MỤC LỤC

[LỜI CẢM ƠN 3](#_Toc529130540)

[CÁC HÌNH VỀ GAME CARO 1 PLAYER 4](#_Toc529130541)

[1. TỔNG QUAN VỀ ĐỒ ÁN. 6](#_Toc529130543)

[*1.1.* Giới thiệu tổng quan về game Caro 1 player*.* 6](#_Toc529130544)

[*1.1.1. Lịch sử hình thành. 6*](#_Toc529130545)

[*1.1.2. Cách giải trong thực tế. 6*](#_Toc529130546)

[*1.1.3. Cách giải khác với cách nhìn của khoa học máy tính. 6*](#_Toc529130547)

[*1.1.4. Ứng dụng . 6*](#_Toc529130548)

[*1.2.* Về đồ án Mô phỏng thuật toán Game Caro*.* 6](#_Toc529130549)

[*1.2.1. Khái niệm về mô phỏng thuật toán. 6*](#_Toc529130550)

[*1.2.2. Tác dụng mô phỏng thuật toán. 6*](#_Toc529130551)

[*1.2.3. Kiến trúc một hệ thống mô phỏng thuật toán. 6*](#_Toc529130552)

[*1.2.4. Mô phỏng thuật toán GAME CARO 1 PLAYER. 6*](#_Toc529130553)

[*1.2.5. Lựa chọn ngôn ngữ cài đặt mô phỏng. 7*](#_Toc529130554)

[*1.3.* Lí do, mục đích chọn đồ án Demo game CARO 1 PLAYER. 7](#_Toc529130555)

[2. NỘI DUNG. 7](#_Toc529130556)

[2.1. Quá trình và công việc thực hiện đồ án. 10](#_Toc529130557)

[2.2. Khái quát thuật toán tháp Hà Nội bằng ngăn xếp. 7](#_Toc529130558)

[*2.2.1. Ngăn xếp (stack) là gì. 7*](#_Toc529130559)

[*2.2.2. Cấu trúc và đặc điểm của ngăn xếp. 7*](#_Toc529130560)

[*2.2.3. Ứng dụng của ngăn xếp. 8*](#_Toc529130561)

[*2.2.4. Thuật toán Tháp Hà Nội bằng ngăn xếp. 8*](#_Toc529130562)

[2.3. Thiết kế giao diện, backdrop. 10](#_Toc529130563)

[2.4. Cài đặt thuật toán và viết chương trình. 11](#_Toc529130564)

[2.5. Kiểm lỗi và debug. 11](#_Toc529130565)

[2.6. Chương trình sản phẩm của đồ án. 11](#_Toc529130566)

[2.7. Cách thức hoạt động của chương trình. 11](#_Toc529130567)

[3. Kết luận và rút kinh nghiệm đồ án. 11](#_Toc529130568)

[3.1. Bản kế hoạch hoàn thành. 11](#_Toc529130569)

[3.2. Ưu điểm. 11](#_Toc529130570)

[3.3. Nhược điểm. 11](#_Toc529130571)

[3.4. Phương án cải thiện. 11](#_Toc529130572)

[4. Phụ lục. 11](#_Toc529130573)

[5. Tài liệu tham khảo. 12](#_Toc529130574)

# 

# LỜI CẢM ƠN

Lời nói đầu tiên, nhóm em xin chân thành cảm ơn thầy cô và các bạn đã giúp đỡ và hoàn thành đồ án cuối kỳ này. Và lời cảm ơn đặc biệt nhất là GVHD thầy Trần Công Tú. Thầy đã hướng dẫn nhóm em rất tận tình trong suốt quá trình thực hiện đề tài.

Để hoàn thành đồ án “ GAME CARO 1 PLAYER có chức năng UNDO ”, nhóm em đã khảo sát, thu nhập dữ liệu và tổng hợp thông tin từ nhiều hướng khác nhau. Nhưng quá trình xây dựng chương trình, nghiên cứu và cố gắng hoàn thành, nhưng vì trình độ còn hạn chế nên có nhiều khó khăn làm chậm tiến độ. Mặt khác, đề tài tuy đã có sự giúp đỡ của GVHD nhưng vẫn không thể tránh khỏi những sai sót.

