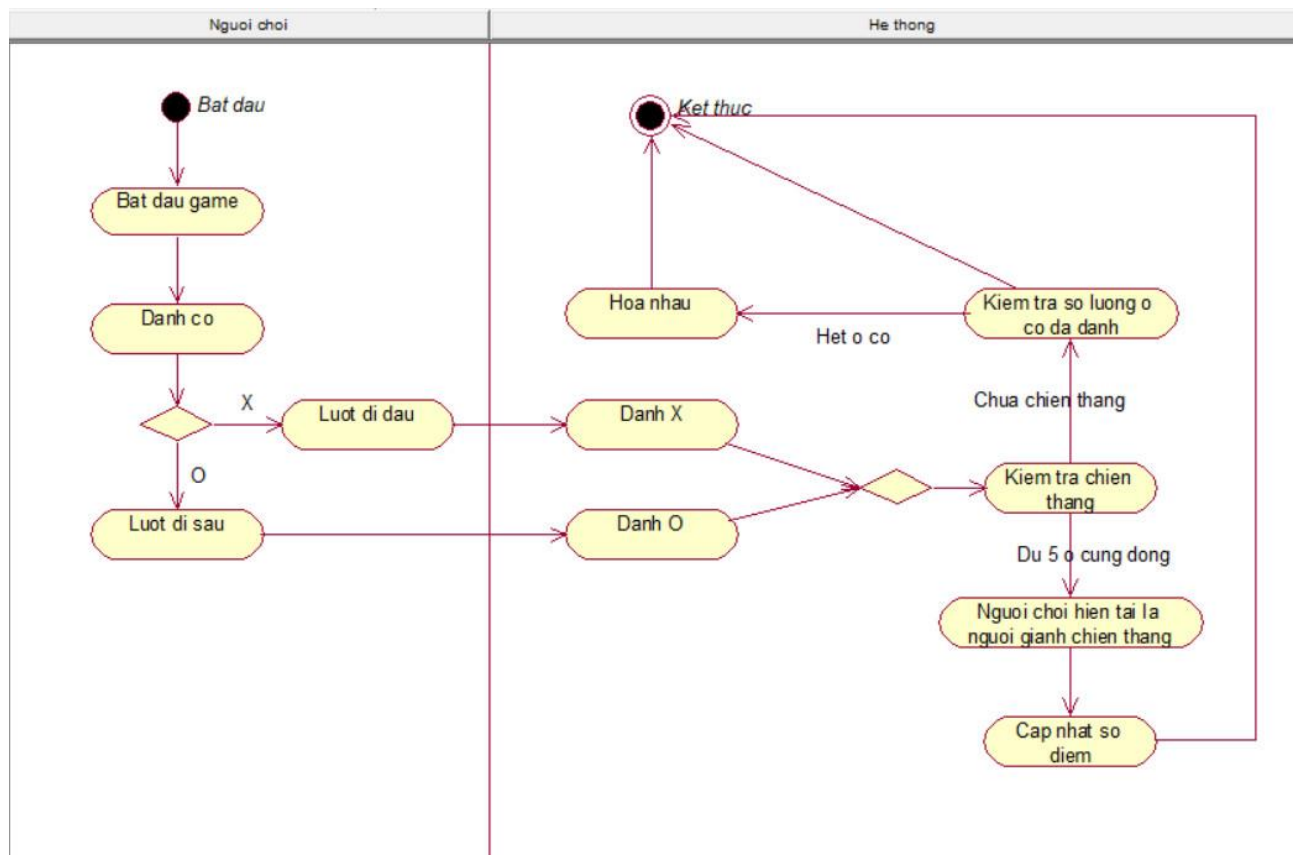
2.3.2. Sơ đồ lớp mức thiết kế.



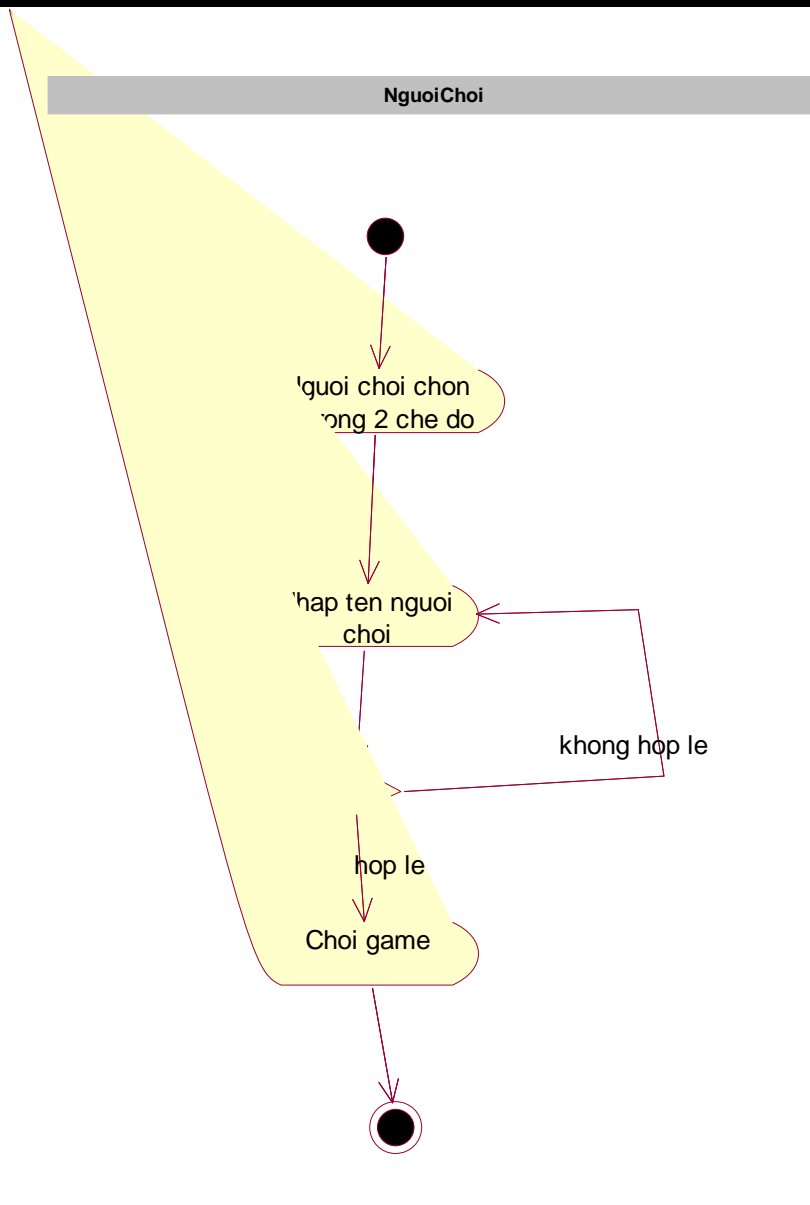
Hình 5 Sơ đồ lớp mức thiết kế

2.4. Xây dựng sơ đồ hóa chức năng.

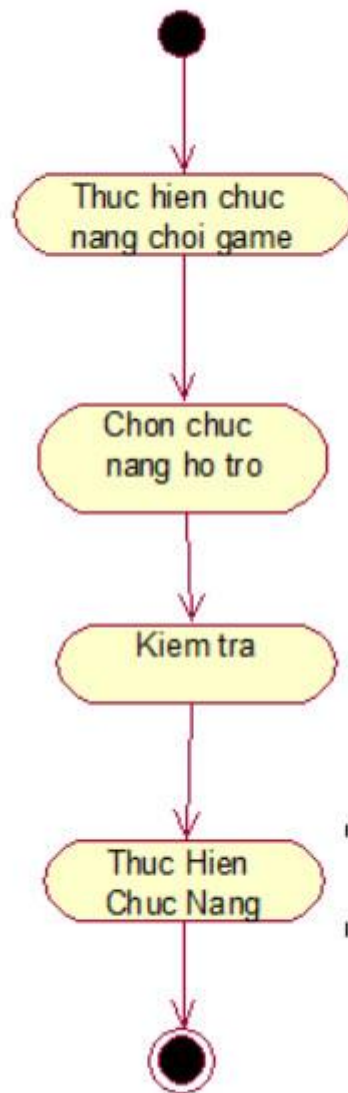
2.4.1. Sơ đồ hoạt động.



