

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỒ ÁN MÔN HỌC
KỸ THUẬT LẬP TRÌNH**

**ĐỒ ÁN HỌC KÌ II
NĂM HỌC: 2022-2023**

GAME CARO

NHÓM THỰC HIỆN: NHÓM 2

LỚP: 22CTT4

GIẢNG VIÊN HƯỚNG DẪN: TRƯƠNG TOÀN THỊNH

1. MỞ ĐẦU	2
Thông tin nhóm.....	2
Ý tưởng game.....	2
2. GIỚI THIỆU CÁC CHỨC NĂNG GAME	4
A. Kỹ thuật, thuật toán đã tìm hiểu và sử dụng.....	6
Kỹ thuật chính	6
Kỹ thuật phụ	7
B. Mô tả code và các ý tưởng của từng chức năng code.....	7
3. TÀI LIỆU THAM KHẢO	14

1. MỞ ĐẦU

Thông tin nhóm

MSSV	Họ tên	Email	Vai trò
22120236	Nguyễn Hoàng Nguyên	22120236@student.hcmus.edu.vn	Tạo quy luật thuật thắng thua, chế độ người đấu với người
22120210	Lê Võ Nhật Minh	22120210@student.hcmus.edu.vn	Thiết kế chế độ người đấu với máy (bao gồm chế độ dễ và khó)
22120196	Thông Lúc	22120196@student.hcmus.edu.vn	Thiết kế phần Save và Load Game của trò chơi
22120204	Nguyễn Đức Mạnh	22120204@student.hcmus.edu.vn	Thiết kế giao diện và chức năng menu
22120227	Nguyễn Hữu Nghĩa	22120227@student.hcmus.edu.vn	Thiết kế hiệu ứng trò chơi, chèn âm thanh trò chơi
22120236	Nguyễn Hoàng Nguyên	22120236@student.hcmus.edu.vn	Tạo quy luật thuật thắng thua, chế độ người đấu với người

