



**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP.HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO ĐỒ ÁN**

**HỌC PHẦN : TRÍ TUỆ NHÂN TẠO**

***Chủ Đề:***

**XÂY DỰNG CHƯƠNG TRÌNH CỜ CARO**

Giảng viên hướng dẫn: **ThS.Nguyễn Thanh Tùng**

Sinh viên thực hiện: MSSV: Lớp:

1. Nguyễn Văn Phụng 2011060853 20DTHC2
2. Hoàng Khánh Nam 2011063919 20DTHC2
3. Phạm Hoàng Bảo 2011060065 20DTHC2

**TP.HỒ CHÍ MINH, 2022**

MỤC LỤC

[**TP.HCM, Tháng 12 năm 2022 1**](#_Toc23944)

[**Bảng phân công công việc:**](#_Toc22150) **3**

[**Lời nói đầu. 4**](#_Toc19866)

[**I. Yêu cầu bài toán. 5**](#_Toc11877)

[**II. Phân tích giải quyết bài toán 5**](#_Toc12339)

[**1.Phân tích yêu cầu 5**](#_Toc11029)

[**2. Phương pháp giải quyết 8**](#_Toc30174)

[**2.1. Tìm kiếm nước đi: 8**](#_Toc8711)

[**Cây trò chơi 9**](#_Toc3667)

[**2.2 Kĩ thuật lượng giá 11**](#_Toc24646)

[**3 Xây dựng các lớp 14**](#_Toc10841)

[**III. Trích dẫn tài liệu tham khảo. 19**](#_Toc16138)

**Bảng phân công công việc:**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thành viên** | **Chi tiết công việc** | **Tỉ lệ đóng góp** |
| **1** | **Nguyễn Văn Phụng** | * Thiết kế giao diện * Tìm hiểu phần phân tích thiết kế hệ thống. * Code các form * Tìm hiểu tối ưu code * Hỗ trợ làm báo cáo | **33,3%** |
| **2** | **Hoàng Khánh Nam** | - Làm báo cáo  - Thiết kế giao diện   * Hỗ trợ tối ưu code * Code các form | **33,3%** |
| **3** | **Phạm Hoàng Bảo** | - Làm báo cáo  - Thiết kế giao diện  - Hỗ trợ thiết kế giao diện  - Hỗ trợ tối ưu code | **33,3%** |

**LỜI NÓI ĐẦU**

Cờ Caro là một trong những trò chơi rất phố biến, đặc biệt là trong giới học sinh, sinh viên. Đây cũng là một trò chơi chúng em rất thích, chính vì vậy chúng em đã chọn đề tài làm game cờ caro cho môn Trí tuệ nhân tạo.

Trong quá trình hoàn thành đề tài này, chúng em đã tìm hiểu được các thuật toán đã được học trong môn Trí tuệ nhân tạo như thuật toán tìm kiếm nước đi Minimax, giải thuật Alpha-Beta cũng như kỹ năng lập trình ngôn ngữ Java.

Chúng em cũng xin cám ơn sự hướng dẫn tận tình của thầy Nguyễn Thanh Tùng, cả về chuyên môn cũng như định hướng. Vì kiến thức còn hạn hẹp nên trong quá trình thực hiện đề tài không thể tránh khỏi thiêu sót. Vì vậy rất mong nhận được sự góp ý của thầy để đề tài có thể hoàn thiện hơn nữa.

**I. YÊU CẦU BÀI TOÁN**

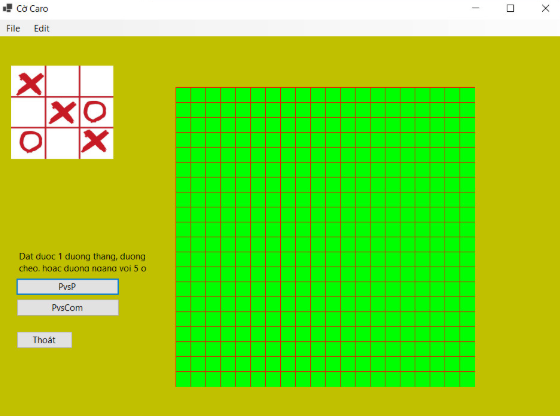
Xây dựng một bản cờ có kẻ các ô vuông với kích thước 20x20. Có 2 quân cờ là trắng, đen