Do đó, nhóm em mong sẽ nhận được sự góp ý chân thành của thầy và các bạn đọc. Những ý kiến đóng góp về đề tài sẽ được nhóm em tiếp nhận để có thêm kinh nghiệm khi làm các đề tài nghiên cứu khác. Và một lần nữa, nhóm em xin cảm ơn Thầy Trần Công Tú và các bạn.

CÁC HÌNH VỀ GAME CARO 1 PLAYER

**ẻtyu**

sdftgy

# NỘI DUNG CHI TIẾT

1. TỔNG QUAN ĐỒ ÁN
   1. Giới thiệu tổng quan về trò chơi Caro một người .
      1. *Lịch sử hình thành.*

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 tuỳ 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 (chuỗi ngọc trai).

Và đến nay tại Việt Nam, trò chơi đã gắn liền với tuổi học trò của mỗi người. Chỉ với một tờ giấy và hai cây bút, luật chơi đơn giản, lũ học sinh sinh viên có thể mê say hàng giờ liền không biết chán. Cờ caro không chỉ là một trò chơi có tính giải trí cao mà còn đầy tính trí tuệ và hấp dẫn.

***Tài liêu tham khảo : gamevh.net***

* + 1. *Cách chơi.*

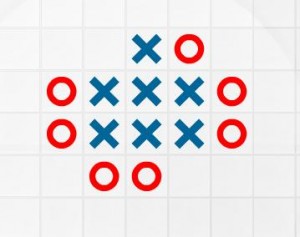
*-* Bàn cờ :

*+* 15x15. ( Loại cổ điển )

*+* 3x3 ( loại bàn cờ 3 nước liền nhau sẽ thắng )

+ Thực tế Bàn cờ tùy vào người chơi quyết định. ( phải trên 25 ô cờ (5x5) )

- Quân cờ : Cơ bản là “ X ” và “ O ”.

- Quân cờ X hoặc O đi trước.

* + 1. *Luật chơi.*

Mỗi người được đi 1 lần duy nhất và phải chờ đối thủ đánh rồi mới được đánh tiếp cờ thứ 2.

Ai tạo được nước nằm trên (1 hàng/cột/xiêng bên trái/xiêng bên phải ) 5 quân liền nhau thì thắng. Nếu nước trên 5 quân liều nhau hoặc bị chặn cả 2 đầu thì coi như chưa phân thắng bại.

* 1. Về đồ án và thuật toán Game Caro 1 player.
     1. *Khái niệm về mô phỏng thuật toán.*
     2. *Tác dụng mô phỏng thuật toán.*
     3. *Kiến trúc một hệ thống mô phỏng thuật toán.*
     4. *Mô phỏng thuật toán Game Caro 1 player.*
     5. *Lựa chọn ngôn ngữ chương trình.*

Nhóm chúng em đã 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 đặt trò chơi này.

* 1. Lí do, mục đích chọn đồ án Game Caro 1 player with UNDO
* Dựa vào những kiến thức đã học cũng như sở thích là tạo một chương trình game, mang lại nhu cầu giải trí sau những công việc nặng nhọc, học tập, …
* Từng là một người chơi nhiều các loại game khác nhau nhưng chưa có thể tạo được một trò chơi điện tử cho riêng mình thiết kế.
* Mục đích chính khi nhóm em chọn đề tài này vì : Muốn tự tay mình làm ra một trò chơi điện tử, từ game nhỏ sẽ thành lớn, từ những thể loại game cơ bản và đơn giản đến phức tạp.
* Với chức năng UNDO, điều mà từng đã thắc mắc khi sử dụng các phần mềm có công cụ đó. Tại sao có thể quay lại bước trước đó ? dù có lỡ tay xóa đi phần quan trọng nào đó, với công cụ này ta có thể quay lại bước trước đó. Từ thắc mắc đến tìm tòi và học hỏi, do vậy nên nhóm em chọn đề tài có chức năng UNDO, để có thể hiểu rõ hơn, vì sao nó làm được như vậy .
* Từ những đặc điểm trên nên nhóm em quyết định chọn game caro 1 người chơi có công cụ UNDO.