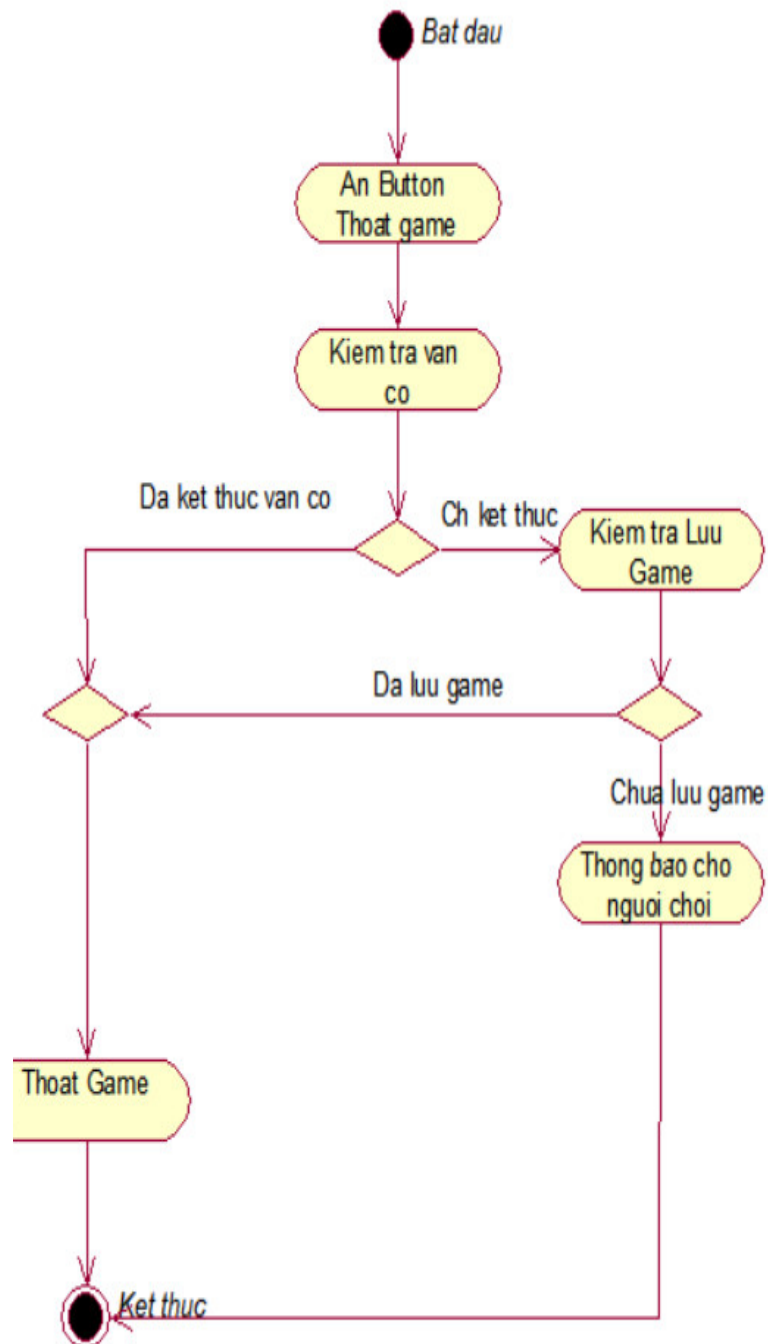
Hình 6 Sơ đồ hoạt động chơi game



Hình 7 Sơ đồ hoạt động chọn chế độ chơi

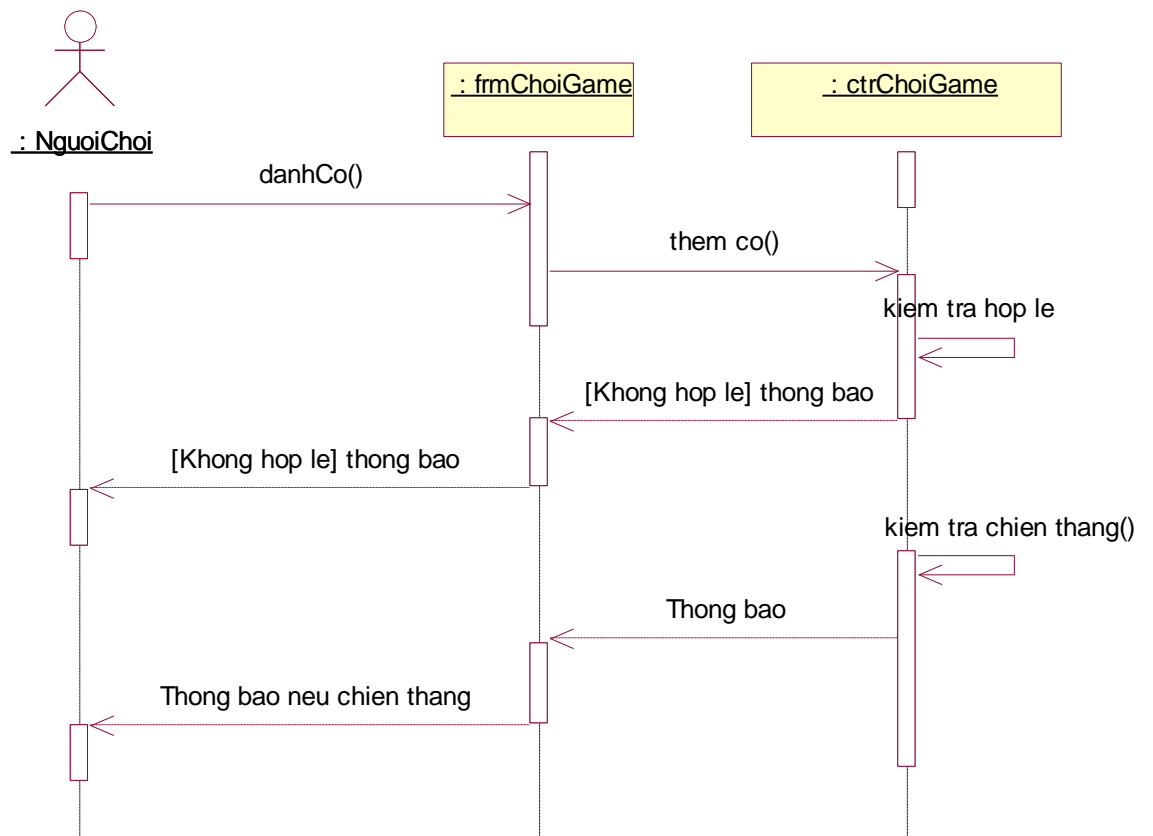


Hình 8 Sơ đồ hoạt động chức năng hỗ trợ

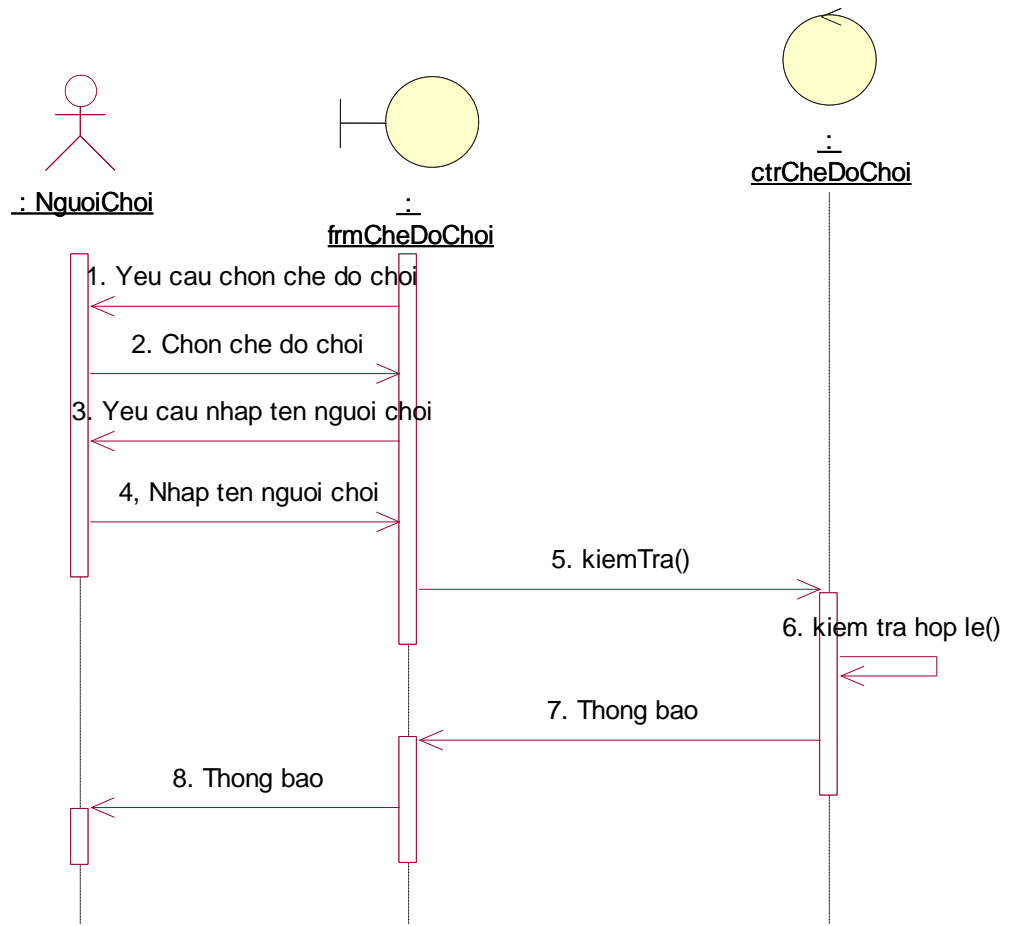


Hình 9 Sơ đồ hoạt động thoát game

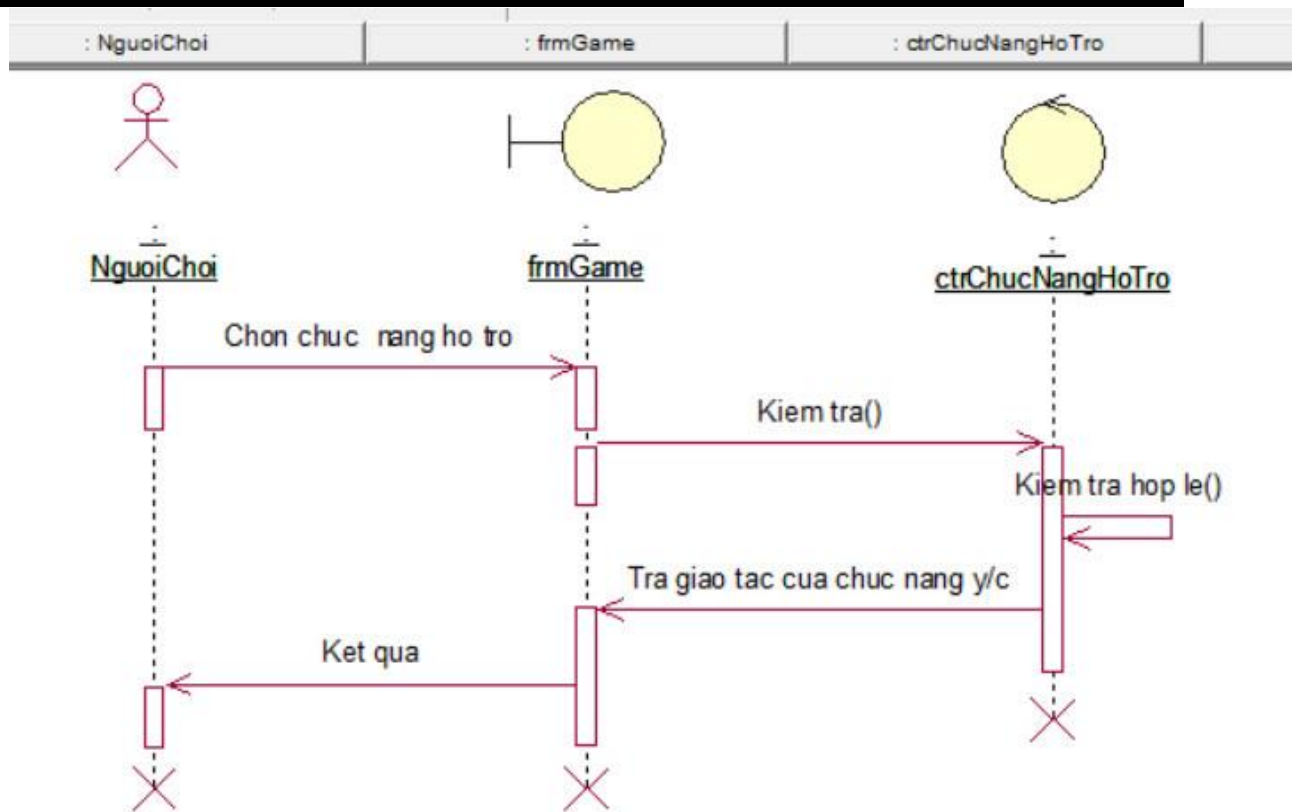
2.4.2.Sơ đồ tuần tự



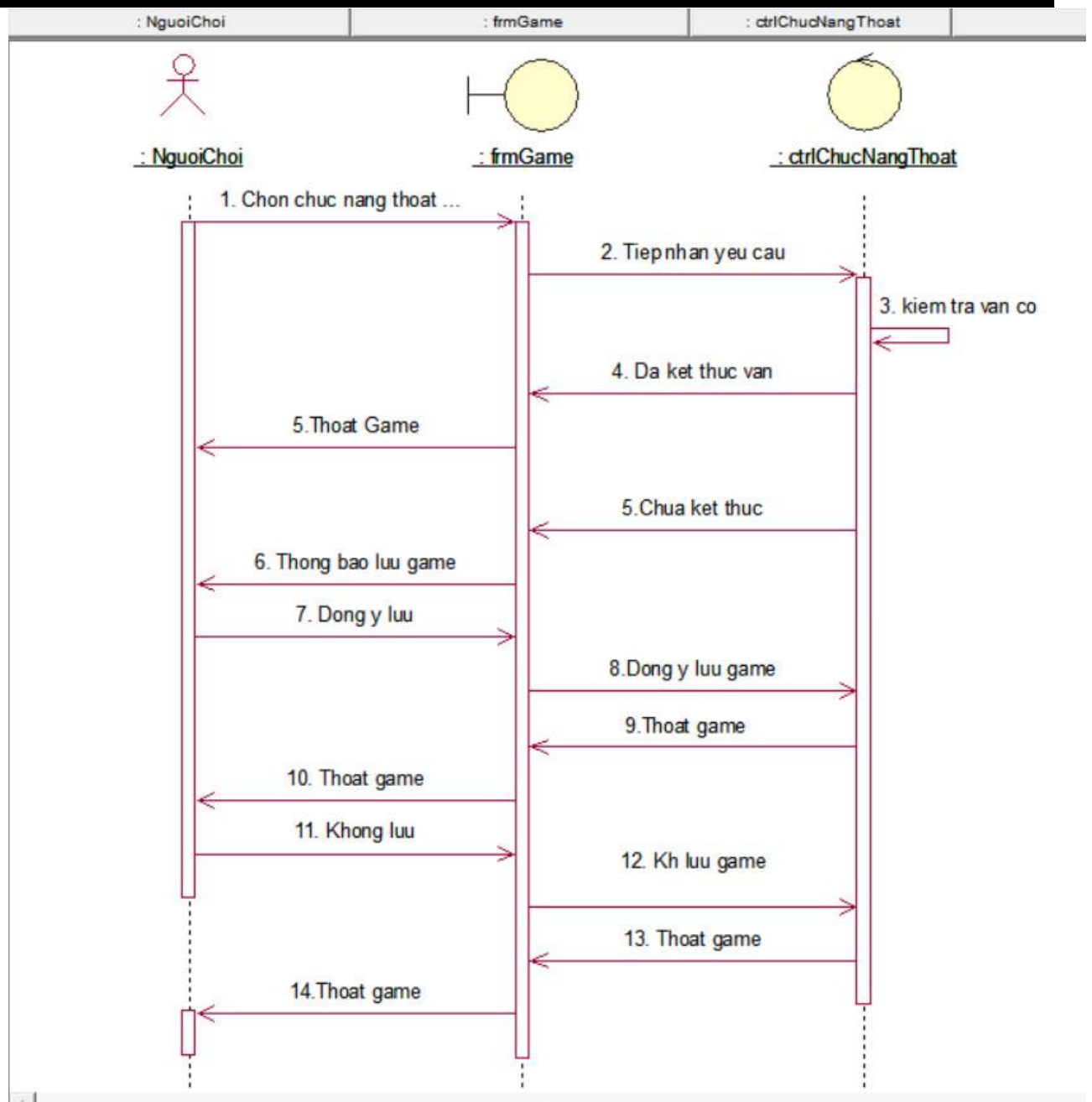
Hình 10 Sơ đồ tuần tự chơi game



Hình 11 Sơ đồ tuần tự chọn chế độ chơi