Người chơi có thể đánh với máy. Người thắng là người đi được 5 quân cờ cùng kiểu trên hàng dọc, hàng ngang hoặc đường chéo. Hai người hoà nhau khi bàn cờ hết chỗ đánh mả vẫn chưa phân được thắng bại

**II. PHÂN TÍCH GIẢI QUYẾT BÀI TOÁN**

**1. Phân tích yêu cầu**

Mô phòng bàn cờ



Bàn cờ (Board) bao gồm các ô cờ ( Pos) được đặt trong một mảng 2 chiều (kích thước a x b)

Trong mỗi Pos có thể xác định được:

Vị trí pos ( Row, collumme)

Trạng thái pos (Status) Bao gồm đang trống (0) nước đi của đối thủ (2) hoặc nước đi của máy (1)

Độ nguy hiểm của ô cờ tuỳ theo trạng thái pos và có thể thay đổi được

Đảnh giá giá trị các pos

Giống như trong thực tế, người chơi thường đánh giá một số nước cờ là nguy hiểm bình thường hoặc ít nguy hiểm, máy tính cũng đánh giá nhưng cụ thẻ hơn bằng các con số

**2. Phương pháp giái quyết**

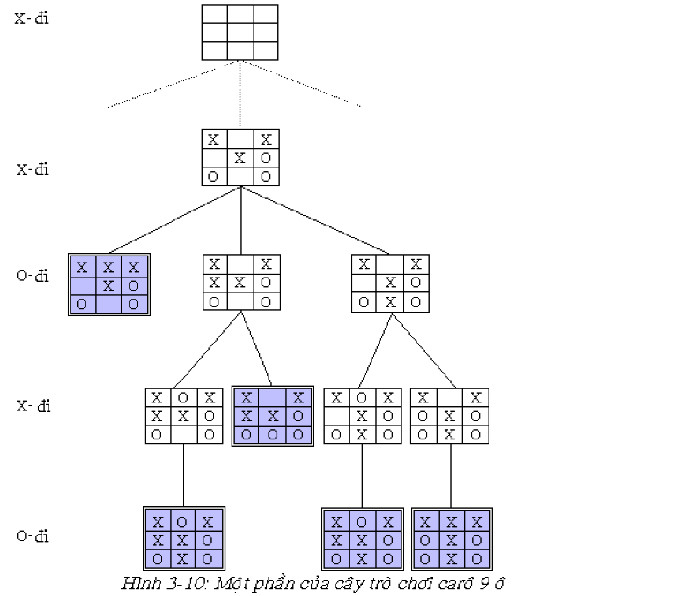
**2.1 Tìm kiếm nước đi**

Giới thiệu về không gian tìm kiểm

Trong trò chơi Caro, cứ sau mỗi nước cờ, mỗi đối thủ sẽ chọn ra từ những ô trống để đi, do đó, sau 1 mỗi nước đi thì số ô trồng còn lại sẽ giảm. Như vậy, việc tìm nước đi tiếp theo cho trạng thái có sẵn chỉ là việc tìm kiểm những ô trỗng còn lại, đồng thời, không gian tìm kiếm sẽ thu hẹp theo số nước đi đã tạo.

Không gian chọn nước đi từ mỗi trạng thái ban đầu là hữu hạn, nhưng không gian tìm kiếm 1 nước đi dẫn dến chiến thắng là rất lớn.Do dó ta không thể vét sạch không gian tìm kiếm nước đi này mà ta phải giới hạn không gian tìm kiếm.

Một không gian tìm kiếm có thê hiện theo 1 cây đa phân vả được gọi là cây tìm kiểm hay cây trò chơi.



**Cây trò chơi**

Dựa vào cái cây trò chơi đã định nghĩa ở trên, việc tìm kiếm nước đi là chọn 1 nút trên cây ( ở mức 1) sao cho nước đó là tốt. Theo thông thường khi chơi, một nước đi tốt hay không là phụ thuộc vào khả năng dành chiến thắng là cao hay thấp sau khi nước đi này được đi.

Do đó, muốn chọn 1 nước đi tốt thì nếu chỉ dựa vào thế cờ hiện tại là chưa đủ, mà phải biết thông tin của những thế cờ sau khi chọn nước này để đi.

Chiến lược minimax để tìm kiếm nước di

Chiến lược này được xác định bằng cách xét giá trị MINIMAX đối với mỗi nút trong cây biểu diễn trò chơi.