1. NỘI DUNG.
   1. Khái quát thuật toán UNDO bằng ngăn xếp.
      1. *Ngăn xếp (stack) là gì.*

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

* + 1. *Cấu trúc và đặc điểm của ngăn xếp.*

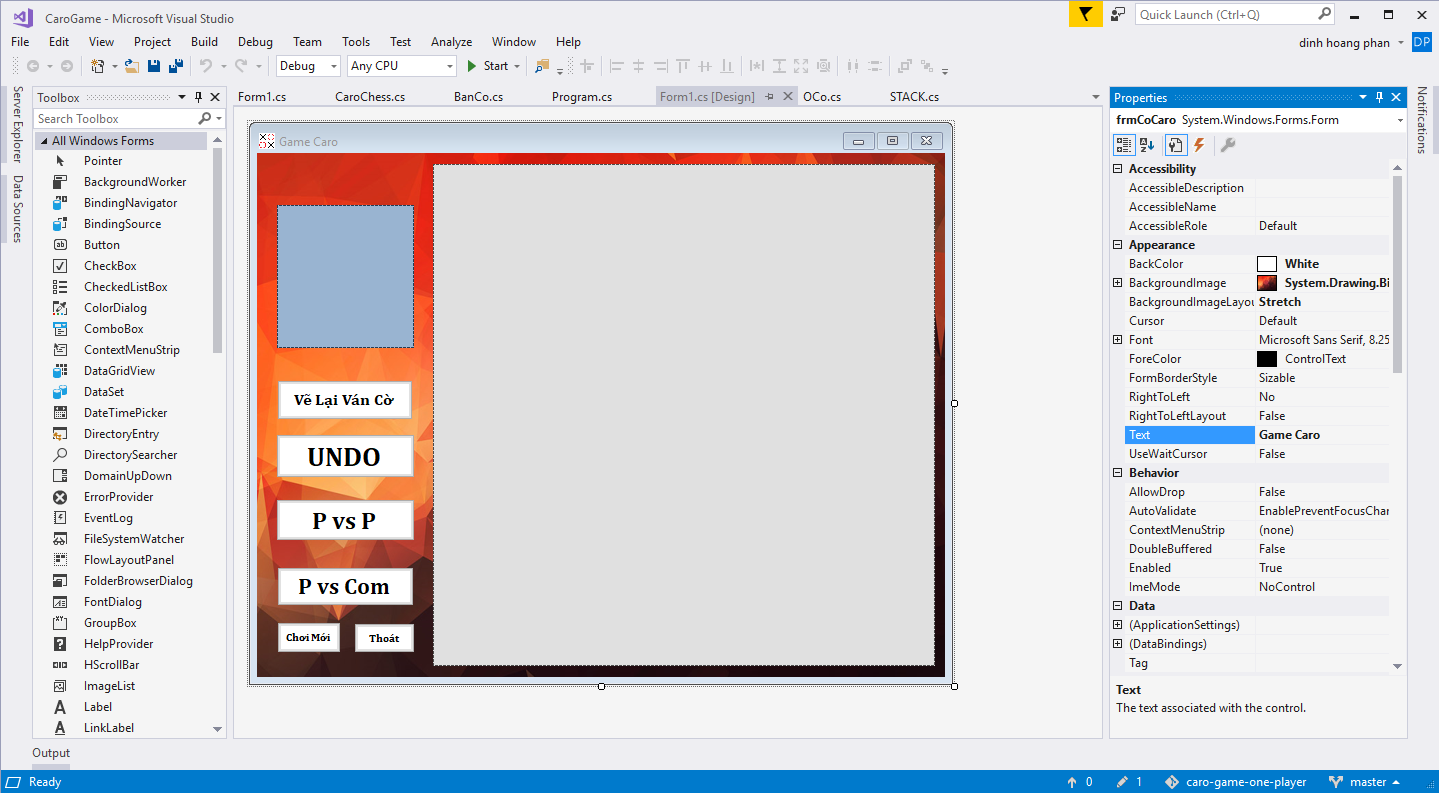
Một ngăn xếp là một cấu trúc dữ liệu dạng thùng chứa (container) của các phần tử (thường gọi là các nút (node)) và có hai phép toán cơ bản: push and pop. Push bổ sung một phần tử vào đỉnh (top) của ngăn xếp, nghĩa là sau các phần tử đã có trong ngăn xếp. Pop giải phóng và trả về phần tử đang đứng ở đỉnh 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. Ngoài ra, stack cũng hỗ trợ một số thao tác khác:

* isEmpty(): Kiểm tra xem stack có rỗng không.
* Top() (Hay peek()): Trả về giá trị của phần tử nằm ở đầu stack mà không hủy nó khỏi stack. Nếu stack rỗng thì lỗi sẽ xảy ra.
  + 1. *Một số ứng dụng của ngăn xếp.*
* Đảo ngược xâu ký tự : hiển thị các ký tự của 1 xâu ký tự theo chiều ngược lại.
* Tính giá trị biểu thức dạng hậu tố.
* Chức năng UNDO : trả lại bước trước đó.
  + 1. *Thuật toán Game Caro 1 player với chức năng UNDO bằng ngăn xếp*.

*Hoàng*

* 1. Quá trình và phân công công việc thực hiện đồ án.

|  |  |  |  |
| --- | --- | --- | --- |
| Công việc thực hiện | Phân công | Ngày dự kiến | Ngày thực tế |
| Tạo giao diện trò chơi, thiết kế hình ảnh, fonts chữ, bố cục,… | Hoàng + Đạt |  |  |
| Vẽ khung bàn cờ, quân cờ, thiết kế quân cờ, màu nền, phân chia tỉ lệ | Hoàng + Đạt |  |  |
| Xây dựng cách đánh quân cờ trên bàn cờ |  |  |  |
| Thiết lập cách đánh quân cờ sao cho : 1 quân cờ này rồi tiếp quân cờ kia ( cờ caro 2 người chơi ) |  |  |  |
| Thiết lập luật chơi cho game caro : khi nào thắng – thua, trường hợp không được tính. Trường hợp nào hợp lệ |  |  |  |
| Thiết lập AI ( trí tuệ nhân tạo ) cho chương trình game. Khả năng đánh cờ trên bàn cờ, tạo nước đi hợp lý và có thể dành chiến thắng | Hoàng |  |  |
| Thiết lập chức năng UNDO cho chương trình game ( sử dụng Stack để lưu lại vị trí ) | Hoàng |  |  |
| Kiểm tra sai xót – các trường hợp làm ảnh hưởng đến luật chơi cần phải sửa chửa lại. không thể làm người chơi cảm thấy Game sai luật. kiểm tra chức năng UNDO. | Hoàng + Đạt |  |  |
| Thiết kế lại giao diện phù hợp, thêm một số âm thanh nền ( nếu có ) | Đạt |  |  |

* 1. Thiết kế giao diện, backdrop.  
     

Cài đặt thuật toán và viết chương trình.  
Chương trình được viết bằng ngôn ngữ C#, được chia thành nhiều Class được phân cấp nhỏ dần để dễ quản lý một cách dễ dàng và có tính logic.

Lớp cao nhất là lớp CaroChess, trong lớp này chứa tất cả các hàm chức năng khởi tạo ván cờ, người chơi, xử lý người đánh cờ, kiểm tra thắng, vưu vết bước đi, undo, tạo chức năng đánh cho máy,…

Những lớp nhỏ hơn là lớp BanCo, trong lớp này chứa các thuộc tính cơ bản của một bàn cờ đó là số dòng và số cột, các hàm vẽ bàn cờ, vẽ quân cờ và xóa quân cờ dùng trong chức năng undo. Và lớp nhỏ nhất là lớp Ô Cờ, trong lớp này chứa thông tin về chiều dài và chiều rộng của 1 ô chứa quân cờ.