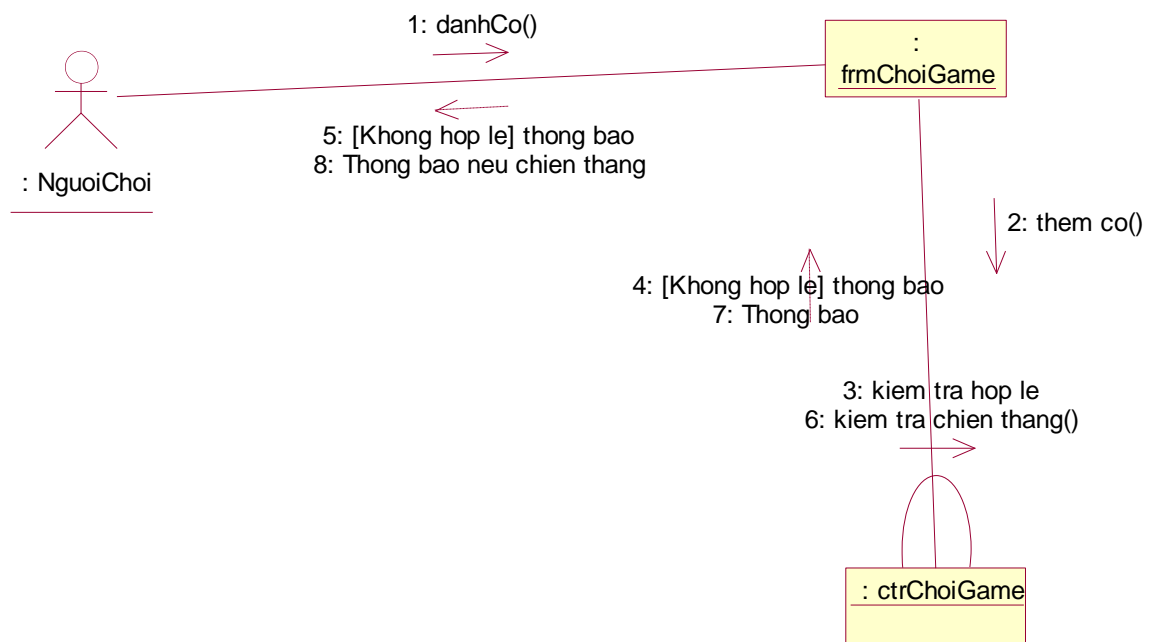


Hình 12 Sơ đồ tuần tự chức năng hỗ trợ

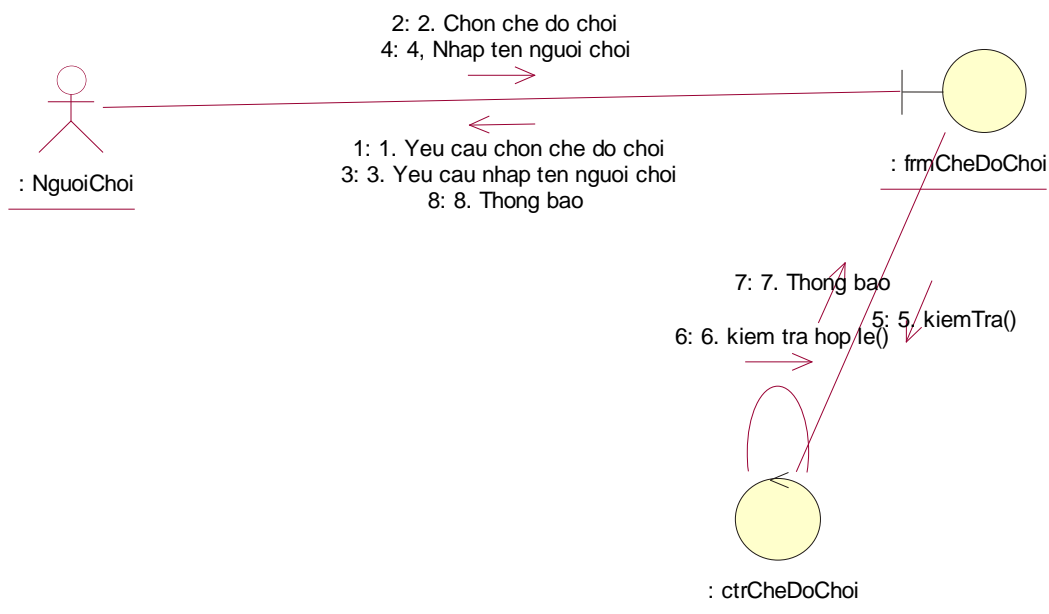


Hình 13 Sơ đồ tuần tự chức năng thoát game.

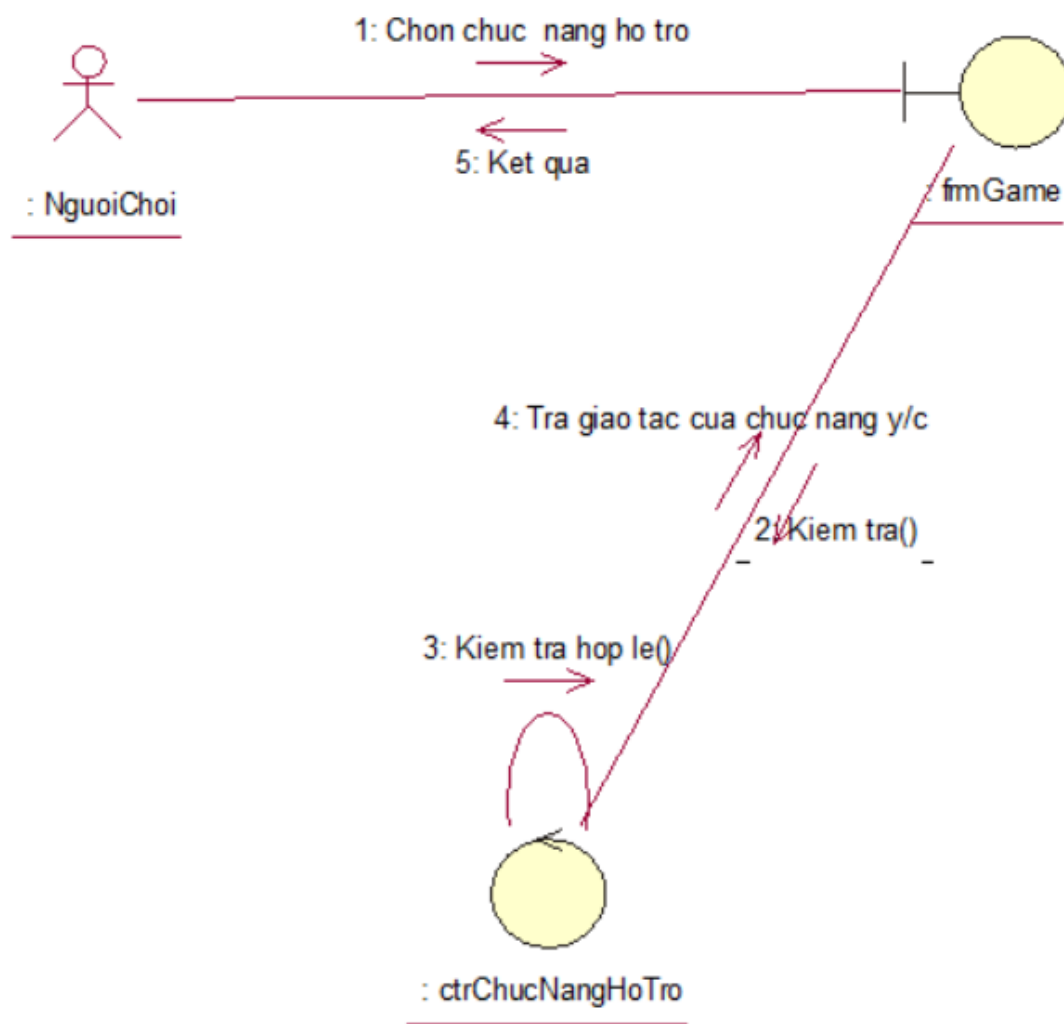
2.4.3. Sơ đồ cộng tác



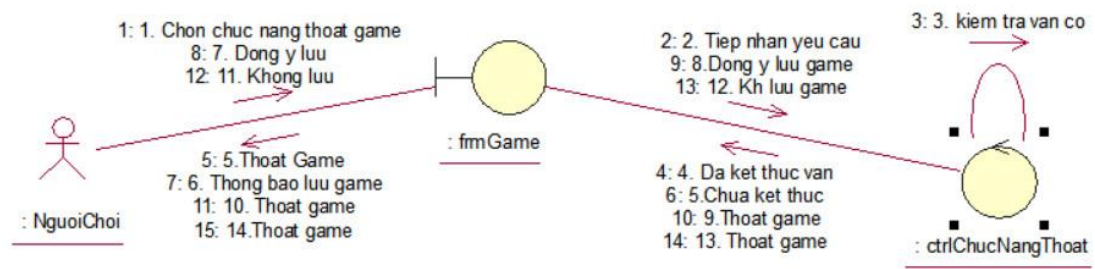
Hình 14 Sơ đồ cộng tác chơi game



Hình 15 Sơ đồ cộng tác chọn chế độ chơi

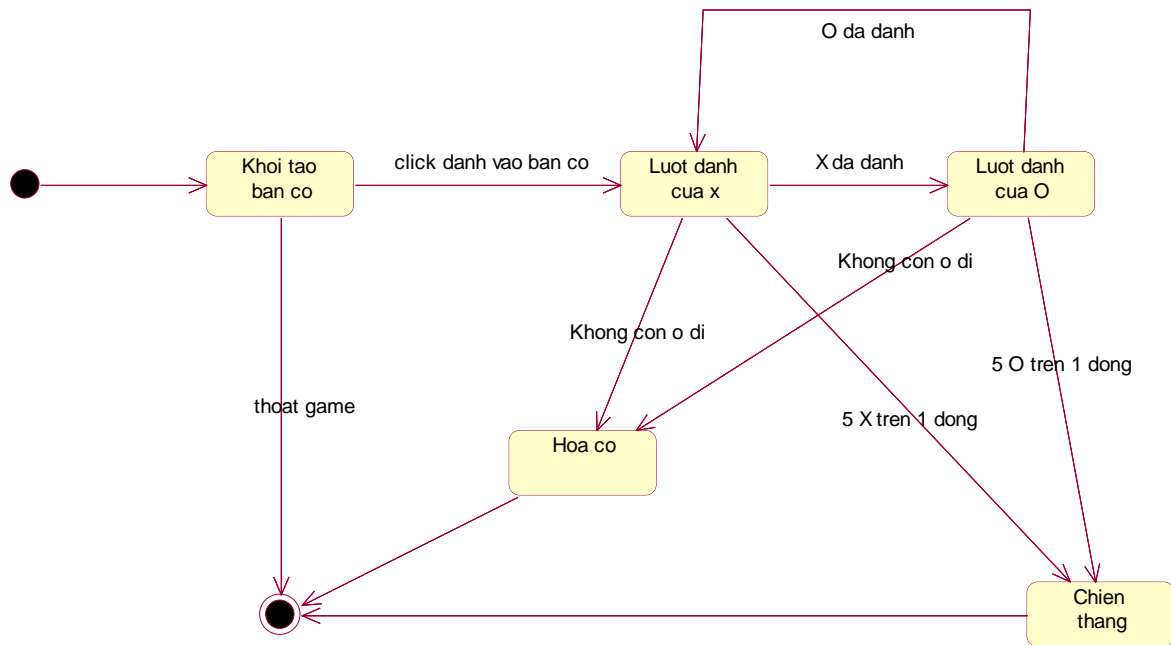


Hình 16 Sơ đồ cộng tác Chọn chức năng hỗ trợ



Hình 17 Sơ đồ công tác thoát Game

2.4.4. Sơ đồ trạng thái



Hình 18 Sơ đồ trạng thái chơi game

3. Thuật toán xây dựng game Caro:

3.1 Vét cạn:

Vét cạn là một phương pháp giải toán trong tin học: tìm nghiệm của một bài toán bằng cách xem xét tất cả các phương án có thể.

Đây là cách tiếp cận cơ bản nhất để giải các bài toán trong tin học, phương pháp này thường được nghĩ đến đầu tiên trong quá trình thiết kế thuật toán để giải một bài toán trong tin học.

