**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



**BÁO CÁO ĐỒ ÁN**

**CẤU TRÚC DỮ LIỆU & GIẢI THUẬT**

**GAME CARO HAI NGƯỜI CHƠI**

**(STACK)**

**SVTH1 : NGUYỄN THÀNH TÂM – 17110219**

**SVTH2 : VÕ THỊ KIỀU DIỄM - 17110109**

**GVPT: ThS. TRẦN CÔNG TÚ**

**TP. Hồ Chí Minh, tháng … năm 2018**

MỤC LỤC

[LỜI MỞ ĐẦU 2](#_Toc530425207)

[Danh mục các hình 3](#_Toc530425208)

[Danh mục các bảng 4](#_Toc530425209)

[CHƯƠNG 1 : TỔNG QUAN ĐỒ ÁN 5](#_Toc530425210)

[1.1 Lịch sử hình thành 5](#_Toc530425211)

[1.2 Giới thiệu game caro hai người chơi 5](#_Toc530425212)

[1.3 Lý do, mục đích chọn đề tài 5](#_Toc530425213)

[1.4 Lựa chọn ngôn ngữ cài đặt mô phỏng 5](#_Toc530425214)

[CHƯƠNG 2 : NỘI DUNG 5](#_Toc530425215)

[2.1 Quá trình và công việc thực hiện đồ án 5](#_Toc530425216)

[2.2 Luật chơi 6](#_Toc530425217)

[2.3 Khái quát thuật toán 6](#_Toc530425218)

[2.3.1 Giới thiệu ngăn xếp (stack) 6](#_Toc530425219)

[2.3.2 Ứng dụng 6](#_Toc530425220)

[2.4 Thiết kế giao diện 6](#_Toc530425221)

[2.5 Cách cài đặt và viết chương trình 8](#_Toc530425222)

[2.6 Cài đặt và kiểm lỗi 16](#_Toc530425223)

[2.7 Sản phẩm của đồ án 16](#_Toc530425224)

[CHƯƠNG 3 : KẾT LUẬN 17](#_Toc530425225)

[3.1 Bản kế hoạch hoàn thành 17](#_Toc530425226)

[3.2 Ưu điểm 17](#_Toc530425227)

[3.3 Nhược điểm 17](#_Toc530425228)

[3.4 Đề xuất phương án cải thiện 17](#_Toc530425229)

[TÀI LIỆU THAM KHẢO 18](#_Toc530425230)

LỜI MỞ ĐẦU

Chúng em xin chân thành cảm ơn sự hướng dẫn tận tình của thầy Trần Công Tú, cả về chuyên môn lẫn định hướng cho chúng em. Vì kiến thức còn hạn hẹp và thời gian có hạn nên chúng em chưa thể tối ưu hết các thuật toán sử dụng trong game cũng như không thể tránh khỏi các thiếu sót. Vì vậy rất mong nhận được sự góp ý của thầy để đồ án của chúng em có thể hoàn thiện hơn.

Danh mục các hình

Danh mục các bảng

CHƯƠNG 1 : TỔNG QUAN ĐỒ ÁN

1.1 Lịch sử hình thành

Ban đầu loại cờ này được chơi bằng các quân cờ vây (quân cờ màu trắng và đen) trên một bàn cờ vây (19x19). Quân đen đi trước và người chơi lần lượt đặt một quân cờ của họ trên giao điểm còn trống. Người thắng là người đầu tiên có được một chuỗi liên tục gồm 5 quân hàng ngang, hoặc dọc, hoặc chéo. Tuy nhiên, vì một khi đã đặt xuống, các quân cờ không thể di chuyển hoặc bỏ ra khỏi bàn, do đó loại cờ này có thể chơi bằng giấy bút. Ở Việt Nam, cờ này thường chơi trên giấy tập học sinh (đã có sẵn các ô ca-rô), dùng bút đánh dấu hình tròn (O) và chữ X để đại diện cho 2 quân cờ.

1.2 Giới thiệu game caro hai người chơi

Game Caro (hay còn gọi là Gomuku) là một game thuộc thể loại game đối kháng, trí truệ giữa hai người chơi, bạn có thể chơi nó ở bất cứ đâu, bất cứ lúc nào.

Trước đây trò chơi này cần có giấy kẻ vuông và cây bút viết, nhưng nay Caro được phát triển trên các thiết bị điện tử và không còn xa lạ với bất cứ ai.