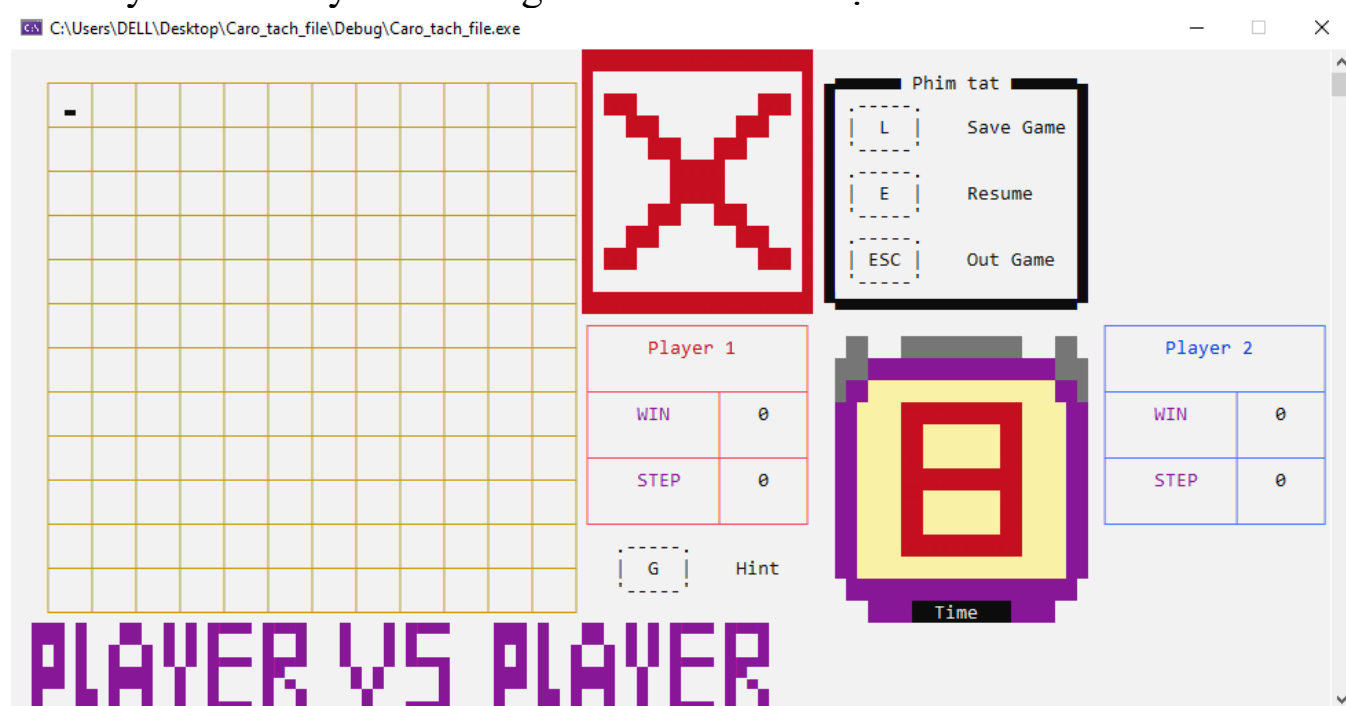
Ý tưởng Game

- Quy luật game: chỉ cần quân X hoặc O có 5 nước chéo, thắng liên tục thì quân đó chiến thắng (không áp dụng *luật chặn hai đầu*).
- Thiết kế game với chế độ người đấu với người và người đấu với máy (bao gồm chế độ dễ và khó). Người chơi có thể sử dụng chức năng hint để có thể nhận được gợi ý cho nước đi kế tiếp.
- Tạo chức năng lưu game và mở lại danh sách game đã lưu.
- Thêm chức năng thoát game khi người dùng muốn kết thúc. Chức năng tạm dừng game để người dùng có thể tạm dừng ván đấu.
- Tạo hiệu ứng chuyển nền khi bắt đầu vào trận Game mà người dùng có thể nhập tên theo ý muốn.
- Thiết kế thêm thời gian (Mỗi lượt chơi sẽ có 10s để suy nghĩ, sau 10s người chơi chưa đánh, người còn lại thắng).
- Thiết kế khả năng tự lưu game khi trước đó người chơi đã mở file game đó lên chơi và thoát ra mà không cần nhập tên lại nữa.
- Thiết kế hiệu ứng âm thanh đồ họa khi tìm ra người chiến thắng. Có thể bật/tắt nhạc ở màn hình chờ ngay đầu game.

-Phần Load Game thiết kế người dùng sử dụng phím mũi tên lên xuống để lựa chọn file cần mở. Người dùng còn có thể tìm kiếm bằng cách nhập từ bàn phím(Kết quả tìm kiếm sẽ được cập nhật mỗi khi người dùng nhập vào).