MAX chọn nước đi ứng với giá trị MINIMAX cực đại (để đạt đượcgiátrị cực đại của hàm mục tiêu) đạt được giá trị cực đại của hàm mục tiêu)

Ngược lại, MIN chọn nước di ứng với giá trị MINIMAX cực tiêu

Giải thuật tìm kiếm MINIMAX vấp phải vẫn đề bùng nỗ (mức hảm mũ) các khả năng nước đi cần phải xét — không phủ hợp với nhiều bài toán trò chơi thực

Chúng ta có thể cắt tỉa (bỏ đi - không xét đến) một số nhánh tìm kiếm trong cây biêu diễn trò chơi

ąβ

Phương pháp cắt tia ą-β (Alpha-bela prumning)

Ý tưởng: Nếu một nhánh tìm kiếm nảo đó không thẻ 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 cất ta các nhánh tìm kiếm không ảnh hưởng đến kết quả cuối cùng ą là giá trị của nước di tốt nhất dối với MAX (giá trị tôi đa) tính đến hiện tại với nhánh tìm kiếm. Nếu v là giá trị tồi hơn ą, MAX sẽ bỏ qua nước di ứng với v => Cắt tia nhánh ứng với v β được định nghĩa tương tự đối với MIN.

Đối với các trò chơi có không gian trạng thái lớn, thì phương pháp cắt tỉa ą-β vẫn không phù hợp. Không gian tìm kiếm (kết hợp cắt ta) vẫn lớn

Có thể hạn chế không gian tìm kiếm bằng cách sử dụng các tri thức cụ thể của bài toán

Trị thức đề cho phép đánh giá mỗi trạng thái của trò chơi.

Trï thức bổ sung (heuristic) này đóng vai trò tương tự như là hàm ước lượng.