1.3 Lý do, mục đích chọn đề tài

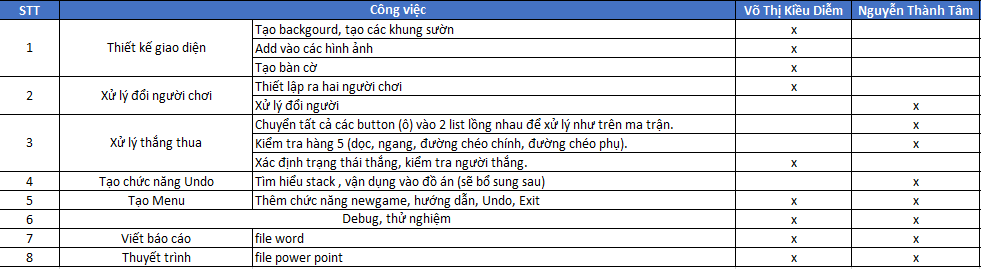
Trong khi học môn cấu trúc dữ liệu, chúng em được tiếp xúc nhiều hơn với các khối kiến thức cơ sở, điều đó giúp chúng em có thể xây dựng các chương trình trên máy tính, áp dụng các kiểu dữ liệu trong thực tế. Thông qua đó chúng em vận dụng kiến thức về ngăn xếp (stack) và hàng đợi (queue) để demo game caro hai người chơi. Với mục đích xây dựng game caro với tính tiện dụng, gần gũi, dễ chơi và tính giải trí cao, phù hợp với mọi lứa tuổi, bám sát kiến thức đã học. Và quan trọng là hiểu rõ hơn về các thuật toán sử dụng, tìm hiểu thêm về thiết kế giao diện trong game, rèn luyện kỹ năng lập trình, làm tiền đề cho các môn học sau này và ứng dụng mai sau. Vì vậy chúng em quyết định chọn đề tài caro hai người chơi để thực hiện đồ án môn học cấu trúc và dữ liệu.

1.4 Lựa chọn ngôn ngữ cài đặt mô phỏng

Vì chương trình mô phỏng của chúng em biễu diễn thuật toán dưới dạng các đối tượng thuộc tính, hành vi rõ ràng nên chúng em quyết định sử dụng ngôn ngữ Microsoft C#.NET với phiên bản .NET Framework 4.5.2; IDE và compiler nằm trong bộ Microsoft Visual Studio 2017 để cài dặt mô phỏng này.

CHƯƠNG 2 : NỘI DUNG

2.1 Quá trình và công việc thực hiện đồ án



2.2 Luật chơi

Mỗi người sẽ lần lượt đi từng nước một, ký hiệu hai bên khác nhau, Ngừng khi đạt điều kiện thắng hoặc hết giới hạn bàn cờ.

Điều kiện thắng : Có 5 quân cờ của người chơi thẳng hàng (ngang, chéo, dọc).

ai có được một hàng 5 nước dọc, ngang hoặc chéo liên tiếp thì sẽ là người chiến thắng!

2.3 Khái quát thuật toán

2.3.1 Giới thiệu ngăn xếp (stack)

Trong khoa học máy tính, một ngăn xếp (còn gọi là bộ xếp chồng, tiếng Anh: stack) là một caấu trúc dữ liệu trừu tượng hoạt động theo nguyên lý "vào sau ra trước" (Last In First Out (LIFO))

Một ngăn xếp là một cấu trúc dữ liệu dạng thùng chứa (container) của các phần tử (thường gọi là các nút (node)) và có hai phép toán cơ bản: push and pop. *Push* bổ sung một phần tử vào đỉnh (top) của ngăn xếp, nghĩa là sau các phần tử đã có trong ngăn xếp. *Pop* giải phóng và trả về phần tử đang đứng ở đỉnh của ngăn xếp. Trong stack, các đối tượng có thể được thêm vào stack bất kỳ lúc nào nhưng chỉ có đối tượng thêm vào sau cùng mới được phép lấy ra khỏi stack.(Nguồn 1)

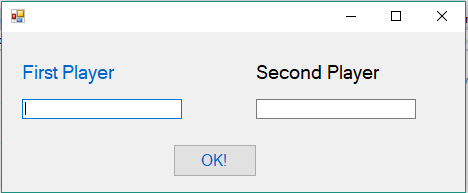
2.3.2 Ứng dụng

Ngăn xếp có nhiều ứng dụng trong khoa học máy tính và cài đặt game. Trong đó, Game caro cũng là một ứng dụng được cài đặt từ stack để lưu trữ dữ liệu bàn cờ.

2.4 Thiết kế giao diện

Chương trình cho phép người chơi có thể chọn trò chơi mới (new game), đi lại (undo), chọn bật nhạc hay tắt nhạc,hướng dẫn chơi game, chọn chặn hai đầu và mô phỏng lại ván cờ. Phía dưới có thể hiện trạng thái đến lượt người nào đi.

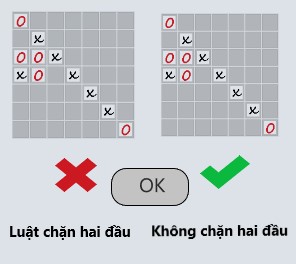
Khi bắt đầu vào game sẽ cho phép chọn người đi trước và nhập tên của hai người chơi.



