

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

..... □  □



BÁO CÁO ĐỒ ÁN 1

Đề tài:

XÂY DỰNG GAME CARO

GVHD: Từ Tuyết Hồng

SVTH: Nguyễn Tấn Quốc Khánh

17110161

TP. Hồ Chí Minh, ngày 20, tháng 5, năm 2022

MỤC LỤC

NỘI DUNG	4
I. Giới thiệu đề tài	4
1. <i>Lý do chọn đề tài</i>	<i>4</i>
2. <i>Mục tiêu đề tài</i>	<i>4</i>
3. <i>Cấu trúc dữ liệu</i>	<i>4</i>
II. Cài đặt và thực thi	5
1. <i>Thiết kế giao diện.....</i>	<i>5</i>
2. <i>Cấu trúc dữ liệu</i>	<i>8</i>
3. <i>Cài đặt và kiểm thử.....</i>	<i>12</i>
III. Phân công công việc.....	16
IV. Kết luận	16
TÀI LIỆU THAM KHẢO	17

DANH MỤC CÁC HÌNH

Hình 1 . Giao diện chính của gam caro	5
Hình 2 . MessageBox thông báo quân X đi trước	13
Hình 3 . Giao diện khi chọn chế độ "Player vs Com"	14
Hình 4 . Giao diện khi đánh cờ	15
Hình 5 . Giao diện sau khi Undo	15

NỘI DUNG

I. Giới thiệu đề tài

1. Lý do chọn đề tài

- Cờ ca-ro là game rất phổ biến, đặc biệt trong giới thanh thiếu niên. Quay lại thời học sinh, chắc chắn ai ai cũng đã từng có những trận đấu caro đầy thú vị diễn vào giờ ra chơi, cuối mỗi buổi học hay thậm chí ngay trong tiết học. Để chiến thắng, bạn cần tạo ra những nước cờ hiểm và độc. Điểm đặc biệt của Caro, là bạn rất dễ bị thua nếu không để ý từng nước đi của đối phương mặc dù bạn có chiến thuật tốt hơn
- Luật chơi Caro khá đơn giản, nhưng không kém phần hấp dẫn giúp người chơi giải trí, thư giãn sau những giờ làm việc và học tập căng thẳng . Xuất phát từ nhu cầu đó, nhóm em đã xây dựng một phần mềm chơi game Caro.
- Với giao diện thân thiện, dễ nhìn giúp người chơi giải trí, ngoài ra phần mềm còn có chức năng đánh lại (undo) cho phép người chơi xóa nước cờ đã đi khi đánh nhầm.

2. Mục tiêu đề tài

- Ứng dụng game caro được xây dựng nhằm mục đích làm quen với ngôn ngữ, các câu lệnh cơ bản và một số công nghệ hướng đối tượng. Xây dựng chức năng Undo trong game caro.
- Ứng dụng còn nhằm mục đích giải trí, giảm căng thẳng cho người chơi.