Các hàm xử lý chính:

* Hàm đánh cờ

public bool DanhCo(int MouseX, int MouseY, Graphics g, PictureBox p)

{

//không cho người chơi đánh ngay chính đường biên

if (MouseX % OCo.ChieuRong == 0 || MouseY % OCo.ChieuCao == 0) return false;

int Cot = MouseX / OCo.ChieuRong;

int Dong = MouseY / OCo.ChieuCao;

if (MangOco[Dong, Cot].SoHuu != 0)

return false;

//đổi lượt đánh

switch (LuotDi)

{

case 1:

{

MangOco[Dong, Cot].SoHuu = 1;

BanCo.VeQuanCo(g, MangOco[Dong, Cot].ViTri, MarkX);

LuotDi = 2;

if(CheDoChoi == 2) p.Image = Image.FromFile("picO.png");

else p.Image = Image.FromFile("picO.png");

break;

}

case 2:

{

MangOco[Dong, Cot].SoHuu = 2;

BanCo.VeQuanCo(g, MangOco[Dong, Cot].ViTri, MarkO);

LuotDi = 1;

if (CheDoChoi == 2) p.Image = Image.FromFile("picO.png");

else p.Image = Image.FromFile("picX.png");

break;

}

default:

MessageBox.Show("Có lỗi!");

break;

}

//Lưu vết bước đi

Stk\_CacNuocDaDi.Push(MangOco[Dong, Cot]);

return true;

}

//Áp dụng hàm này vào lớp Form để tạo khả năng đánh khi click chuột vào ô cờ

private void pnlBanCo\_MouseClick(object sender, MouseEventArgs e)

{

if (!caroChess.SanSang) return;

if(caroChess.DanhCo(e.X, e.Y,grs,p))

{

if (caroChess.KiemTraThang())

caroChess.KetThucTroChoi();

else

{

if (caroChess.CheDoChoi == 2)

{

caroChess.KhoiDongCom(grs);

if (caroChess.KiemTraThang())

caroChess.KetThucTroChoi();

}

}

}

}

//Hàm xử lý Undo

public void Undo(Graphics g)

{

if (CheDoChoi == 1)

{

if (Stk\_CacNuocDaDi.Count() != 0)

{

OCo oco = Stk\_CacNuocDaDi.Pop();

MangOco[oco.Dong, oco.Cot].SoHuu = 0;

BanCo.XoaQuanCo(g, oco.ViTri, sbScreen);

}

}

else

{

if (Stk\_CacNuocDaDi.Count() >= 1)

{

OCo oco = Stk\_CacNuocDaDi.Pop();

MangOco[oco.Dong, oco.Cot].SoHuu = 0;

BanCo.XoaQuanCo(g, oco.ViTri, sbScreen);

OCo oco2 = Stk\_CacNuocDaDi.Pop();

MangOco[oco2.Dong, oco2.Cot].SoHuu = 0;

BanCo.XoaQuanCo(g, oco2.ViTri, sbScreen);

}

}

}

//Hàm undo trong lớp Form

private void btnUndo\_Click(object sender, EventArgs e)

{

caroChess.Undo(grs);

if(caroChess.CheDoChoi == 1 && caroChess.Stk\_CacNuocDaDi.Count != 0)

{

if (caroChess.LuotDi == 1)

{

caroChess.LuotDi = 2;

pctbMark.Image = Image.FromFile("picO.png");

}

else

{

caroChess.LuotDi = 1;

pctbMark.Image = Image.FromFile("picX.png");

}

}

}

* 1. Kiểm lỗi và debug.
  2. Chương trình sản phẩm của đồ án.
  3. Cách thức hoạt động của chương trình.

1. Kết luận và rút kinh nghiệm đồ án.
   1. Bản kế hoạch hoàn thành.
   2. Ưu điểm.
   3. Nhược điểm.
   4. Phương án cải thiện.
2. Tài liệu tham khảo.
   1. *Myclass.vn.*