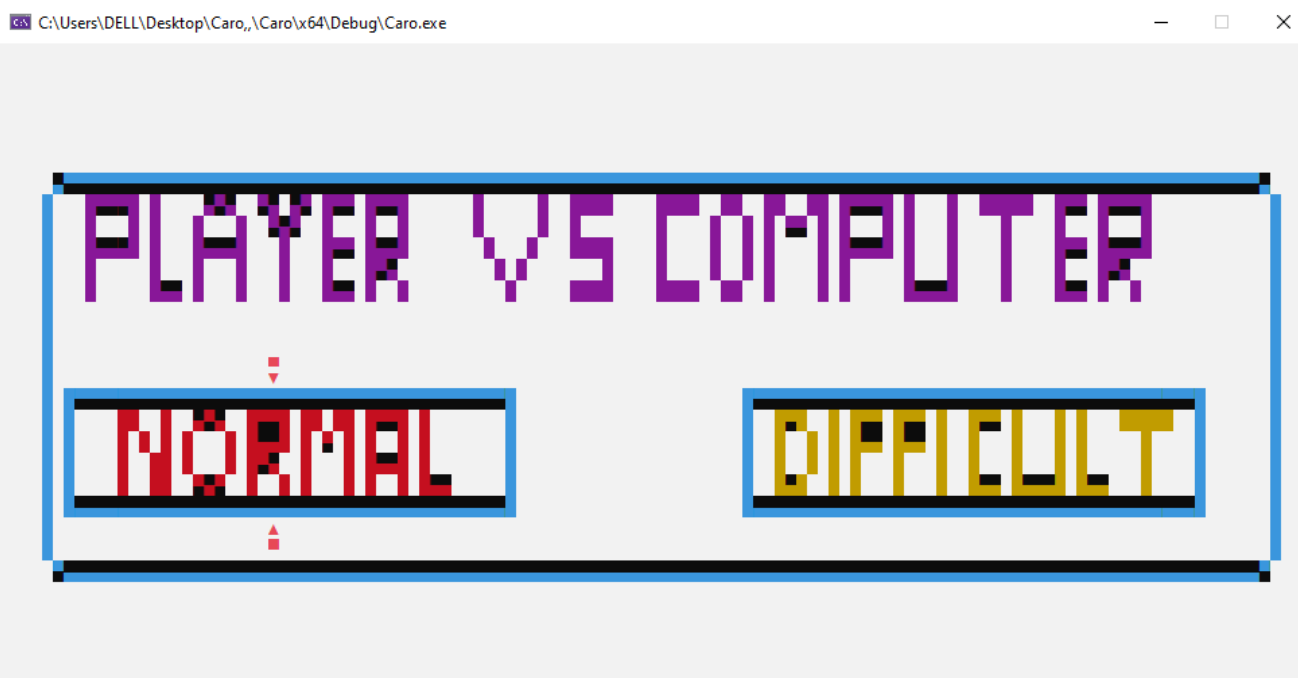
2. GIỚI THIỆU CÁC CHỨC NĂNG GAME

1. Player vs Player : 2 người chơi lần lượt là X và O đấu với nhau.

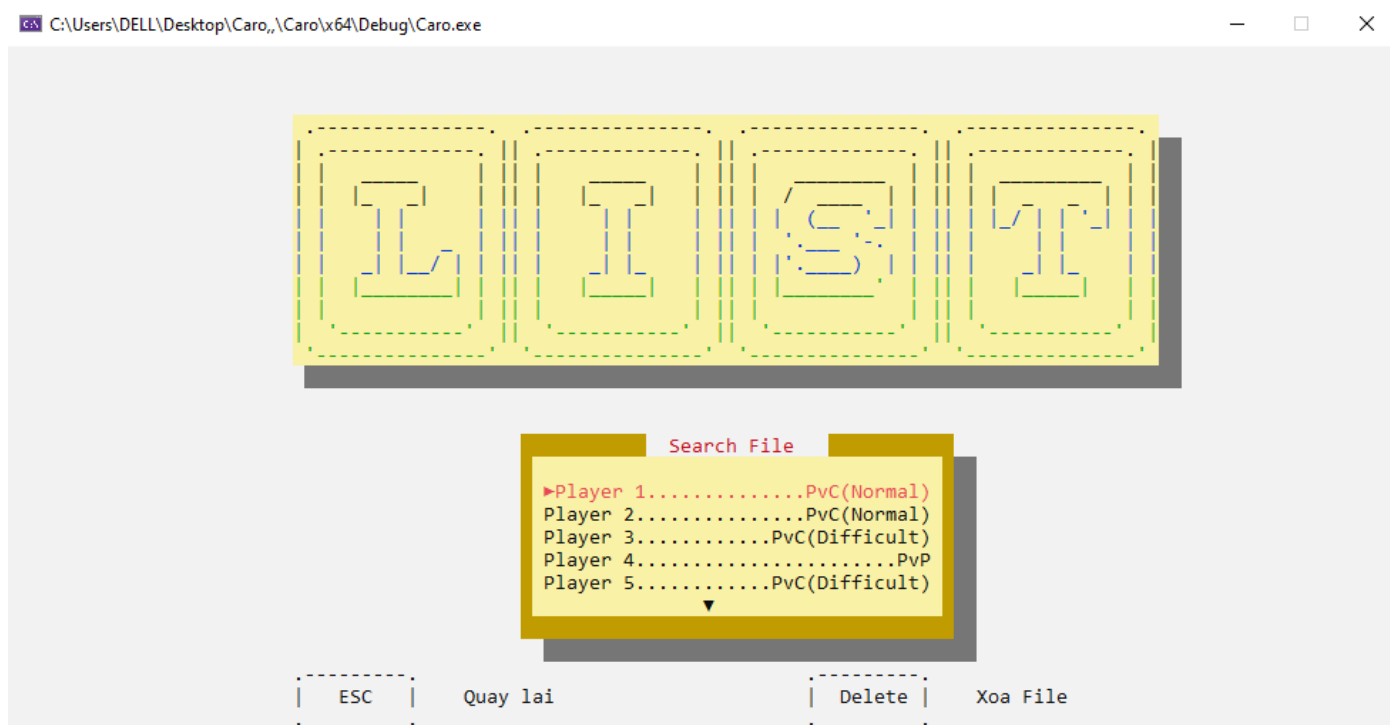


2. Player vs Bot (Normal) : Người chơi với máy chế độ bình thường .

3. Player vs Bot (Difficult) : Người chơi với máy chế độ khó.

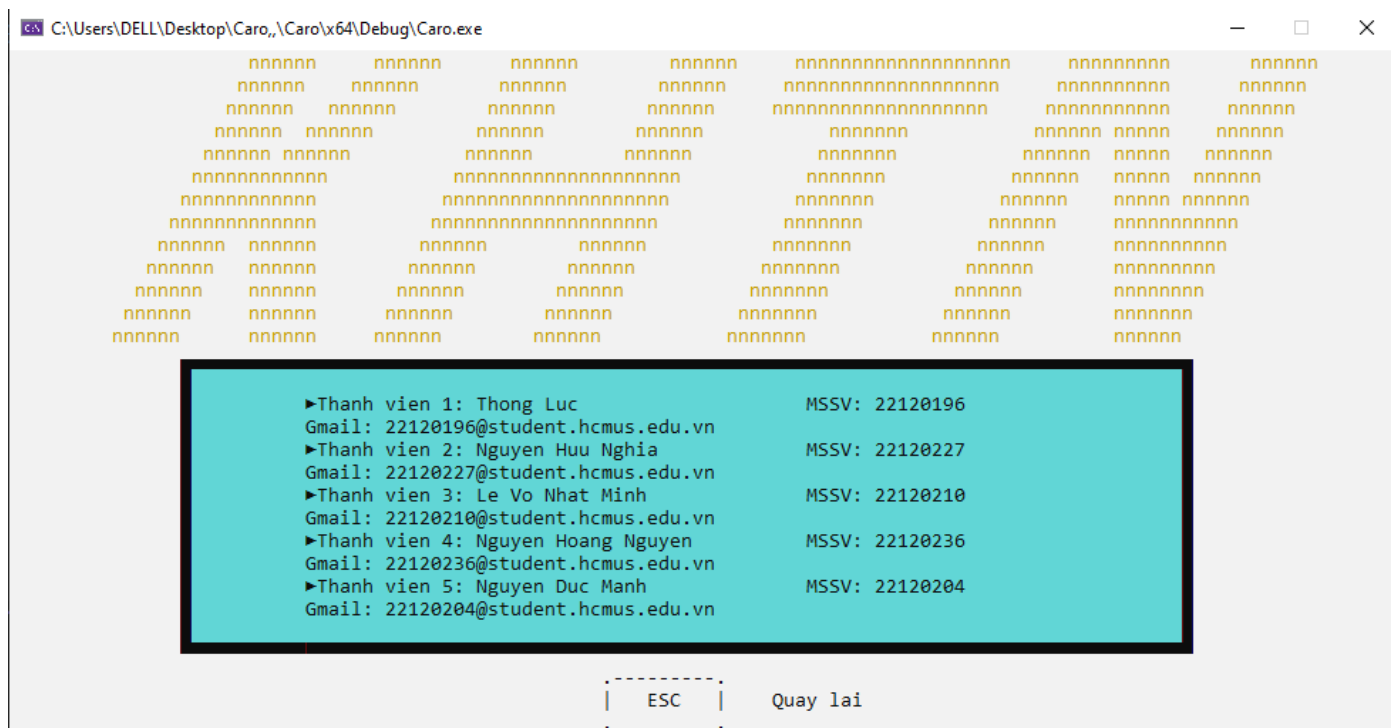


4. Load Game : Load lại trận đấu đã tạm dừng được lưu dưới dạng file “.txt”.



5. Save Game : Lưu game trận đấu đang chơi .

6. About : Thông tin về nhà phát triển.



7. Help: Hướng dẫn chơi game.



8. Exit : Thoát game (Khi thoát game sẽ xuất hiện dòng chữ “Thank You”).