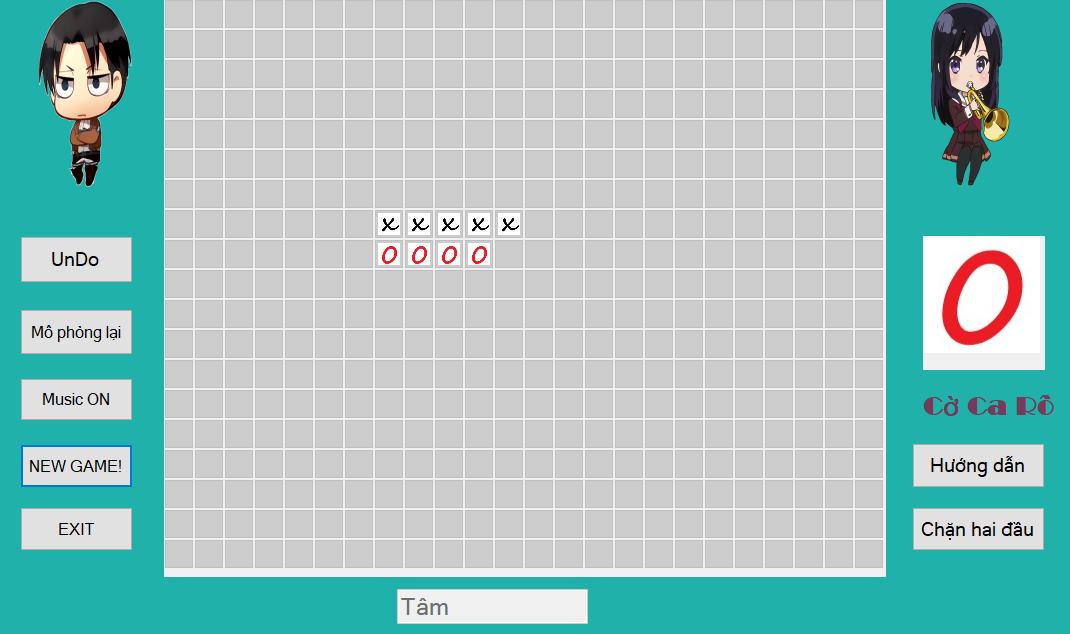
Khi có người thắng thì hiển thị ra thông báo thắng hiệp và cho phép chọn các tùy chọn chơi tiếp hay thoát.

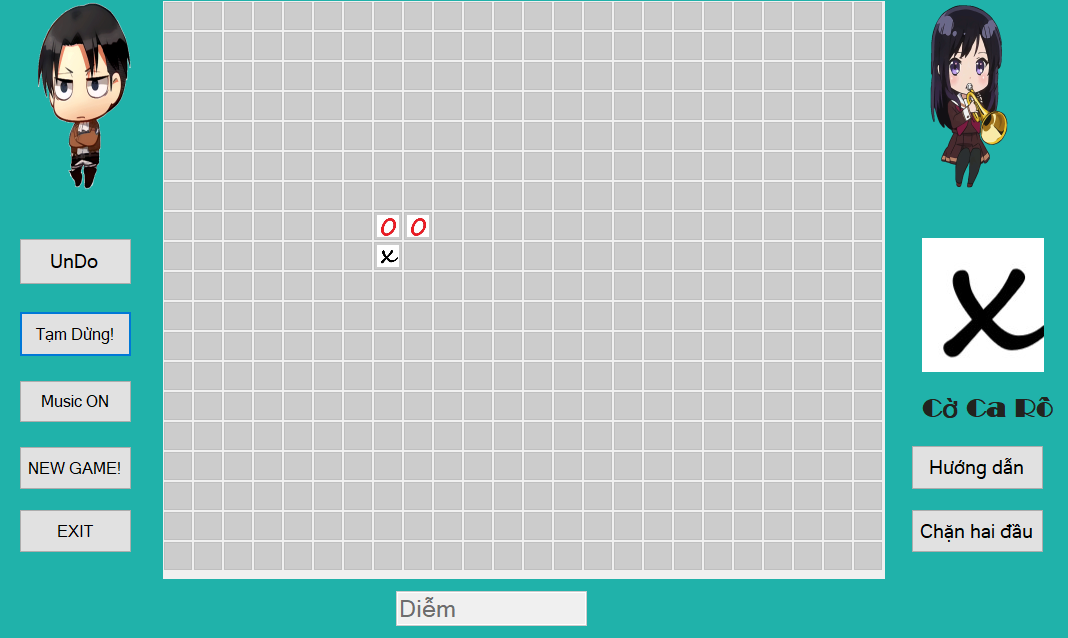


Bên cạnh đó còn bao gồm phần hướng dẫn cho người chơi giúp người chơi dễ dàng nắm được luật chơi game.

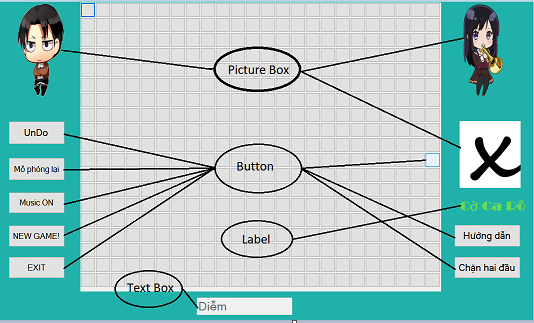


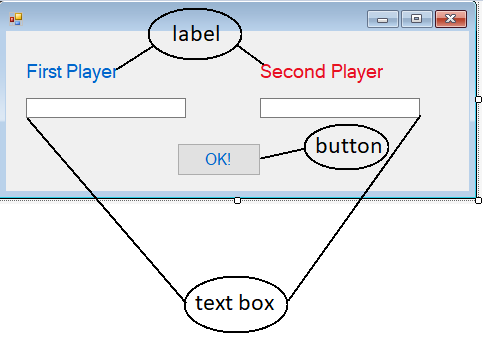
Ngoài ra còn có phần mô phỏng lại ván cờ cho phép người chơi xem lại từng nước cờ mình đã đi.