Ưu điểm của phương pháp này là luôn đảm bảo tìm ra nghiệm đúng, chính xác. Tuy nhiên, hạn chế của phương pháp này là thời gian thực thi lâu, độ phức tạp lớn. Do đó vét cạn thường chỉ phù hợp với các bài toán có kích thước nhỏ.

Kết Luận: Tìm kiếm vét cạn không phải là một giải pháp tối ưu để giải quyết một bài toán, nhưng nó luôn là 1 cách tiếp cận dễ để cho ta 1 giải pháp ban đầu đúng đắn. Từ lời giải này ta có thể áp dụng các kỹ thuật như tìm kiếm nhánh cận hoặc chia để trị để tối ưu dần

3.2 Heuristics:

Là các kỹ thuật dựa trên kinh nghiệm để giải quyết vấn đề, học hỏi hay khám phá nhằm đưa ra một giải pháp mà không được đảm bảo là tối ưu. Với việc nghiên cứu khảo sát không có tính thực tế, các phương pháp heuristic được dùng nhằm tăng nhanh quá trình tìm kiếm với các giải pháp hợp lý thông qua các suy nghĩ rút gọn để giảm bớt việc nhận thức vấn đề khi đưa ra quyết định. Ví dụ của phương pháp này bao gồm sử dụng một luật ngón tay cái, giả thuyết, phán đoán trực giác, khuôn mẫu hay nhận thức thông thường.

Thuật giải Heuristic là một sự mở rộng khái niệm thuật toán. Nó thể hiện cách giải bài toán với các đặc tính sau :

- + Thường tìm được lời giải tốt (nhưng không chắc là lời giải tốt nhất)
- + Giải bài toán theo thuật giải Heuristic thường dễ dàng và nhanh chóng đưa ra kết quả hơn so với giải thuật tối ưu, vì vậy chi phí thấp hơn.
- + Thuật giải Heuristic thường thể hiện khá tự nhiên, gần gũi với cách suy nghĩ và hành động của con người.

Có nhiều phương pháp để xây dựng một thuật giải Heuristic, trong đó người ta thường dựa vào một số nguyên lý cơ sở như sau:

Nguyên lý vét cạn thông minh :

Trong một bài toán tìm kiếm nào đó, khi không gian tìm kiếm lớn, ta thường tìm cách giới hạn lại không gian tìm kiếm hoặc thực hiện một kiểu dò tìm đặc biệt dựa vào đặc thù của bài toán để nhanh chóng tìm ra mục tiêu.

Nguyên lý tham lam (Greedy):

Lấy tiêu chuẩn tối ưu (trên phạm vi toàn cục) của bài toán để làm tiêu chuẩn chọn lựa hành động cho phạm vi cục bộ của từng bước (hay từng giai đoạn) trong quá trình tìm kiếm lời giải.

Nguyên lý thứ tự :

Thực hiện hành động dựa trên một cấu trúc thứ tự hợp lý của không gian khảo sát nhằm nhanh chóng đạt được một lời giải tốt.

Hàm Heuristic:

Trong việc xây dựng các thuật giải Heuristic, người ta thường dùng các *hàm Heuristic*. Đó là các hàm đánh giá thô, giá trị của hàm phụ thuộc vào trạng thái hiện tại của bài toán tại mỗi bước giải. Nhờ giá trị này, ta có thể chọn được cách hành động tương đối hợp lý trong từng bước của thuật giải.

3.3 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 định giá trị cho các Node trên cây trò chơi sau đó tìm Node có giá trị phù hợp để đi bước tiếp theo.

Như các bạn đã biết thì có rất nhiều thuật toán tìm kiếm để làm AI trong game như A*, Heuristic... Mỗi thuật toán thì sẽ phù hợp với từng loại game cho nó. Những game đối kháng trong đó người chơi luân phiên đánh như cờ vua, cờ tướng, caro... Khi chơi bạn có thể khai triển hết không gian trạng thái nhưng khó khăn chủ yếu là bạn phải tính toán được phản ứng và nước đi của đối thủ mình như thế nào? Cách xử lý đơn giản là bạn giả sử đối thủ của bạn cũng sử dụng kiến thức về không gian trạng thái giống bạn. Giải thuật Minimax áp dụng giả thuyết này để tìm kiếm không gian trạng thái của trò chơi. Trường hợp này thuật toán minimax sẽ đáp ứng những gì mình cần.

Một số khái niệm:

- Cây trò chơi (Game tree) - Đại khái là một sơ đồ hình cây thể hiện từng trạng thái, từng trường hợp của trò chơi theo từng nước đi.
- Mỗi node biểu diễn 1 trạng thái của trò chơi hiện tại trên cây trò chơi.
- Node được gọi nút lá là tại đó trò chơi kết thúc (trạng thái trò chơi lúc đó có thể thắng, thua hoặc hòa).

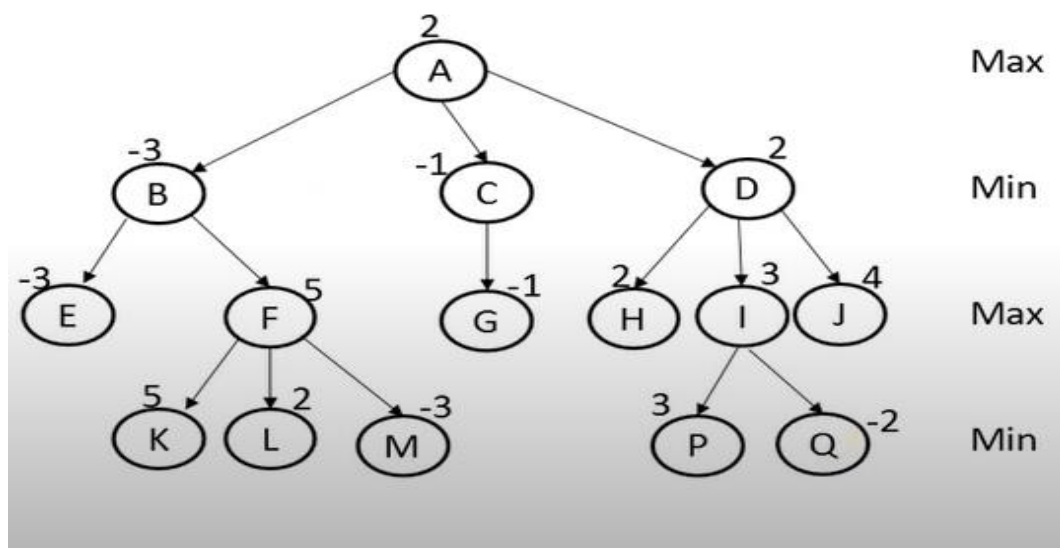
Giải thuật Minimax Hai người chơi trong game được đại diện là MAX và MIN. MAX đại diện cho người chơi luôn muốn chiến thắng và cố gắng tối ưu hóa ưu thế của mình còn MIN đại diện cho người chơi cố gắng cho người MAX giành số điểm càng thấp càng tốt. Giải thuật Minimax thể hiện bằng cách định trị các 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.

Nhận định:

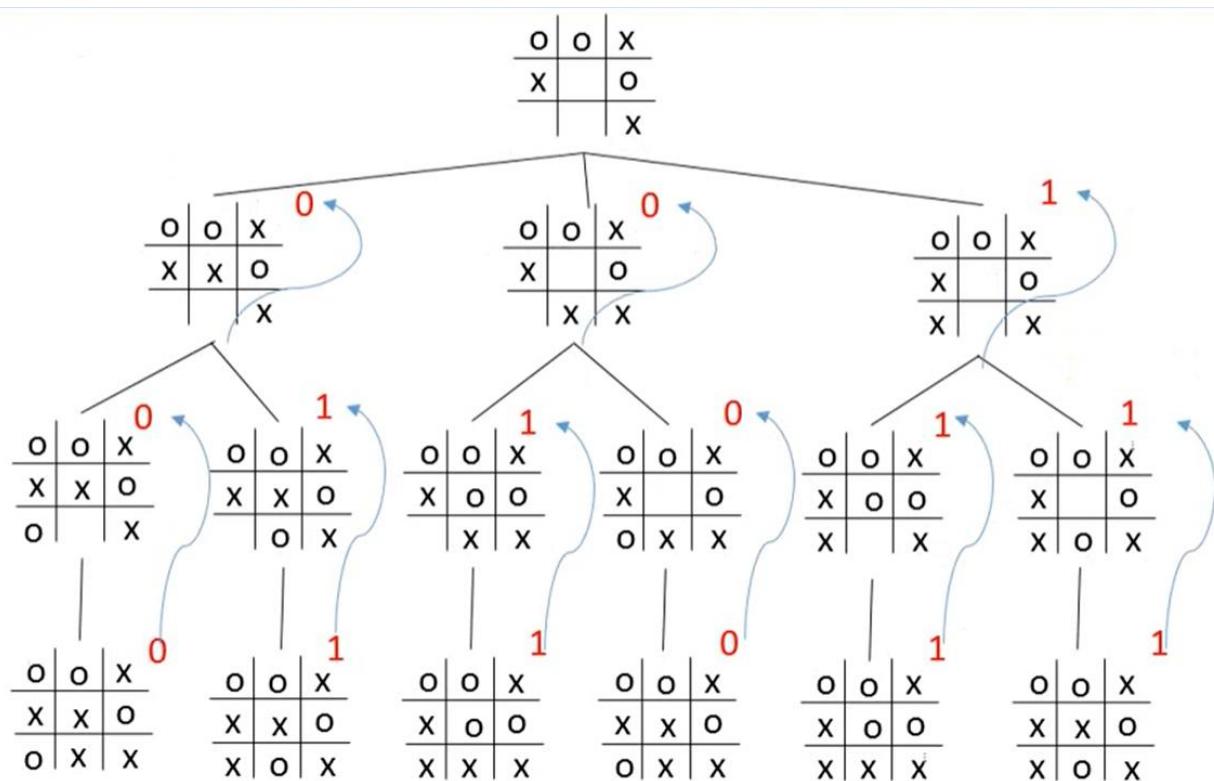
- Giả sử đến một thời điểm đường đi đã dẫn đến đỉnh u
- Nước đi tối ưu cho trạng thái u là nước đi dẫn tới đỉnh v(tốt nhất trong tất cả các đỉnh của u).
- Để chọn nước đi tốt nhất cho u, ta cần xác định giá trị các đỉnh của cây trò chơi có gốc u.