3. Cấu trúc dữ liệu

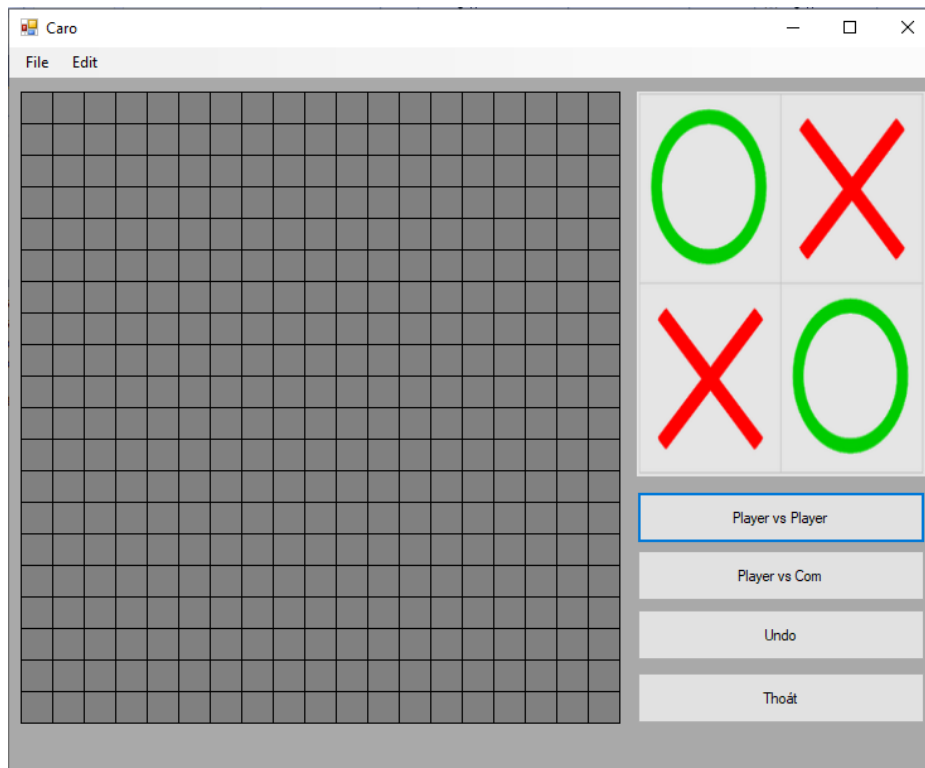
- Sử dụng ngăn xếp (stack) và các hàm của ngăn xếp để thiết kế chức năng undo. Ngoài ra, kết hợp với các hàm xử lý thắng thua, tổ chức máy AI,... để cài đặt phần mềm. Chương trình sử dụng giao diện Winform dễ nhìn, các thao tác đánh cờ, đổi chế độ chơi, undo chỉ cần click chuột.

- Mô tả giải thuật lưu tất cả các nước đi của hai người chơi vào ngăn xếp, các nước đi theo thứ tự chồng lên nhau, nước đi cuối cùng nằm trên cùng ngăn xếp. Khi chạy chương trình và bắt đầu chơi, dữ liệu các nước đi sẽ được lưu vào một stack, khi nào cần thì lấy ra theo nguyên lý LIFO (Last In First Out), chương trình sẽ xóa nước đi đó trên màn hình và gán lại lượt đi và cho người chơi đánh lại.

II. Cài đặt và thực thi

1. Thiết kế giao diện

- **Form1** là giao diện chính của trò chơi dùng để đánh cờ, chuyển đổi giữa các chế độ chơi, undo và thoát khỏi trò chơi.



Hình 1. Giao diện chính của game caro

- Thiết kế **Form1** cần thực hiện những bước sau:
 - Tạo các thành phần phục vụ cho từng chức năng:

Thành phần	Name	Text	Mô tả
Panel	pnlBanCo		Dùng để đánh cờ và hiển thị diễn biến trận đánh.
Button	btnPlayervsPlayer	Player vs Player	Khi click vào button này, chương trình sẽ chuyển sang chế độ người đánh với người.
	btnPlayervsCom	Player vs Com	Khi click vào button này, chương trình sẽ chuyển sang chế độ người đánh với máy.
	btnUndo	Undo	Khi click vào button này, chương trình sẽ xóa nước đi vừa mới đánh khỏi bàn cờ và cho phép người chơi đi lại.
	btnThoat	Thoát	Khi click vào button này, chương trình sẽ hiện MessageBox “Bạn có muốn thoát khỏi game”, click OK để thoát ra, click Cancel để ở lại.
PictureBox	pictureBox1		Hiện hình ảnh minh họa cho trò chơi.

- Tạo một menu gồm các mục có chức năng riêng:

Thành phần	Name	Text	Mô tả
MenuStrip	fileToolStripMenuItem	&File	Khi click vào sẽ hiện ra một menu gồm mục new và mục exit.
	newToolStripMenuItem	&New	Khi click vào sẽ hiện ra một menu gồm mục Player vs Player và mục Player vs Com.
	playerVsPlayerToolStripMenuItem	&Player vs Player	Khi click vào mục này, chương trình sẽ chuyển sang chế độ người đánh với người.
	playerVsComToolStripMenuItem	&Player vs Com	Khi click vào mục này, chương trình sẽ chuyển sang chế độ người đánh với máy.
	exitToolStripMenuItem	&Exit	Khi click vào mục này, chương trình sẽ hiện MessageBox “Bạn có muốn thoát khỏi game”, click OK để thoát ra, click Cancel để ở lại.
	editToolStripMenuItem	&Edit	Khi click vào sẽ hiện ra mục undo
	undoToolStripMenuItem	&Undo	Khi click vào mục này, chương trình sẽ xóa nước đi vừa mới đánh khỏi bàn cờ và cho phép người chơi đi lại.