2.5 Cách cài đặt và viết chương trình





* Đồ án của nhóm em gồm 5 Form và 5 class gồm:

+ FormNhapThongTin (Cho phép nhập tên người chơi)

+ FormGame ( Form chính, bao gồm bàn cờ và những phím chức năng)

+ FormChienThang ( Thông báo đã kết thúc game và một số hiệu ứng)

+ FormHuongDan ( Hướng dẫn chơi game)

+ FormThoat (Hiệu ứng và quyết định có thoát game hay không)

+ Class Player: lớp người chơi bao gồm tên ký hiệu và âm thanh

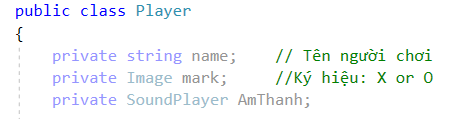
+ Class XuLyBanCo: gồm các hàm xử lý trên một bàn cờ.

+ Class Cons: Gồm các hằng được tạo (chiều dài, rộng bàn cờ, button…)

+ Class MysTack: Tạo stack và các hàm trong nó.

+ Class Program: Chương trình chính.

* Hầu hết các hàm xử lý đều được viết trong class XuLyBanCo, và sau đây là một số hàm quan trọng.
* Đầu tiên tạo ra một class player gồm các thuộc tính như Tên, ký hiệu, âm thanh khi đánh và các propeties.

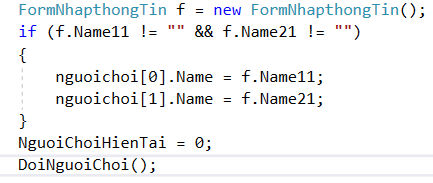


* Sau đó tạo một list player gồm 2 người chơi. (sử dụng list có thể phát triển game cao hơn, như tăng số lượng người chơi…).

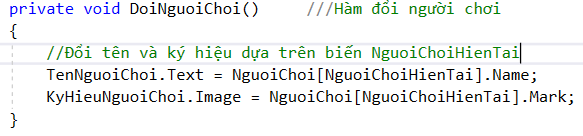


+ Một số khởi tạo:

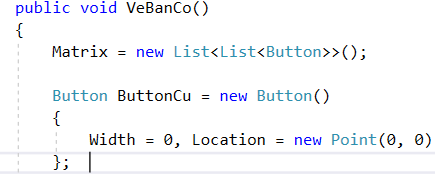




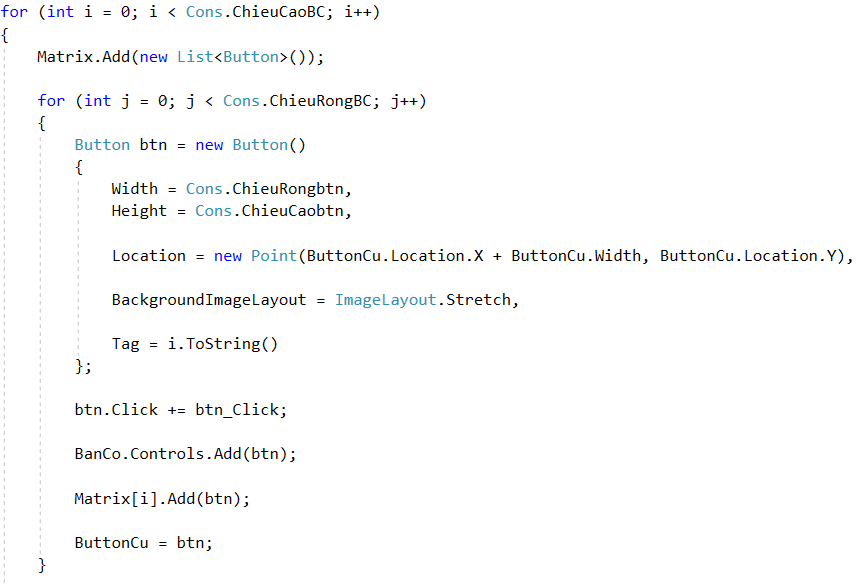
* Lấy thông tin từ form nhập, nếu không nhập thì sẽ không cập nhật tên người chơi (mặc định là *first player* và *second player.*
* Mặc định người chơi thứ nhất đi trước ( X đi trước).
* Sử dụng hàm đổi người chơi để piture mark hiện ảnh khi mới vào game và cho biết ai đi trước.
* Hàm DoiNguoiChoi:



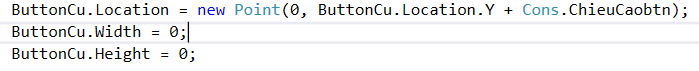
* Tạo bàn cờ: đây là bước quan trọng và tương đối khó.