A. Kỹ thuật, thuật toán đã tìm hiểu và sử dụng:

* **Kỹ thuật chính:**

- **Kiểm tra thắng thua:** (Không áp dụng luật chặn 2 đầu)
 - ✓ Xét nước đi theo chiều dọc
 - ✓ Xét nước đi theo chiều ngang
 - ✓ Xét nước đi theo đường chéo chính
 - ✓ Xét nước đi theo đường chéo phụ
- **Thuật toán:** Lấy ra từng ma trận 5x5 trong bàn cờ và kiểm tra trên 1 hàng, 1 cột, đường chéo chính, đường chéo phụ có thỏa mãn 5 nước X hoặc O thắng hàng không, nếu thỏa mãn thì trò chơi tìm ra người chiến thắng. Nếu đến khi ô cuối cùng chơi bàn cờ được đánh nhưng không tìm ra người thắng thì trận đấu hòa.

* Kỹ thuật phụ:

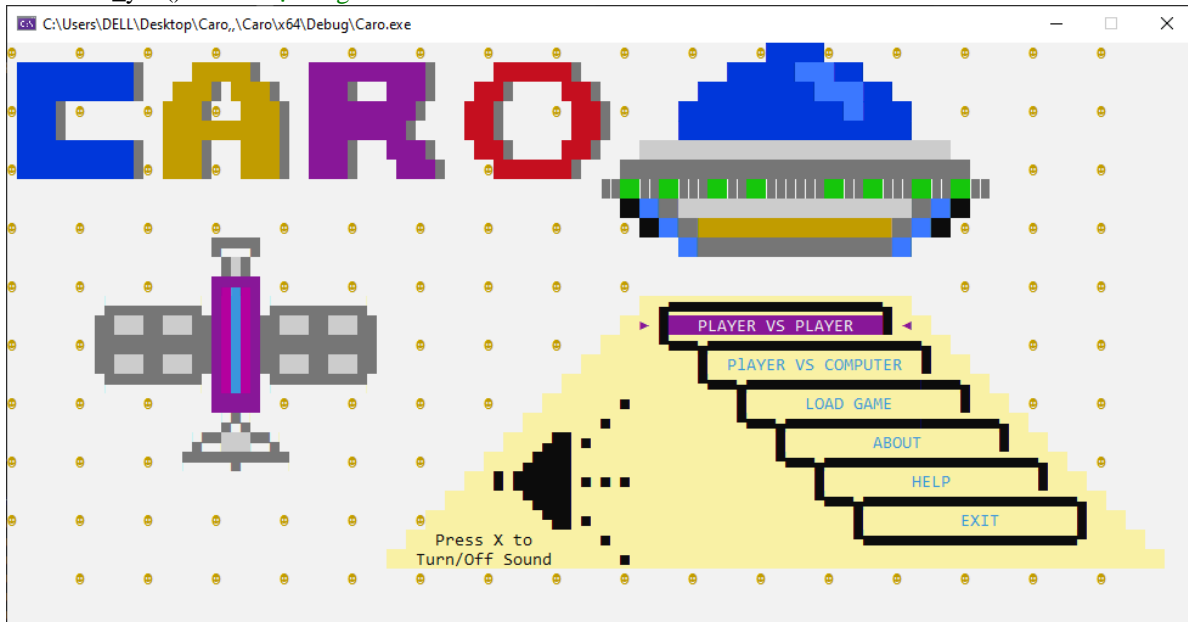
- Kỹ thuật di chuyển con trỏ.
- Kỹ thuật chèn âm thanh, thay đổi màu chữ, màu background.
- Kỹ thuật hiệu ứng thắng, thua, hòa, chuyển nền.
- Kỹ thuật cố định kích thước màn hình console.
- Kỹ thuật vẽ bằng mã ASCII.
- Kỹ thuật đọc, ghi và xóa file.
- Kỹ thuật xử dụng đa luồng cho thời gian trong game.
- Kỹ thuật xử dụng vòng lặp for, while, Sleep tạo hiệu ứng chuyển động.
- Thuật giải Heuristic để tính điểm và đánh giá nước đi cho máy và chức năng hint.
- Sử dụng vector trong mục Load Game để hiển thị các file đã lưu.