2. Cấu trúc dữ liệu

2.1 Cấu trúc dữ liệu chính

2.1.1 Tổng quan về ngăn xếp (stack)

- Stack (ngăn xếp) là một cấu trúc dữ liệu trừu tượng chứa các đối tượng làm việc theo cơ chế LIFO (Last In First Out), tức việc thêm 1 đối tượng vào Stack hoặc lấy 1 đối tượng ra khỏi Stack được thực hiện theo cơ chế “vào trước ra sau”.

2.2 Các thao tác cơ bản trong ngăn xếp

- Các hoạt động cơ bản trên ngăn xếp có thể liên quan tới việc khởi tạo ngăn xếp, sử dụng nó và sau đó xóa nó. Ngoài các hoạt động cơ bản này, một ngăn xếp có hai hoạt động nguyên sơ liên quan tới khái niệm:

Tên phương thức	Ý nghĩa
Push()	Thêm một phần tử vào vị trí trên cùng của ngăn xếp.
Pop()	Xóa phần tử được thêm vào cuối cùng trong ngăn xếp.

2.3 Ứng dụng ngăn xếp (stack) vào chức năng undo

- Khi khởi động chương trình và bắt đầu trận đấu, dữ liệu của các nước đi sẽ được lưu vào ngăn xếp theo cơ chế LIFO để thuận tiện cho việc lấy ra. Do đó, nhóm em khai báo ngăn xếp như sau:

- private Stack<Oco> stack_CacNuocDaDi;

- Các nước đi có kiểu dữ liệu là Oco. Trong class Oco gồm 4 thuộc tính:

- class Oco{

- public int Dong;

- public int Cot;


```

public Point Vitri;

public int SoHuu;    }

```

Thuộc tính	Thể hiện
dong	Dòng của ô cờ.
cot	Cột của ô cờ.
vitri	Tọa độ thực tế khi click chuột vào để đánh.
sohuu	Ô cờ thuộc sở hữu của người chơi nào.

- Ở trong class Oco, khai báo thêm một thuộc tính là vitri với kiểu Point. Mục đích là để khi click chuột vào ô cờ để đánh, nếu click không đúng vào ô hoặc click vào dòng, cột của bàn cờ thì chương trình sẽ không cho phép đánh.
 - `public bool DanhCo(int MouseX, int MouseY,Graphics g)`

```

{

    if (MouseX % Oco._ChieuRong == 0 || MouseY %
        Oco._ChieuCao == 0)

        return false;

}

```
- Khi đánh cờ thì các nước mà hai bên đã đi sẽ được thêm vào ngăn xếp bằng hoạt động Push.
 - `stack_CacNuocDaDi.Push(oco);`

- Sử dụng giải thuật đếm để kiểm tra số nước đi và xóa phần tử nước đi trong ngăn xếp. Nếu ngăn xếp rỗng thì ngưng hoạt động xóa.

```
if (stack_CacNuocDaDi.Count != 0)
```

```
Oco oco = stack_CacNuocDaDi.Pop();
```

- Do game caro sẽ có hai chế độ chơi. Nếu ở chế độ PlayerVsCom thì Undo sẽ sử dụng hoạt động Pop hai lần. Tương tự với chế độ PlayervsPlayer, ta chỉ dùng Pop một lần.

1 reference

```
private void btnUndo_Click(object sender, EventArgs e)
{
    dieukhienCaro.Undo(grs);
    if (dieukhienCaro.CheDoChoi == 2)
        dieukhienCaro.Undo(grs);
}
```

- Khi sử dụng hoạt động pop thì nó chỉ xóa phần tử là nước đi trong ngăn xếp chứ còn trên bàn cờ vẫn hiện lên nước đi đó. Chính vì vậy mà ta cần phải lấp đầy ô cờ bằng màu của bàn cờ.