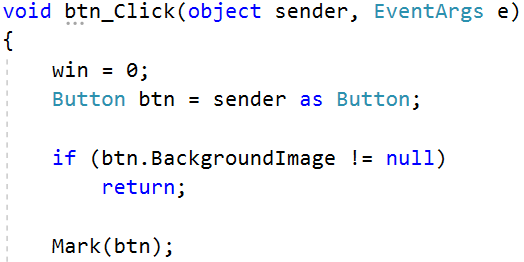
* Đầu tiên, khởi tạo matrix (2 list button lồng nhau).
* Khởi tạo ButtonCu làm mốc để tạo ra các button (ô đánh) trên panel bàn cờ.



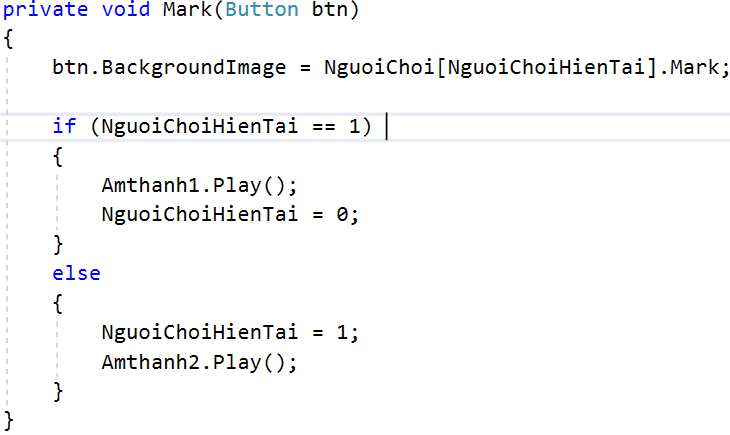
* Sử dụng hai vòng lặp để tạo ra bàn cờ, mỗi ô đánh là một button.
* Vị trí ô được tạo sẽ bằng tọa độ ô cũ cách sang bên phải ô cũ đúng bằng chiều rộng của nó.
* Vị trí hàng của button sẽ được lưu vào Tag của button.
* Tạo event khi click chuột vào button.
* Thêm button mới tạo vào panel bàn cờ và thêm vào Matrix.
* Gán button cũ thành button mới tạo để tạo ra các button tiếp theo.
* Tiếp tục làm như vậy cho đến khi hết bài cờ.



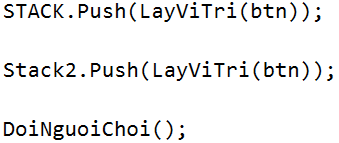
* Sau khi tạo xong một hàng, tiếp tục tạo ra một button ở đầu hàng tiếp theo làm gốc và tiếp tục làm như trên cho đến khi vẽ hết bàn cờ.
* Tạo event cho mỗi button trên bàn cờ:



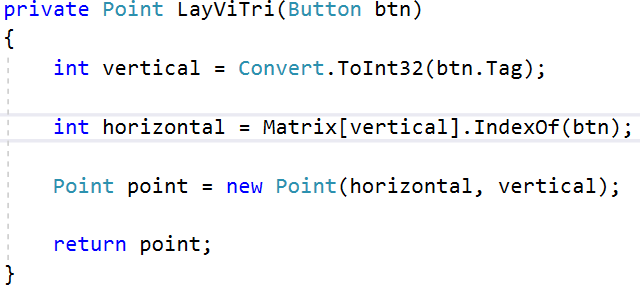
* Lúc đầu khởi tạo win = 0 (chưa thắng).
* Ép kiểu lại với button để biết đã click button nào.
* Kiểm tra xem ô này đã được đánh chưa, nếu rồi thì không được đánh vào.
* Nếu chưa, thì đánh vào ô đó bằng hàm Mark():



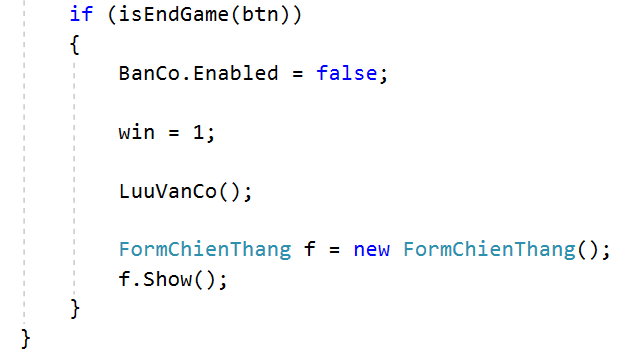
* Hàm Mark thực hiện việc đổi Image của button theo người chơi hiện tại, chạy âm thanh của người đánh và đổi NguoiChoiHienTai sau mỗi lần đánh.



* Mỗi lần đánh, lưu vị trí button đó vào STACK, Stack2 và đổi người chơi.
* Hàm lấy vị trí:



* Lấy vị trí hàng của button từ Tag đã lưu khi vẽ bàn cờ.
* Lấy vị trí cột của button từ hàng thứ vartical thông qua IndexOf(btn).
* Trả về tọa độ của button.



* Kiểm tra nếu kết thúc game:

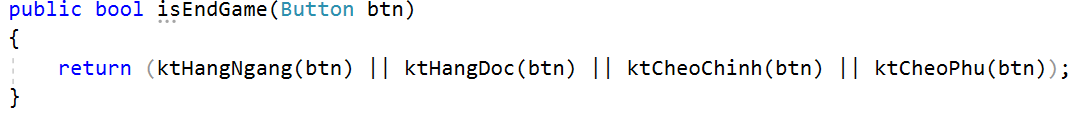
+ Disable bàn cờ.