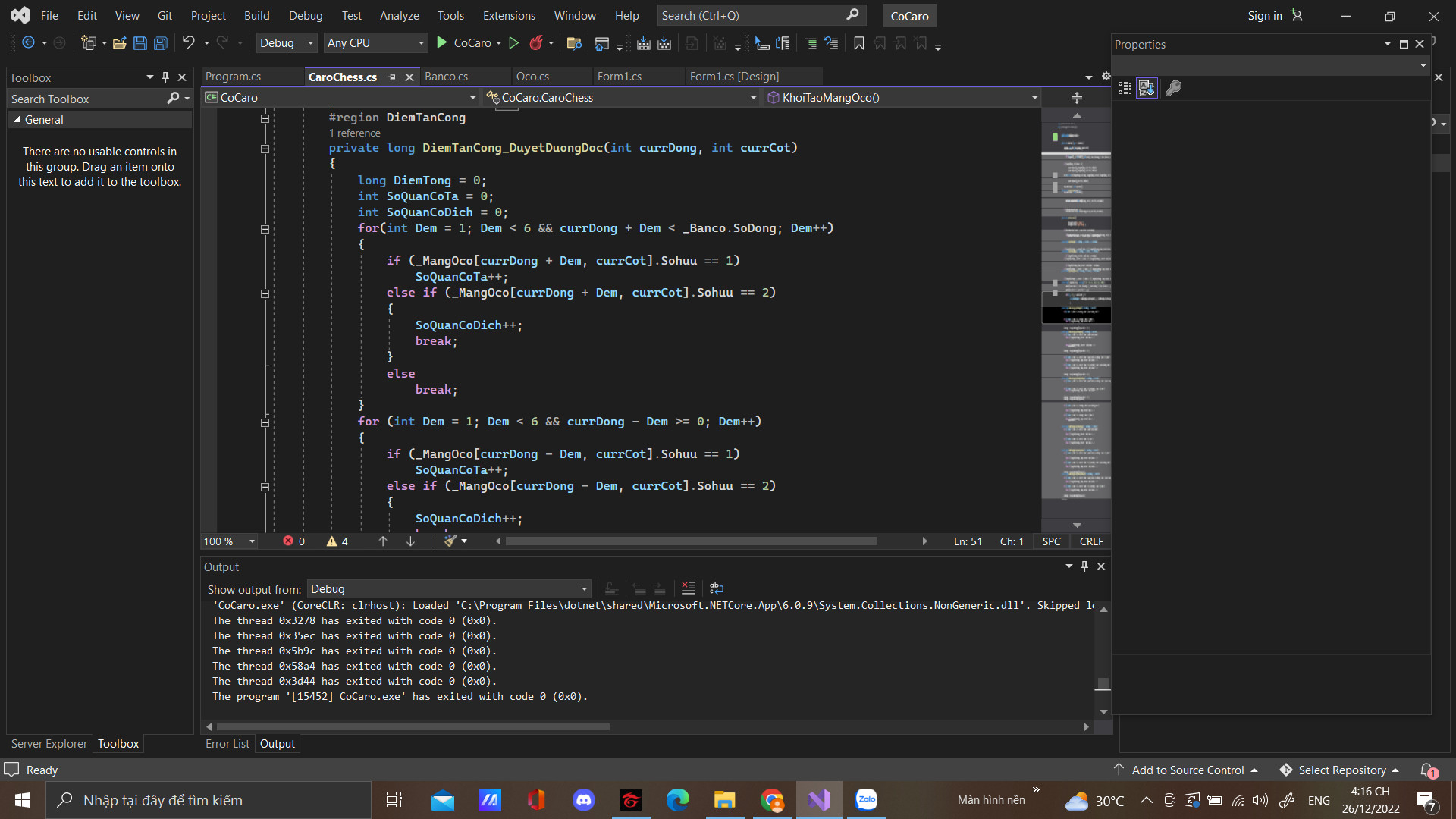
hín) trong giải thuật tìm kiếm A\*

**2.2 Kỹ thuật lượng giá**

Kỹ thuật lượng giá là một kỹ thuật quan trọng trong việc xây dựng trò chơi cờ caro. Kĩ thuật nảy giúp cho điểm trạng thái của bản cờ để từ đó xây dựng trò chơi.

Việc xây dựng hàm lượng giá hợp lý, chính xác sẽ giúp cho hệ thông có đánh giá chính xác về trạng thái bản cờ để đưa ra nước đi thông minh hơn.

Đối với bài toán cờ caro, ta có thẻ đùng 1 hàm lượng giá để đánh giá tính "tốt, xấu" tại 1 thời điểm. Những ô nào ở gần các quân đã đánh trước sẽ được điểm cao hơn. Những ô càng xa thì được cảng ít điểm. Tuy nhiên đây chỉ là Heuristic nên ta phải bổ sung thêm các Heuristie khác nữa, ví dụ vùng có 2, 3, 4... quân liên tiếp thì sẽ được cộng thêm 1 số điểm thưởng nảo đó cho vùng đó dựa vào trọng số quân (tức là nhiều quân liên tiếp thì được cộng nhiều điểm thưởng hơn).



Sau mỗi nước đi, hệ thống sẽ kiểm tra bàn cờ tìm các thế cờ đó rồi tùy vào độ lợi thế đã định trước đề tính ra điểm. Cụ thể là:

\*THI: Trường hợp chắc thắng (+5000 điểm)

{0,1, 1,1,1,1}, {1,0,1,1,1} {1,1,0,1,1}

\* TH2: Trường hợp thuận lợi (+585 điểm).

{0.0, 1,1,1,0}, {0, 1,0, 1, 1,01, {1,0, 1,0, 1,0,1}

\*TH3: Trường hợp bình đẳng ( +9 điểm )