Các cách tính điểm cho các đỉnh trên trò chơi:

- Để xác định giá trị tại đỉnh có gốc là u, ta sẽ phải đi từ mức thấp nhất cho đến đỉnh u
- Cây trò chơi sẽ được phân thành các lớp Min- Max xen kẽ nhau:
 - ◆ Lớp Min: Lấy giá trị nhỏ nhất của các node con.
 - ◆ Lớp Max: Lấy giá trị lớn nhất của các node con.



Hình 19 Ví dụ 1 Minimax



Hình 20 Ví dụ 2 MiniMax

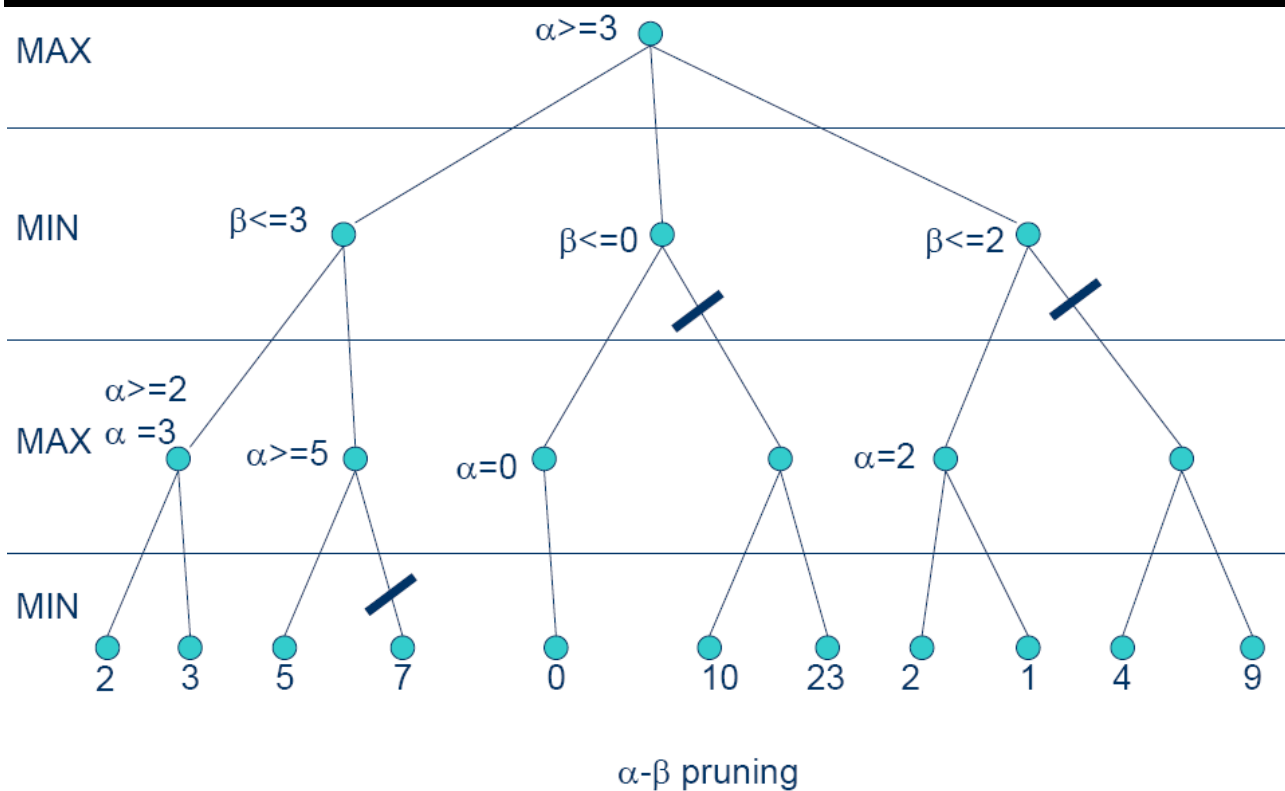
3.4 Thuật toán cắt tỉa Alpha-Beta:

Giới thiệu:

- Mục đích của thuật toán là làm giảm số nhánh trong cây tìm kiếm.
- Không làm ảnh hưởng đến sự đánh giá của đỉnh đang xét.
- Phương pháp cắt tỉa này cho phép cắt tỉa các nhánh không cần thiết cho sự đánh giá của đỉnh đang xét.

Chiến lược:

- Tìm kiếm theo kiểu depth-first
- Nút MAX có 1 giá trị (luôn tăng)
- Nút MIN có 1 giá trị (luôn giảm)
- Tìm kiếm có thể kết thúc dưới bất kỳ:
 - Nút MIN nào có **Beta** \leq **Alpha** của bất kỳ nút cha MAX nào
 - Nút MAX nào có **Alpha** \Rightarrow **Beta** của bất kỳ nút cha MIN nào
- Cắt tỉa - thể hiện mối quan hệ giữa các nút ở mức n và $n+2$, mà tại đó toàn bộ cây có gốc tại mức $n+1$ có thể cắt bỏ



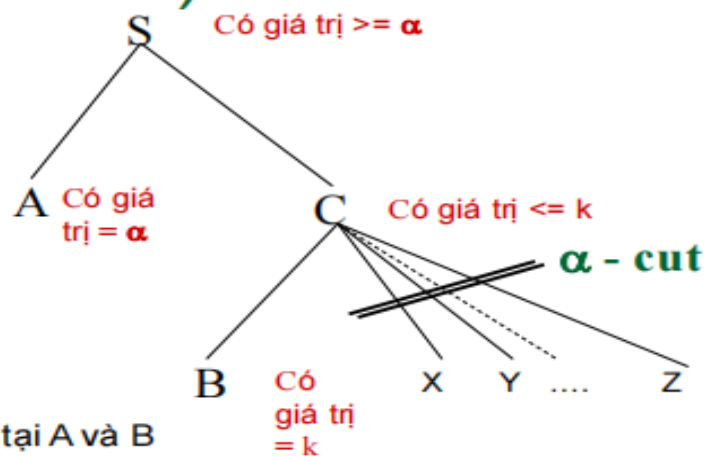
Hình 21. Ví dụ thuật toán cắt tỉa Alpha-Beta

Cắt tỉa α (vị trí MAX)

MAX

MIN

MAX



Điều kiện 1: Chỉ cần biết giá trị tại A và B

Điều kiện 2: Giá trị A > giá trị B

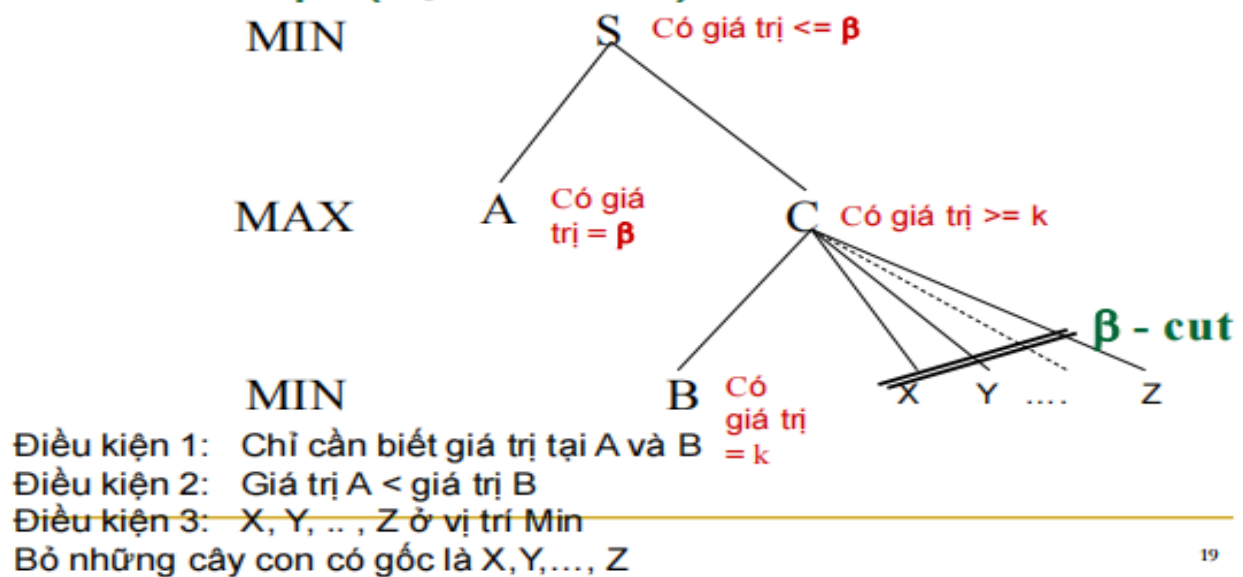
Điều kiện 3: X, Y, .., Z ở vị trí Max

Bỏ những cây con có gốc là X, Y, ..., Z

18

Hình 22 Ví dụ cắt tỉa Alpha

Cắt tỉa β (vị trí Min)