+ Trạng thái thắng = 1;

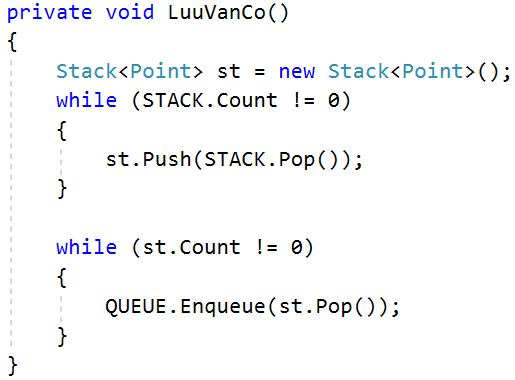
+ Lưu lại ván cờ vào Queue để có thể mô phỏng lại quá trình chơi.

+ Hiện Form chiến thắng lên.

* Hàm isEndGame()



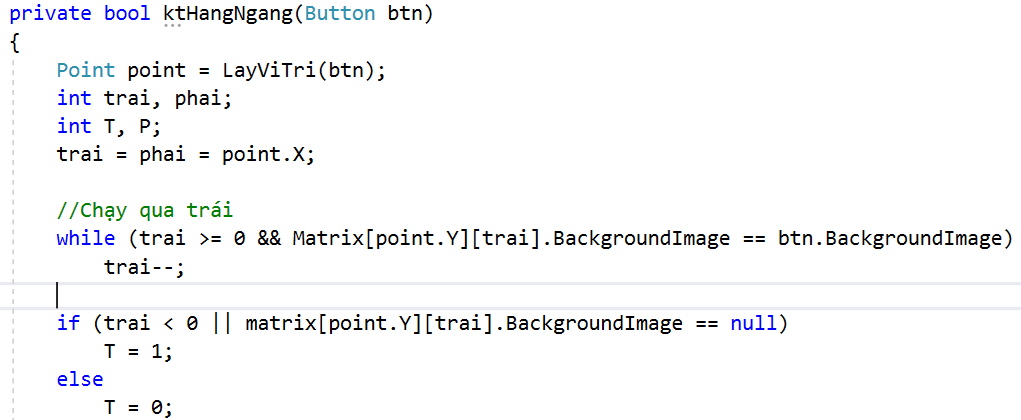
* Kiểm tra xem đã kết thúc game hay chưa, nếu có 1 trong 4 điều kiện là có 5 nước đi liên tục trên hàng dọc, ngang, chéo chính và chéo phụ thì kết thúc.
* Hàm LuuVanCo()

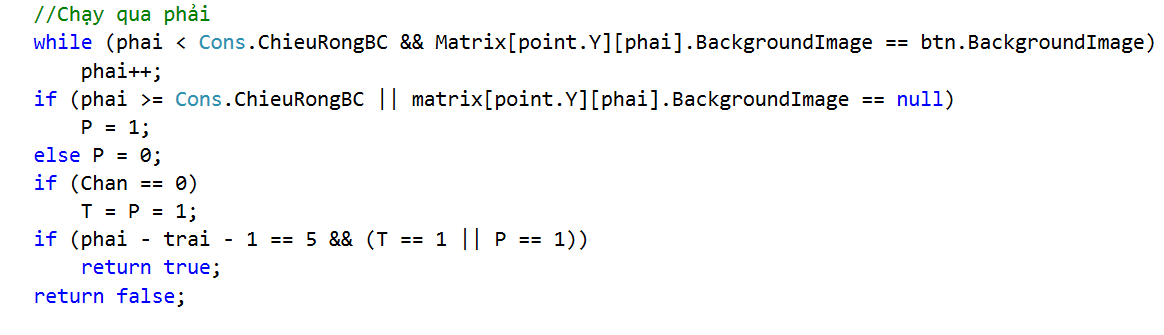


* Ý tưởng: đổ các phần tử của STACK vào một Stack st khác, rồi mới đưa các phần tử vào trong Queue.
* Không đưa trực tiếp từ STACK vào Queue vì như vậy sẽ bị ngược, và cũng không đưa vào Queue trong lúc đánh, vì khi Undo, ván cờ được lưu sẽ không còn chính xác.
* Sai khi lưu ván cờ lại, Form chiến thắng sẽ hiện lên:



* Sau đây là các hàm kiểm tra kết thúc game, gồm kiểm tra hàng dọc, hàng ngang, đường chéo chính và đường chéo phụ, nếu đánh liên tiếp 5 nước liên tục trên một đường thẳng thì kết thúc game.
* Kiểm tra hàng ngang:



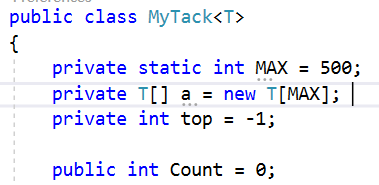


* Luật không chặn hai đầu: ý tưởng là từ ô (button) mới đánh, ta kiểm tra bên trái, và bên phải ô đó có Mark giống với ô này không, nếu giống thì tiếp tục kiểm tra ô kế bên, nếu tổng số ô liên tục đúng bằng 5 thì kết thúc game.
* Luật chặn hai đầu: Kiểm tra thêm 2 ô ở 2 đầu hàng 5 xem có bị chặn bởi người khác không, nếu bị chặn thì không tính thắng.
* Từ button trên Form chính ta chọn chế độ chơi, nếu không chặn hai đầu ( Chặn = 0) thì ta bỏ qua điều kiện chặn hai đầu.