{0,1.1,1,0}, {0,0, 1,1, 1}, {0, 1,0, 1,1}

{1,0, 1,1,0, 1}, {0,0,1,1, 1}, {1,0 1,1,0},

{1/0/0,1,11, {1,0,1,0,1}, {1,0,0, 1,1},

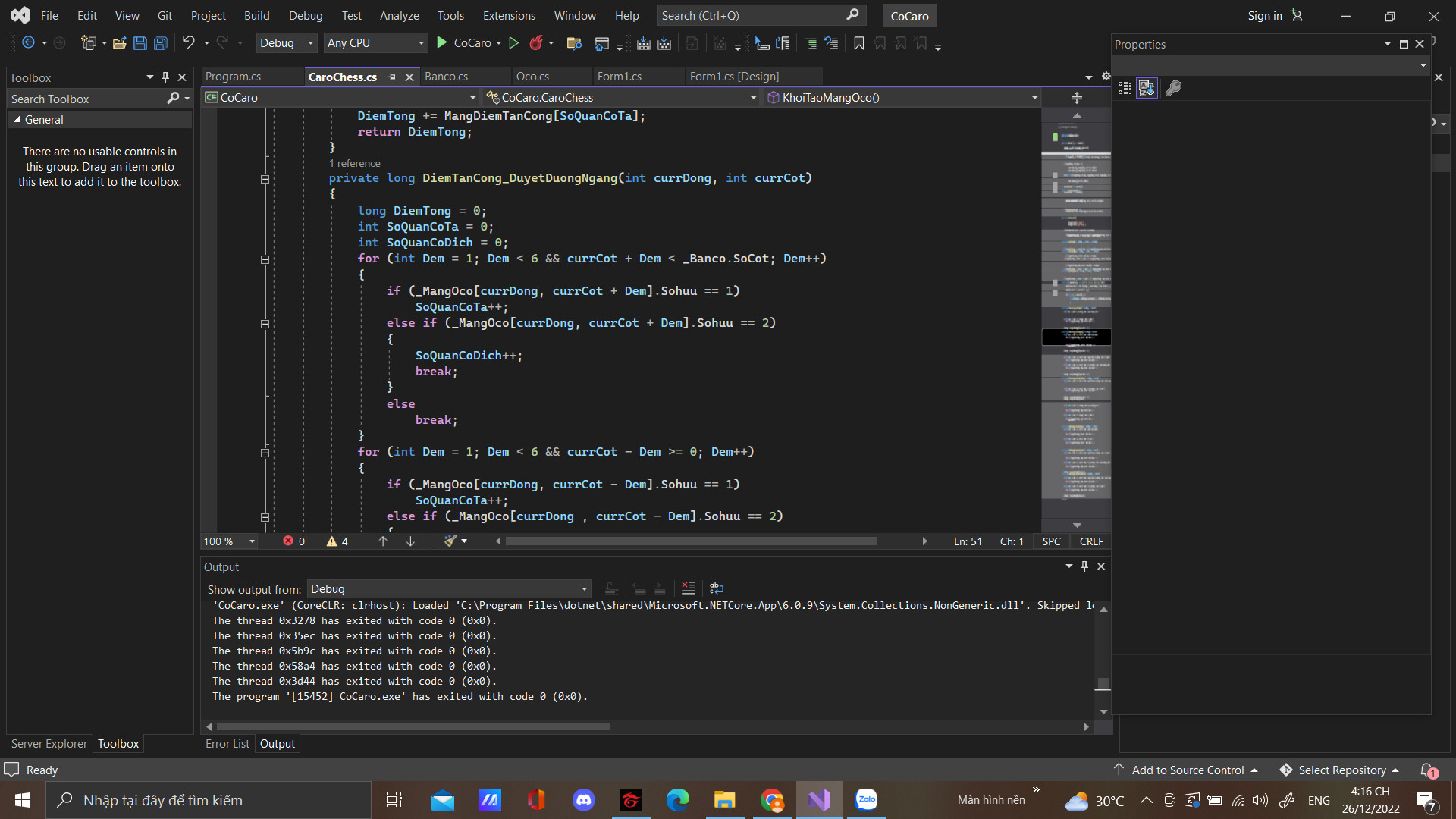
{0.1.0.1,1}

\*TH3: Trường hợp bình thường (+9 điểm)