B. Mô tả code và các ý tưởng của từng chức năng code

Menu.h: Nhóm hàm chứa hiệu ứng và giao diện ở màn hình chờ.

- `void Menu()`: Vẽ in chữ hiển thị các chế độ ra màn hình.
- `void C()`: Vẽ chữ C trong CARO.
- `void A()`: Vẽ chữ A trong CARO.
- `void R()`: Vẽ chữ R trong CARO.
- `void _O()`: Vẽ chữ O trong CARO.
- `void c()`: Vẽ bóng của chữ C.
- `void _a()`: Vẽ bóng của chữ A.
- `void r()`: Vẽ bóng của chữ R.

- `void o()`: Vẽ bóng của chữ O.
- `void chay()`: Chạy vẽ màu chữ CARO thay đổi màu sắc liên tục.
- `void hello()`: Vẽ chữ Hello
- `void help()`: Thiết kế chế độ Help hướng dẫn người chơi.
- `void team()`: Giới thiệu thành viên nhóm.
- `void About_Menu()`: Thiết kế chế độ about giới thiệu nhóm.
- `int dichuyen(int y)`: Thiết kế chức năng di chuyển chọn chế độ trong menu.
- `void Move_Up()`: Di chuyển lên.
- `void Move_Down()`: Di chuyển xuống.
- `void kiểmtra(int y)`: Thiết kế kiểm tra tham số đầu vào là gì để chọn chế độ tương ứng với tham số đó.
- `void ve_thang_menu()`: Vẽ bậc thang chế độ cho người dùng chọn trong menu.
- `void nguoi_ngoai_hanh_tinh()`: Vẽ người ngoài hành tinh.
- `void UFO()`: Vẽ UFO.
- `void vetinh()`: Vẽ vệ tinh.
- `void Chay_anh_sang()`: Vẽ ánh sáng chuyển động liên tục.
- `void VeLoa()`: Vẽ Loa.
- `void Bat_Tat_Loa(int check)`: Vẽ bật/tắt loa.
- `void thank_you()`: Hàm hiệu ứng vẽ chữ Thank You khi nhấn Exit.