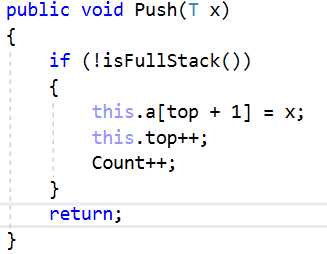
+ Ta kiểm tra hàng dọc, đường chéo chính, chéo phụ tương tự như kiểm tra hàng ngang.

* Class Stack:

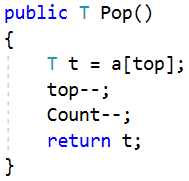
Vận dụng kiến thức đã học từ môn cấu trúc dữ liệu và giải thuật và môn lập trình hướng đối tượng, chúng em đã tạo stack từ Generic.



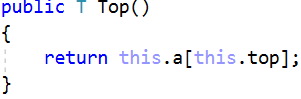
* Các phần tử được lưu vào mảng một chiều với tối đa 500 phần tử.
* Khởi tạo rỗng top = -1;
* Biến Count: để biết số lượng phần tử trong stack.
* Hàm Push():



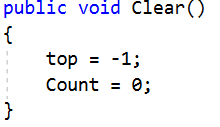
* Kiểm tra xem đã tràn stack chưa, nếu chưa thì thêm vào stack và tăng biến đếm.
* Hàm Pop():



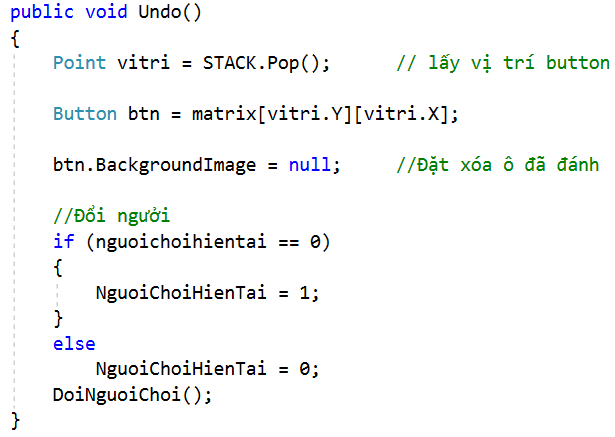
* Lấy ra và trả về giá trị cuối cùng của stack.
* Hàm Top():



* Trả về (Không lấy ra) giá trị cuối của stack.
* Hàm Clear():

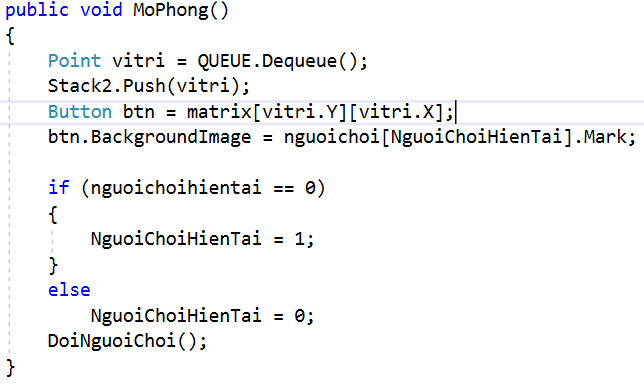


* Đặt lại stack.
* Hàm Undo():

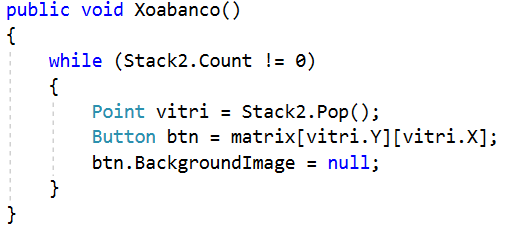


* Lấy ra vị trí đã lưu trong stack.
* Gán lại background của button là null.
* Đổi người chơi.
* Hàm MoPhong():

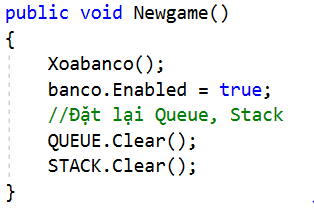
Hàm dùng để mô phỏng lại từng nước mà người chơi đã đánh, sau khi kết thúc game chúng ta có thể xem lại quá trình chơi.