1 reference

```
public void Delete(Graphics g, Point point)
{
    SolidBrush sb = new SolidBrush(Color.Gray);
    g.FillRectangle(sb, point.X, point.Y, Oco._ChieuRong, Oco._ChieuCao);
}
```

- Để thực hiện Undo nhanh hơn ta vẽ lại bàn cờ và phục hồi lại các nước đi còn lại trong ngăn xếp, đồng thời gán lại lượt đi cho người chơi.
- Trong thao tác khôi phục quân cờ, Lúc này phần tử nước đi trên cùng đã bị xóa, ta chỉ cần kiểm tra lại số lượng phần tử trong ngăn xếp và cho chạy vòng lặp đánh lại các quân cờ còn lại trong ngăn xếp.

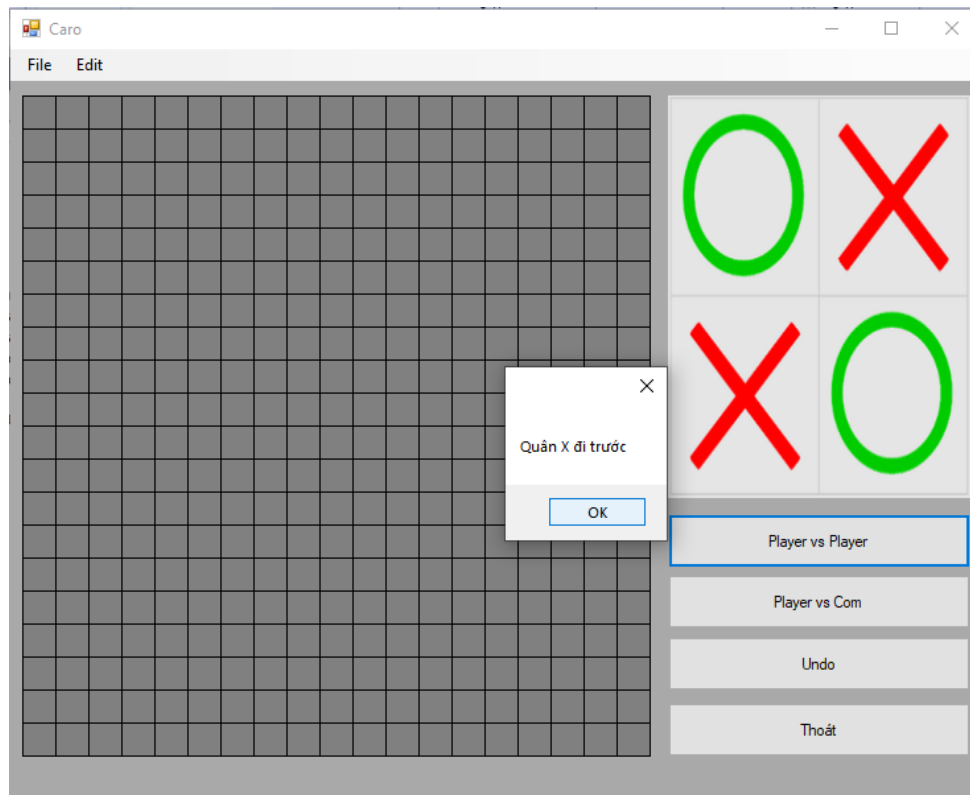
- Ở mỗi hướng duyệt, ta đều phải kiểm tra hai đầu để xem có bị chặn hai đầu không. Ví dụ với cách duyệt dọc, phải duyệt từ trên xuống dưới và từ dưới lên trên:
 - `for (int dem = 1; dem < 6 && curDong + dem < _BanCo.SoDong; dem++){ }`
 - `for (int dem = 1; dem < 6 && curDong - dem >= 0; dem++){ }`
- Nếu bắt gặp phải quân địch thì sẽ trừ điểm dựa trên số quân địch (tối đa quân địch là 2). Tương tự cũng sẽ cộng điểm dựa trên số quân ta:
 - `diemTong -= MangDiemPhongThu[soQuanDich];`
 - `diemTong += MangDiemTanCong[soQuanTa];`
- Ở phần điểm phòng ngự, cũng duyệt theo bốn hướng với cách duyệt tương tự trên, với cách tính điểm:
 - `diemTong += MangDiemPhongThu[soQuanDich];`
 - `diemTong -= MangDiemTanCong[soQuanTa] * 2;`
- Ta cũng tính điểm tổng dựa trên số quân địch và quân ta. Cứ mỗi lần số quân địch tăng lên thì ta sẽ trừ đi hai lần mảng điểm tấn công vì phải chặn hai đầu.