19

Hình 23 Ví dụ cắt tỉa Beta

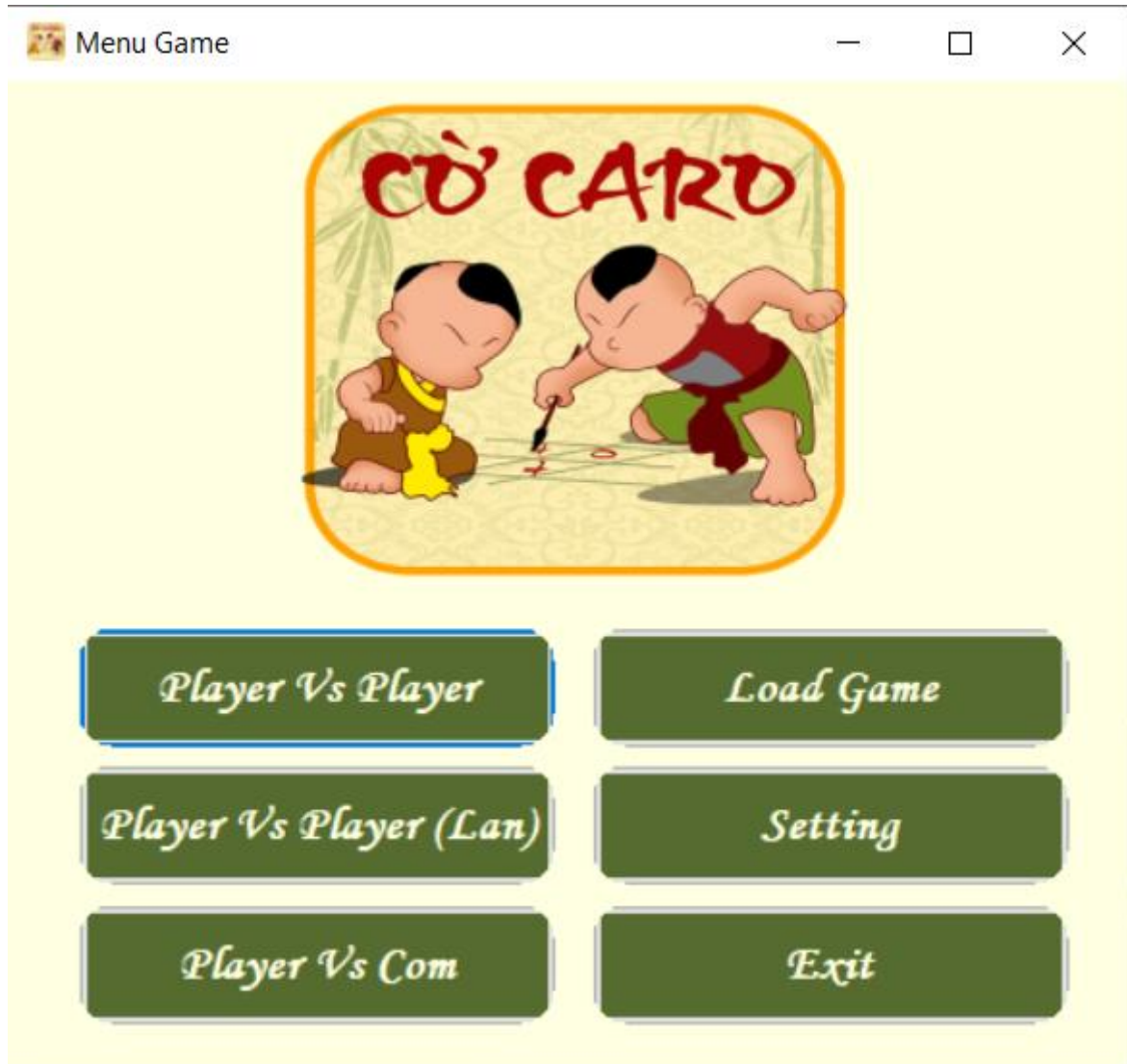
3.5 So sánh Minimax và Alpha-Beta:

Thuật toán **Alpha-Beta** làm giảm số nút cần thiết của việc tìm kiếm để không lãng phí thời gian tìm kiếm những nước đi đã bất lợi rõ cho người chơi.

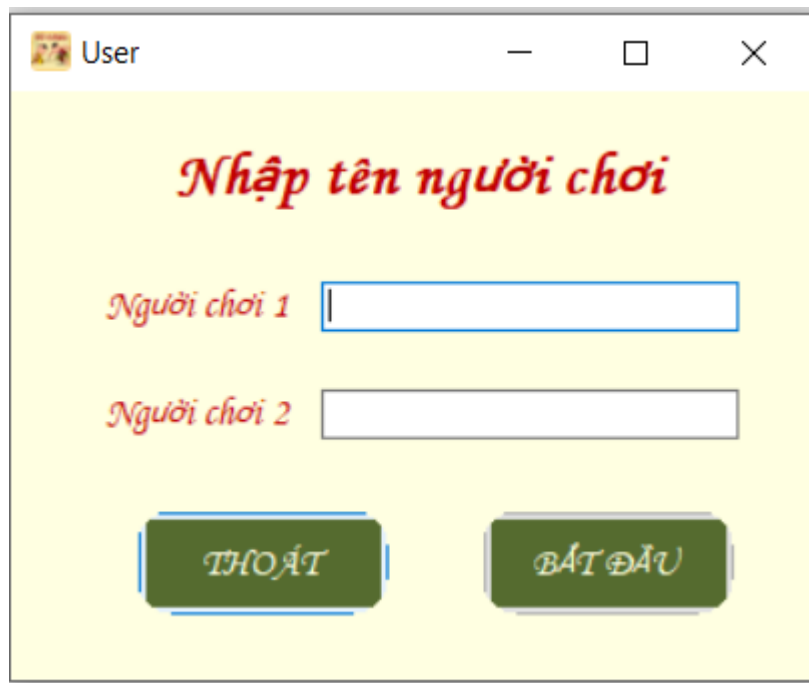
Giải thuật **Alpha –Beta** cải tiến so với **Min – Max** bằng cách thêm vào 2 tham số là alpha và beta. Chúng cho biết các giá trị nằm ngoài khoảng $[\alpha, \beta]$ là các điểm không cần xem xét nữa. Thủ tục Alpha – Beta được bắt đầu tại nút gốc với giá trị của alpha là $-\infty$ và beta là $+\infty$. Thủ tục sẽ tự gọi đệ quy chính nó với khoảng cách giữa các giá trị alpha và beta ngày càng hẹp dần.

CHƯƠNG 3: THIẾT KẾ GIAO DIỆN

I. Giao diện chương trình:

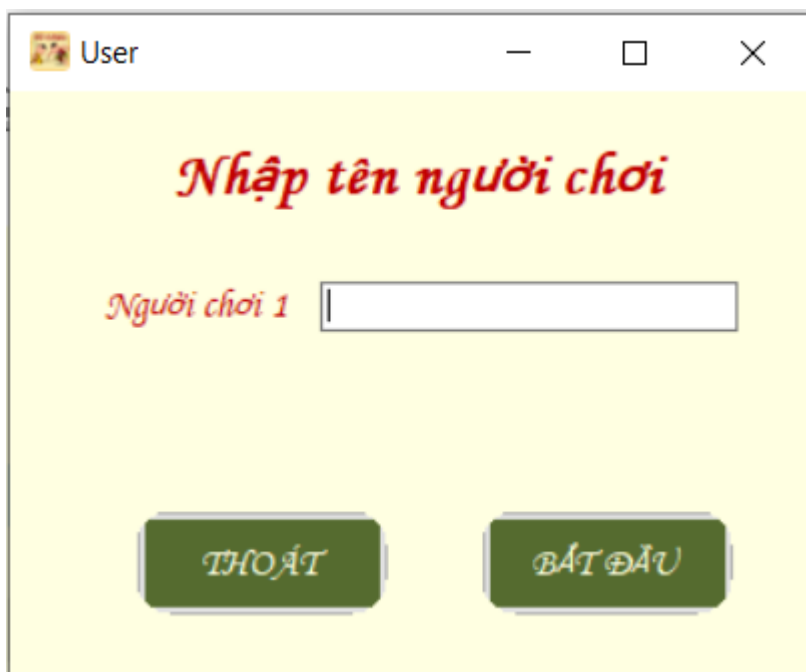


Hình 24 Menu Game



The screenshot shows a standard Windows application window with the title bar 'User'. The main content area has a yellow background. At the top, the text 'Nhập tên người chơi' is displayed in a large, red, stylized font. Below this, there are two input fields. The first is labeled 'Người chơi 1' in red, and the second is labeled 'Người chơi 2' in red. Both labels are in a stylized font. At the bottom of the window, there are two green buttons with white text. The left button is labeled 'THOÁT' and the right button is labeled 'BẮT ĐẦU'. Both buttons have a slight 3D effect.