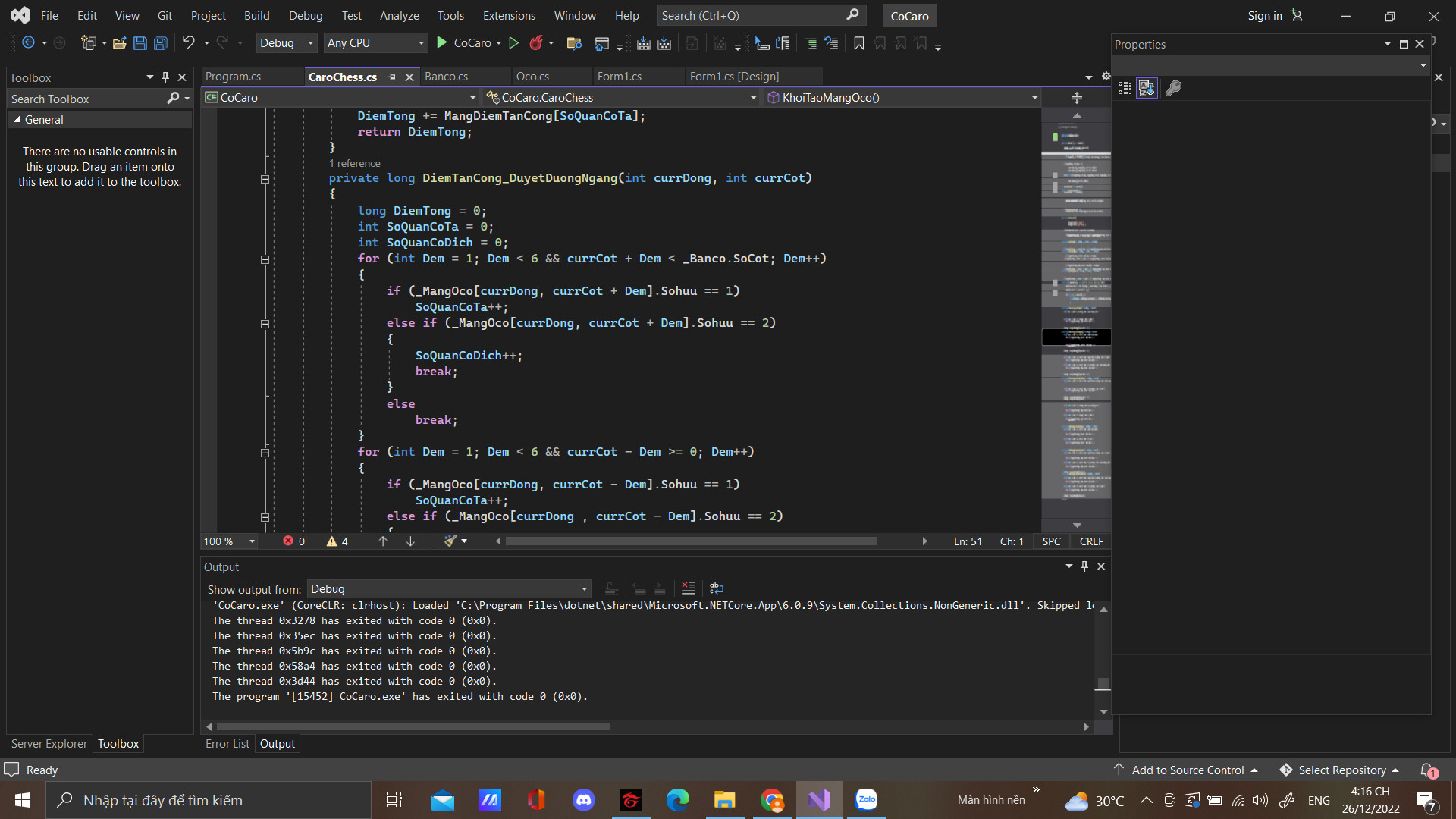
{0,0, 1,1,0, 0}, {0, 1,0, 1, 0,0}

{0, 1,1,0,0, 0}, {0, 1,0, 1,0, 0}, {0, 1, 1,0, 0, 0},

{0.1.0.0, 1,0}



1. **Xây dựng các lớp**



+ Lớp CaroBoard

Biểu diễn các trạng thái của bàn cờ Caro, danh sách các nước dánh. Cập nhật tìm điểm đánh cho người chơi "class Player", Có thể coi như thành phần lưu trữ thông tin của class Player

-public void updateStateOfTloard(Point poin0: cập nhật trạng thái bản cờ. Bỏ điểm đã dánh ra khỏi các đỉnh chưa đánh, cập nhật các điểm còn lại

-public int getMarkOfPointByPlayer(Point p. int player): tínhđiểm cho một pos (một ô trên bàn cờ )

-publie Point getPoinCMaxMark(int điff, int đefendOrAttack): Chọnô có điểm lớn nhất

-public float getAttackDefend(Point point, int player): Tính tỉ lệ điểm của máy và tỉ lệ điểm của người chơi nều đánh thử vào ô point một quân player.

-public int seleetPoint(int diff, int defendOrAttack): Chọn một ô (pos) trong số các ô để đánh, ô này thỏa mãn giải thuật Minimax.