3. Cài đặt và kiểm thử

3.1 Các trường hợp kiểm thử

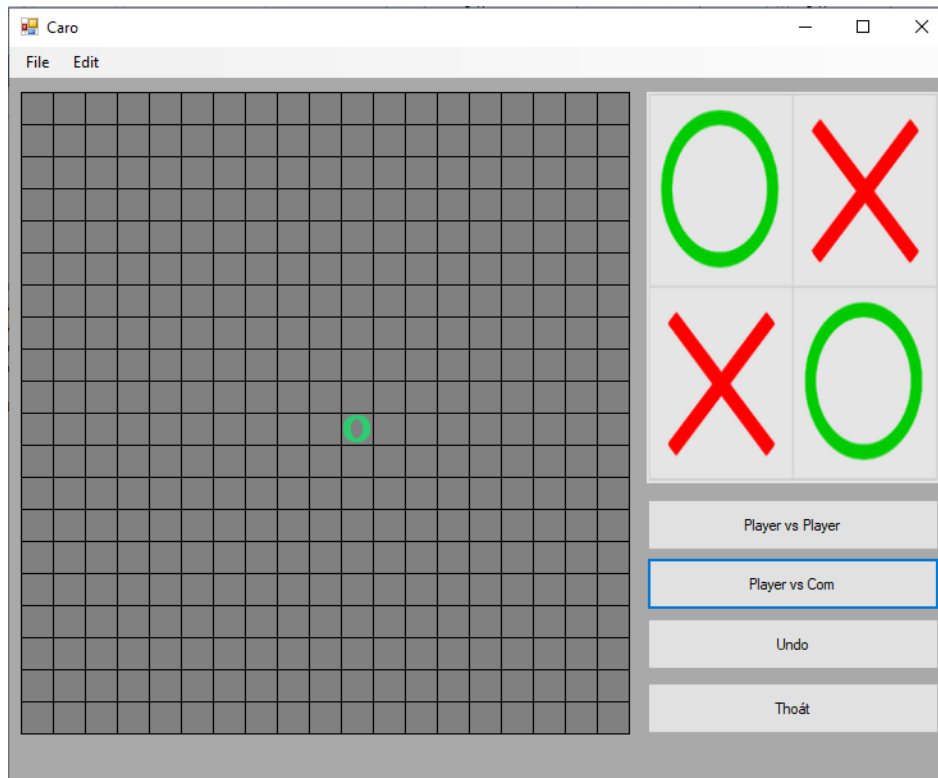
3.1.1 Chế độ chơi

- Khởi động chương trình game caro và click chuột vào **btnPlayervsPlayer** hoặc click chuột vào **fileToolStripMenuItem** → **newToolStripMenuItem** → **playerVsPlayerToolStripMenuItem**. Ngay lập tức, chương trình hiển thị một MessageBox thông báo ngẫu nhiên “Quân O đi trước” hoặc “Quân X đi trước”.



Hình 2. MessageBox thông báo quân X đi trước

- Trường hợp click chuột vào **btnPlayervsPlayer** hoặc click chuột vào **fileToolStrip**
- **MenuItem** → **newToolStripMenuItem** → **playerVsComToolStripMenuItem**
.Ngay lập tức, chương trình hiển thị mặc định quân O giữa bàn cờ do máy mới đánh.