* Lấy vị trí từ Queue và đổi background của button đó.
* Đổi người chơi.
* Hàm Xoabanco():



* Từ stack2 (lưu những nước đã đánh), ta đưa background của các button về null.
* Hàm Newgame():

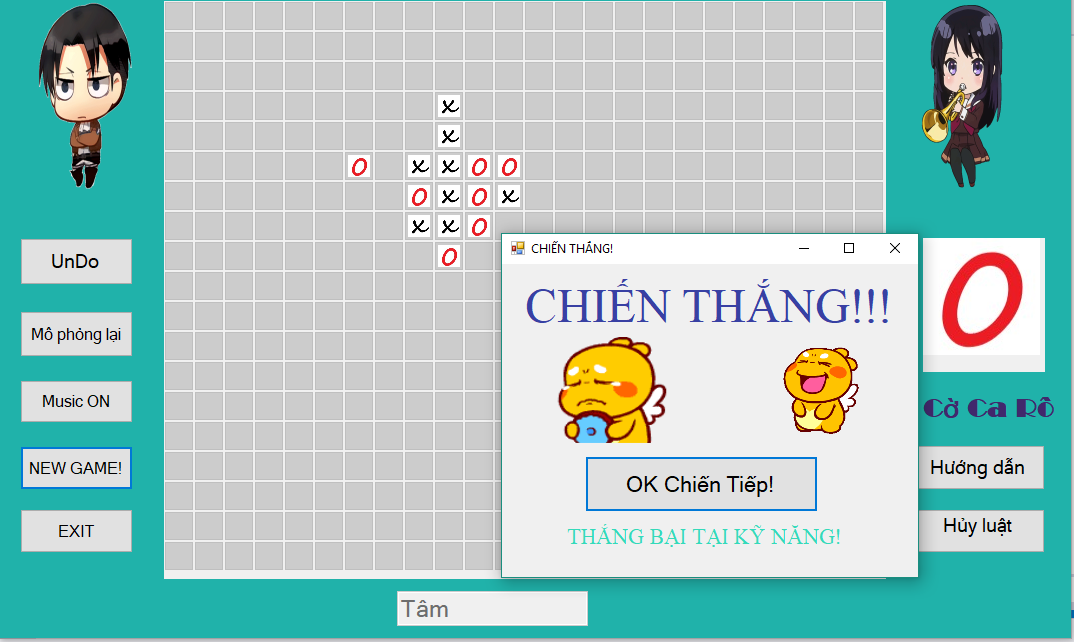


* Xóa bàn cờ, queue và stack.

2.6 Cài đặt và kiểm lỗi

2.7 Sản phẩm của đồ án

Một game caro hoàn chỉnh dành cho hai người chơi.



CHƯƠNG 3 : KẾT LUẬN

3.1 Bản kế hoạch hoàn thành

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Công việc | | Võ Thị Kiều Diễm | Nguyễn Thành Tâm | Ngày bắt đầu dự kiến | Ngày kết thúc dự kiến | Ngày bắt đầu thực tế | Ngày kết thúc thực tế | Ghi chú |
| Thiết kế giao | Tạo background, tạo các khung sườn |  |  |  |  |  |  |  |
| Add vào các hình ảnh |  |  |  |  |  |  |  |
| Tạo bàn cờ |  |  |  |  |  |  |  |
| Xử lý đổi người chơi | Thiết lập ra hai người chơi |  |  |  |  |  |  |  |
| Xử lý đổi người |  |  |  |  |  |  |  |
| Xử lý thắng thua | Chuyển tất cả các button vào 2 list lồng nhau để xử lý như trên ma trận. |  |  |  |  |  |  |  |
| Kiểm tra hàng 5 (ngang, dọc, chéo chính, chéo phụ) |  |  |  |  |  |  |  |
| Xác định trạng thái thắng, kiểm tra người thắng. |  |  |  |  |  |  |  |
| Tạo chức năng Undo | Tìm hiểu stack |  |  |  |  |  |  |  |
| Tạo các phím chức năng | New game, hướng dẫn, mô phỏng, undo, exit, music, chế độ chơi |  |  |  |  |  |  |  |
| Mô phỏng ván cờ | Lưu ván cờ vào Queue và trình chiếu lên bàn cờ |  |  |  |  |  |  |  |
| Debug và thử nghiệm | |  |  |  |  |  |  |  |
| Viết báo cáo | File Word |  |  |  |  |  |  |  |
| Thuyết trình | File Power Point |  |  |  |  |  |  |  |

3.2 Ưu điểm

Sử dụng hợp lý các chức năng của stack (ngăn xếp) vào cài đặt game caro hai người chơi với mục đích lưu trữ các nước cờ của người chơi kết hợp với queue (hàng đợi) để mô phỏng lại ván cờ của người chơi giúp người chơi có thể xem lại quá trình chơi cờ.

Giao diện dễ nhìn, phù hợp với mọi lứa tuổi.

3.3 Nhược điểm

Bên cạnh những ưu điểm trên thì đồ án vẫn còn những hạn chế, khuyết điểm, như hạn chế thời gian, kiến thức còn hạn hẹp, đồ án chắc hẳn vẫn có một số bug mà chúng em chưa tìm ra.

Vẫn chưa kết nối được mọi người với nhau như qua LAN, internet…

3.4 Đề xuất phương án cải thiện

TÀI LIỆU THAM KHẢO

Nguồn 1 -https://vi.wikipedia.org/wiki/Ng%C4%83n\_x%E1%BA%BFp

Nguồn 2 - https://www.howkteam.vn/course/lap-trinh-game-caro-voi-c-winform-14