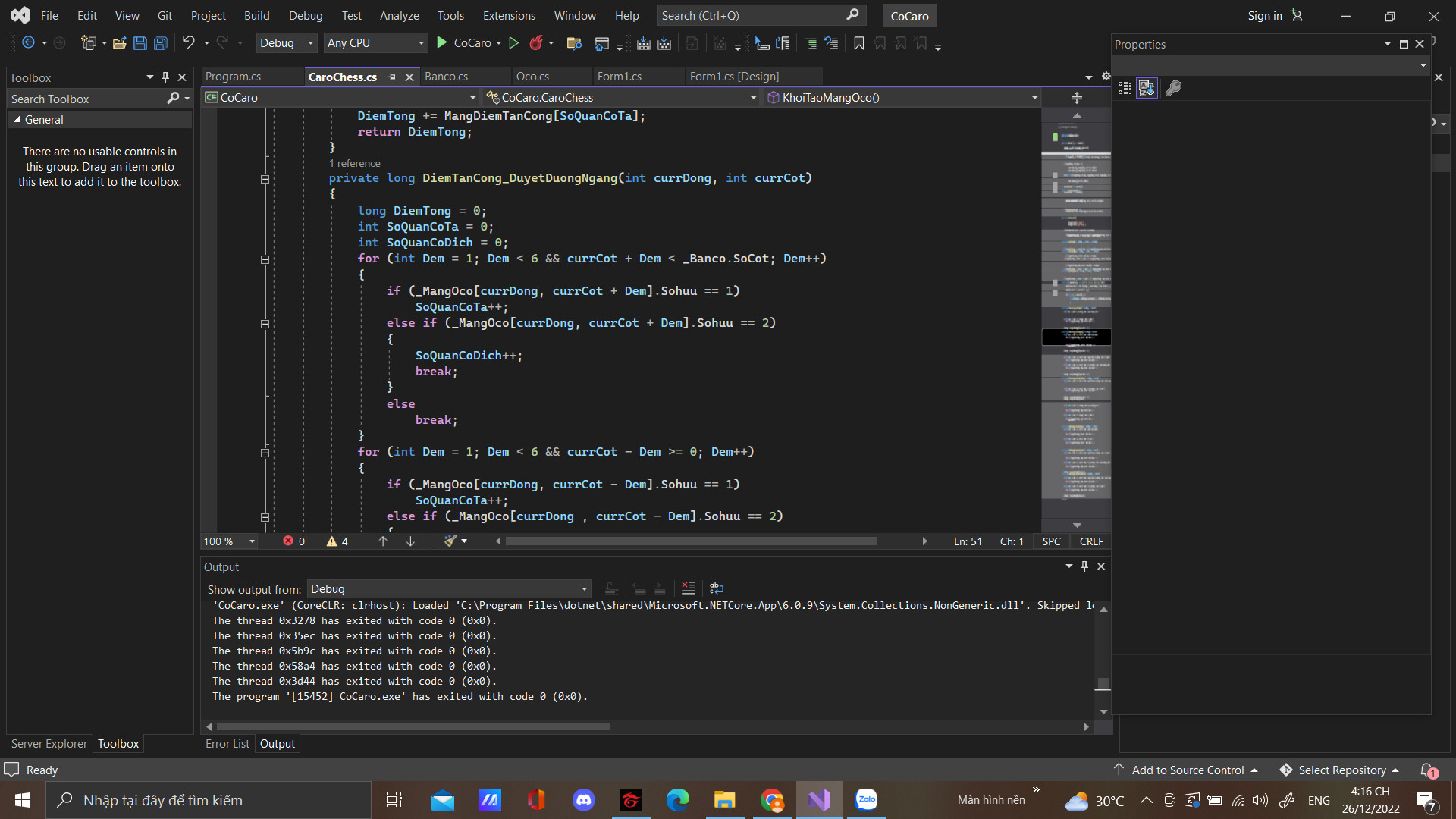
-public int getNormalPos(int diff, int DefendOrAttack): Đánh nước đi nếu điểm các pos bình thường

-publie int getPostOnePlayer(int max, int difP, int defendOrAttaek): chọn nước đi ở thế cơ tấn công

-publie in( getPostOFTWwoPlayer(int max nước đi ở thế cờ phòng thủ

-public int getPosition(int max); chọn nước đi để chặn người chơi không thắng được

-public int cheekITasWiner(Point p): kiểm tra xem ai đã giành phần thắng



\*- Lớp Chesemari:

Danh sách cách thế cờ dùng để tính điểm dược goi trong một

số thú tục của class CaroBoard

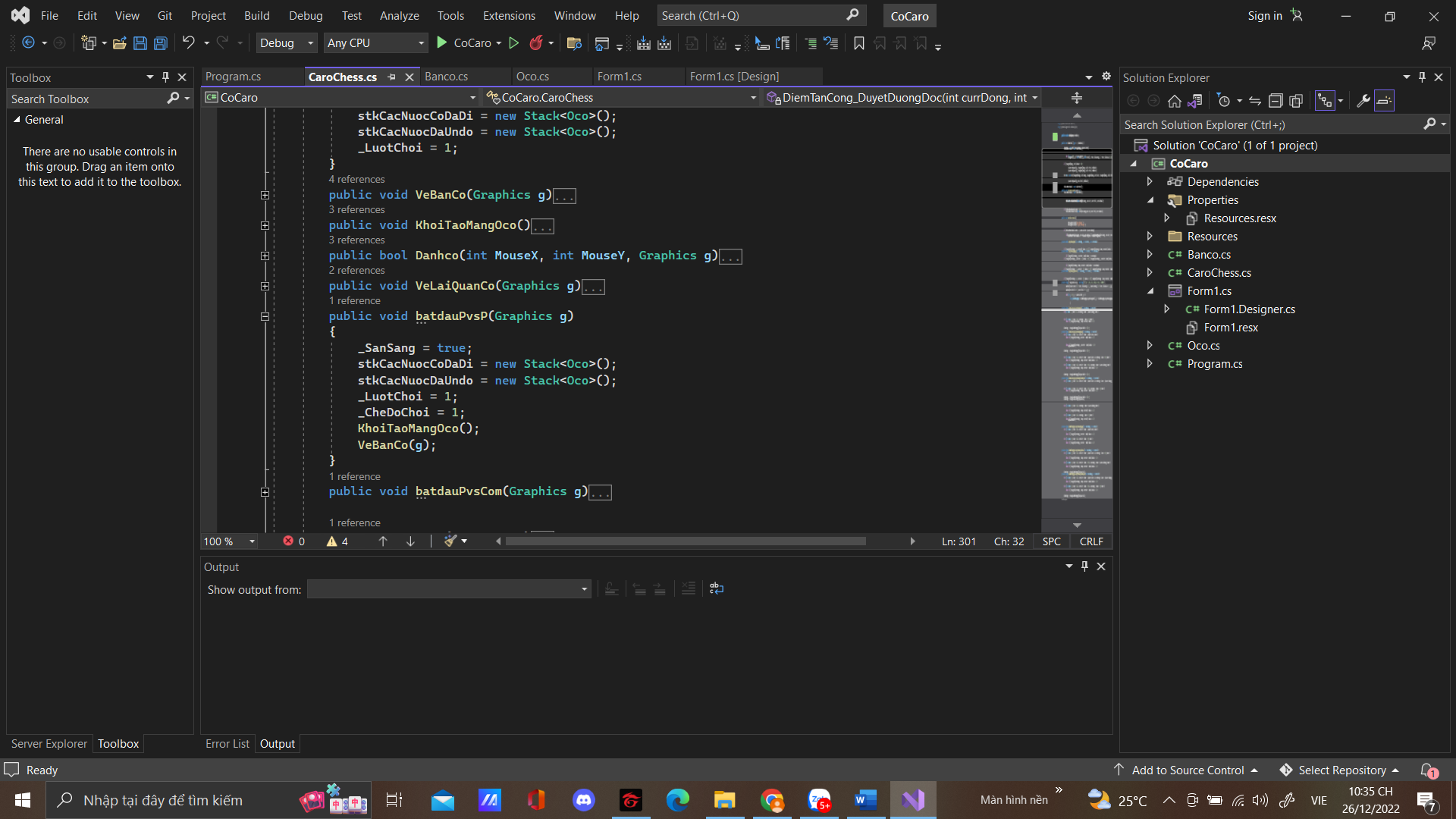
-public static inL MarkOfChessArea(int| | arr, int player): tính điểm của ô, tương ứng với

\*\_ Lớp player

Class Player ; tượng trưng cho một máy tính với nhằm thực hiện nước di, thao tác với người sử dụng.

-public Player(int pDiff, in( pAttaekOrlDepend, inLpDegreeltandom, int width, intheieth)

=publie void reset(int pDiff, int pAttaekOrDepend, int pDegreeRandom) le thông số



TÀI LIỆU THAM KHẢO

1. https://docs.microsoft.com/enus/dotnet/api/system.diagnostics.stopwatch.elapsedmilliseconds?view=netframework-4.85.
2. <https://stackoverflow.com/questions/6068856/c-sharp-swapping-objects-without-a-placeholder6.20>
3. https://www.brainkart.com/article/Best-First-Search--Concept,-Algorithm,-Implementation,-Advantages,-Disadvantages\_8881/2.