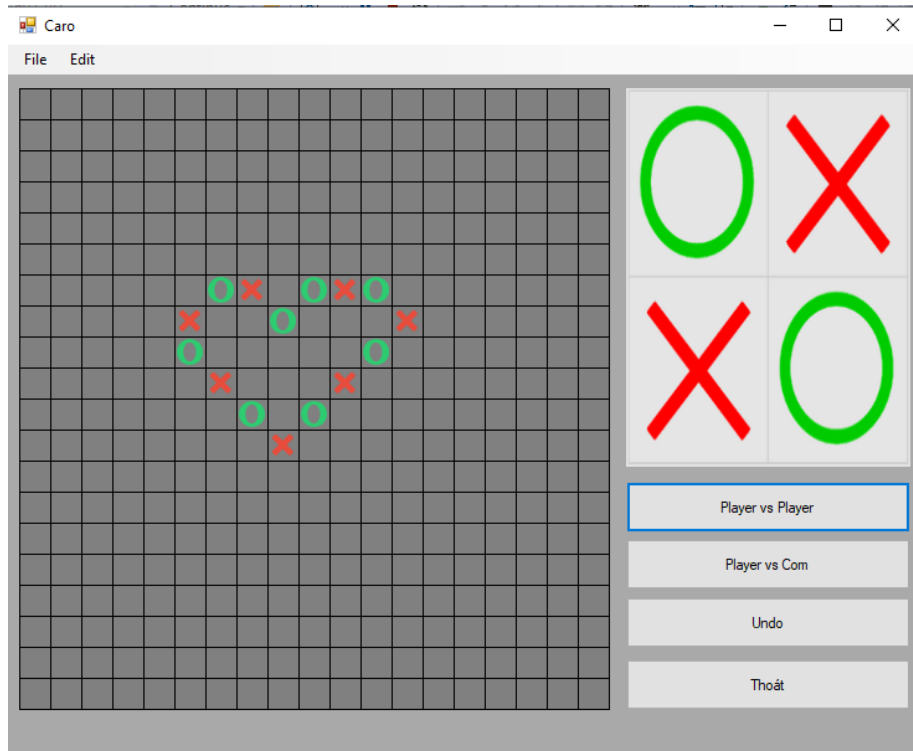


Hình 3. Giao diện khi chọn chế độ "Player vs Com"

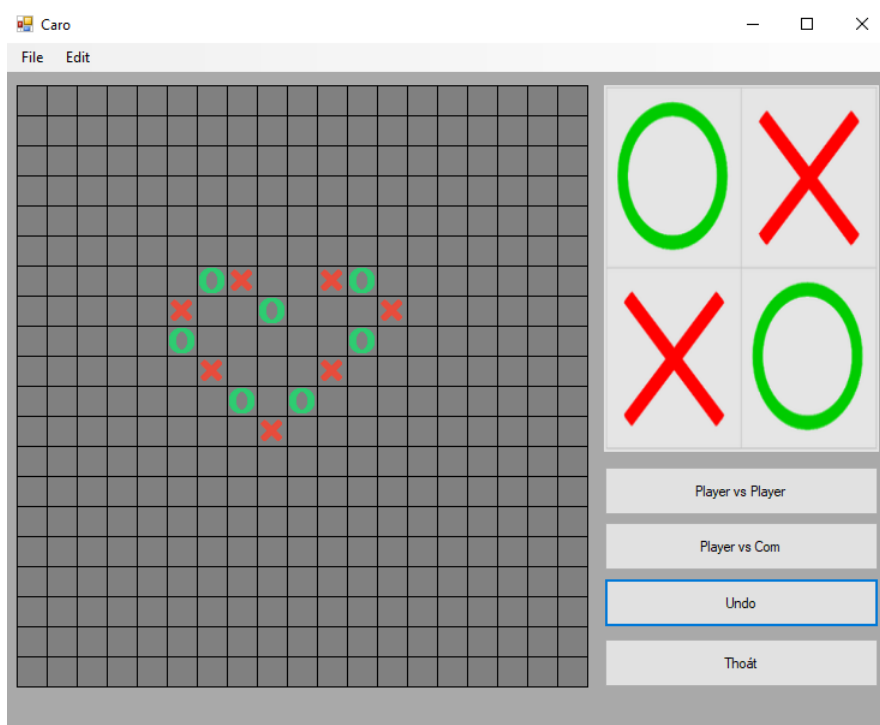
- Trường hợp không chọn chế độ chơi thì chương trình sẽ không cho phép người chơi đánh.