Hình 25 Nhập tên khi người chơi với người

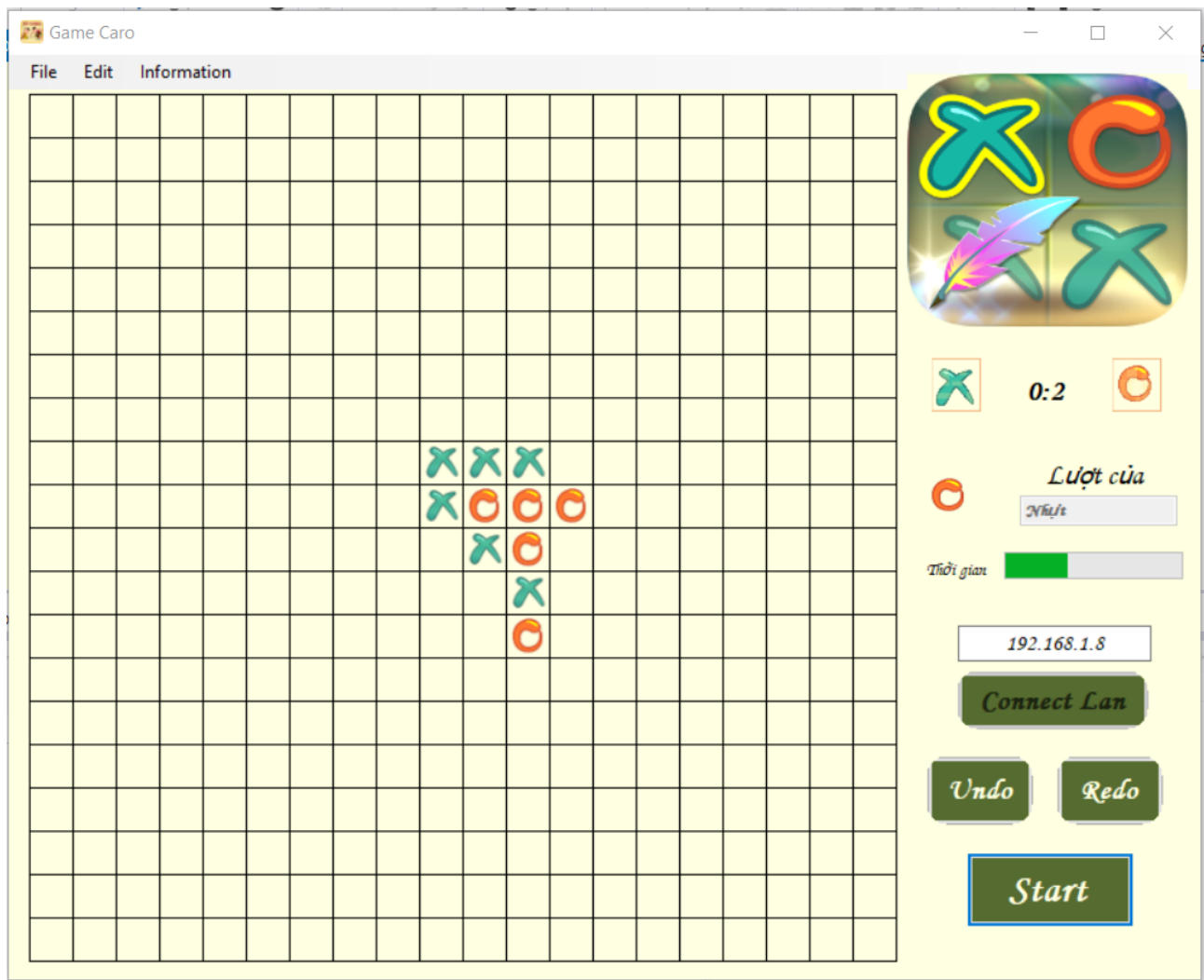


This screenshot is similar to the previous one, but it only shows one input field labeled 'Người chơi 1' in red. The rest of the interface, including the title bar 'User', the yellow background, the red title 'Nhập tên người chơi', and the two green buttons 'THOÁT' and 'BẮT ĐẦU', remains the same.

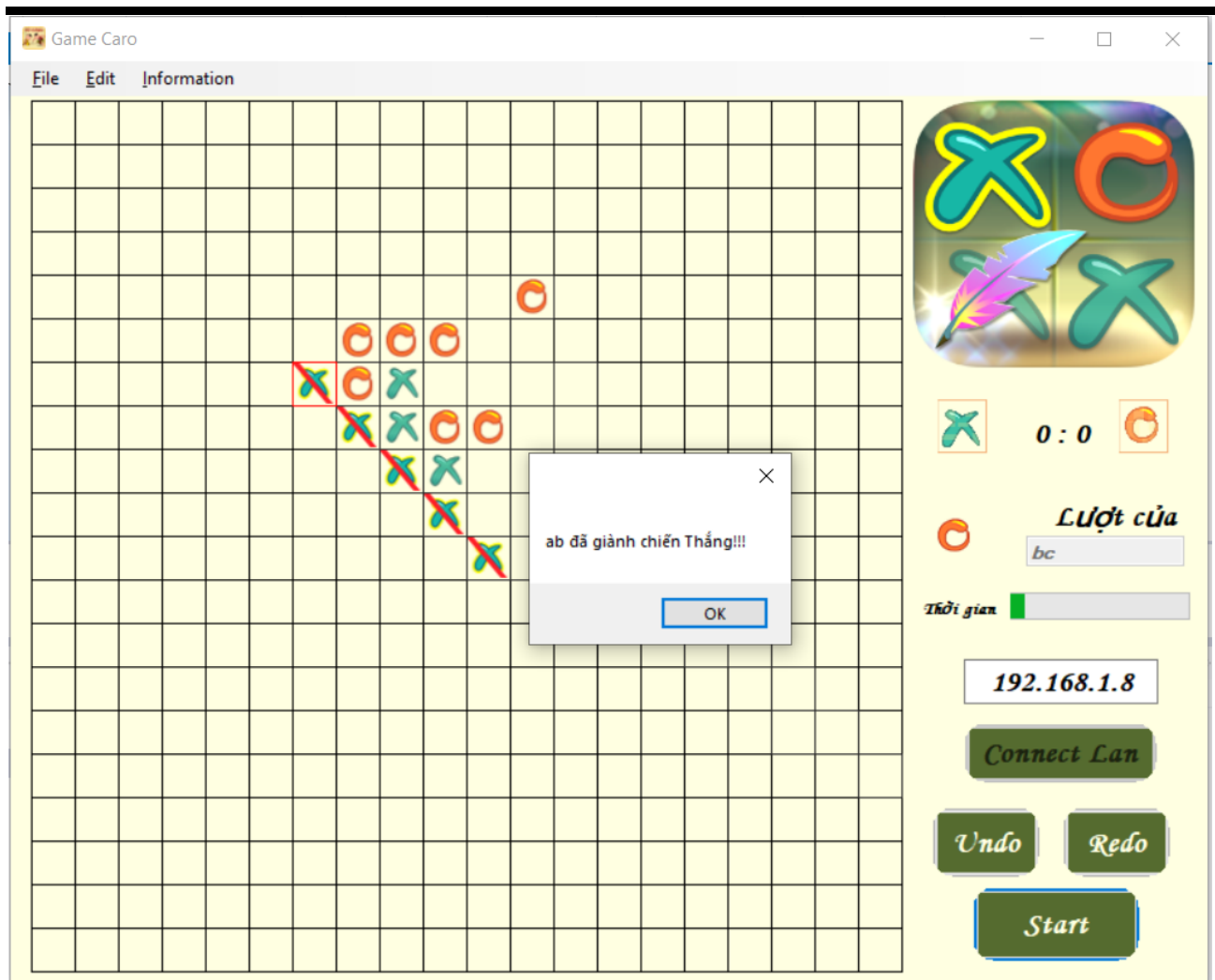
Hình 26 Nhập tên khi chơi với máy



Hình 27 Setting



Hình 28 Màn hình chính



Hình 29 Màn hình khi thắng/thua

CHƯƠNG 4: KẾT LUẬN

Sau quá trình thực hiện đồ án nhóm chúng em có những ưu điểm và nhược điểm như sau:

Ưu điểm:

- Giao diện đơn giản, trực quan, dễ thao tác với người dùng lần đầu.
- Viết trên ngôn ngữ C# tương đối phổ biến nên dễ chỉnh sửa về sau.
- Có đầy đủ nhưng chức năng cơ bản của Game.

Nhược điểm:

- Chưa khắc phục được hoàn toàn lỗi khi sử dụng phần mềm
- Giao diện, cách sắp xếp dữ liệu chưa chuyên nghiệp
- Chưa có sao lưu và phục hồi dữ liệu...
- Thuật toán cho máy chưa khắc phục được việc khi người chơi áp dụng 1 thế cờ chiến thắng vào nhiều ván cờ thì máy vẫn lặp đi lặp lại các nước cờ.

TÀI LIỆU THAM KHẢO

<http://docshare02.docshare.tips/files/24124/241249261.pdf>

[https://vallicon.com/post/t%C3%ACm-ki%E1%BA%BFm-v%C3%A9t-c%E1%BA%A1n-\(complete-search\)-DPZNpnDEwoq](https://vallicon.com/post/t%C3%ACm-ki%E1%BA%BFm-v%C3%A9t-c%E1%BA%A1n-(complete-search)-DPZNpnDEwoq)

<https://www.howkteam.vn/d/chat-lan-su-dung-socket-trong-c-288>

https://vi.wikipedia.org/wiki/C%E1%BB%9D_ca-r%C3%B4