3.1.2 Chức năng Undo

- Đối thủ sắp chiến thắng và ta không kịp chặn. Lúc này, khi click vào **btnUndo** hoặc **fileToolStripMenuItem** → **editToolStripMenuItem** → **undoToolStripMenuItem**. Chương trình sẽ xóa nước ta vừa mới đi và cho phép ta đi lại.



Hình 4. Giao diện khi đánh cờ



Hình 5. Giao diện sau khi Undo

III. Phân công công việc

Danh sách sinh viên thực hiện

Mã số SV	Họ và tên	Tỷ lệ đóng góp
17110161	Nguyễn Tấn Quốc Khánh	100%

IV. Kết luận

4.1 Đánh giá mức độ hoàn thành

- Trong đồ án này, nhóm em đã tìm hiểu được các kiến thức về ngăn xếp (stack), cách khai báo cũng như các thao tác trên ngăn xếp như xóa một phần tử, đếm phần tử. Mục đích áp dụng vào game caro để thực hiện chức năng Undo. Có thể nói các thuật toán rất thích hợp với game caro này.
- Sản phẩm nhóm em đặt ra không có yêu 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. Tuy nhiên, sản phẩm cũng chỉ mô phỏng các thuật toán trong ngăn xếp. Để có thể đem sản phẩm này vào thực tế, nhóm em cần thêm rất nhiều thời gian và kinh nghiệm.

4.2 Các khó khăn

- Trong quá trình chạy đồ án này thì nhóm em gặp cũng khá nhiều khó khăn. Bởi em phải tự tìm hiểu tất các tài liệu về ngăn xếp, và viết báo cáo. Ngoài ra, các yếu tố thời gian và khối lượng công việc cũng gây không ít khó khăn cho nhóm em. Tuy nhiên, về sau cùng nhóm em cũng đã hoàn thành sản phẩm bằng tất cả khả năng và nỗ lực của mình.

4.3 Hướng phát triển

- Giao diện thân thiện, đơn giản, dễ sử dụng và thao tác. Tuy nhiên vẫn còn một số lỗi như bàn cờ còn hạn chế, chưa đẹp mắt. Tuy vậy, do những hạn chế về

trình độ, thời gian cũng như thiết bị nên chúng em không khỏi những vướng mắc, sai sót trong quá trình tìm hiểu và nghiên cứu. Chúng em rất mong nhận được những đánh giá và chỉnh sửa từ các thầy cô hướng dẫn.

- Trong thời gian tiếp theo, nhóm em sẽ tiếp tục cải thiện và phát triển một số tính năng khác để có thể game caro ra thực tiễn.
- Cụ thể, nhóm em sẽ tìm hiểu thêm về các thao tác khác của ngăn xếp. Từ đó, nhóm em sẽ thêm một tính năng khác cho game caro nữa là redo.
- Ngoài ra, nhóm em cũng sẽ phát triển thêm tính năng phát nhạc nền khi chơi game. Tạo sự hứng thú cho người chơi đồng thời giúp người chơi thư giãn đầu óc.
- Cuối cùng, nhóm em sẽ phát triển game caro theo hướng có thể chơi được từ hai máy khác nhau thông qua mạng LAN để giúp sản phẩm hoàn thiện hơn.

TÀI LIỆU THAM KHẢO

1. How Kteam. Stack trong C#,

<https://www.howkteam.vn/course/khoa-hoc-lap-trinh-c-nang-cao/stack-trong-c-1562>

2. Mai Chi, Ngăn xếp (stack) trong C#: cài đặt, ứng dụng, lớp Stack, _

<https://tuhocict.com/ngan-xep-stack-cau-truc-cai-dat-ung-dung-lop-stack-trong-c/>

3. Kame, [Hướng dẫn] System.Drawing- Vài phương thức căn bản,

<https://www.tapatalk.com/groups/noimforum/h-ng-d-n-system-drawing-vai-ph-ng-th-c-c- n-b-n-t1767.html>

4. Khoa khoa học & kỹ thuật Máy tính, trường ĐH Bách Khoa Tp.HCM, Tương tác với người dùng trong ứng dụng C#,

<http://www.cse.hcmut.edu.vn/~hiep/LaptrinhHDT/LyThuyet/Chuong7.pdf>