Bật/Tắt âm thanh ở màn hình chờ của Game

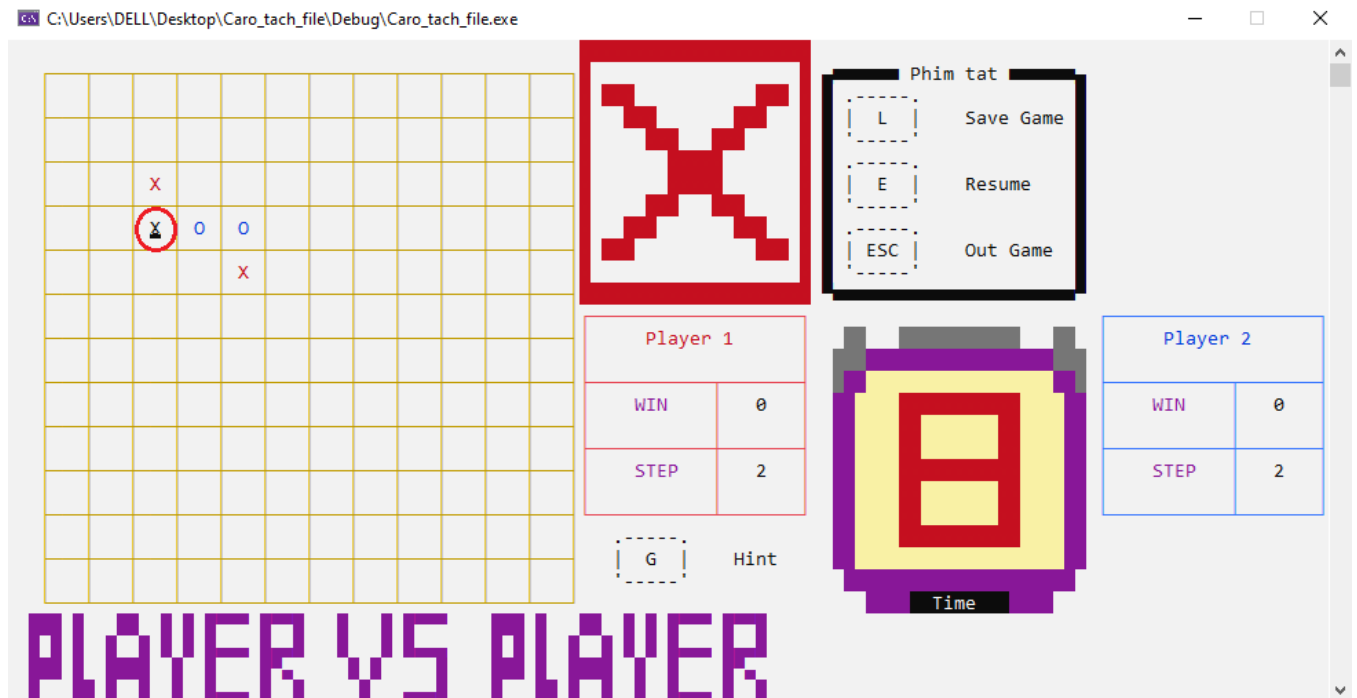
Cai_dat_he_thong.h: Nhóm hàm khởi tạo các cài đặt gốc cho trò chơi.

- `void SetColor(int background_color, int text_color):` Hàm này có chức năng như để thay đổi màu chữ
- `void gotoXY(int column, int line):` Hàm này có chức năng chuyển con trỏ đến tọa độ (dùng khi di chuyển trên bàn cờ và để vẽ các hiệu ứng).
- `void AnTroChuot():` Ẩn trỏ chuột.
- `void HienTroChuot():` Hiện trỏ chuột.

Chuc_nang_che_do.h: Nhóm hàm chứa các chức năng phụ hỗ trợ cho người dùng trong trò chơi.

- `void Huong_dan_phim_tat():` hàm hiệu ứng để vẽ bảng hướng dẫn các phím tắt.
- `int Want_to_continue():` hàm hiệu ứng hỏi người chơi có muốn tiếp tục chơi ván mới không (tiếp tục trả về 1 không tiếp tục trả về -1).
- `void Want_to_resume():` hàm hiệu ứng hỏi người chơi có muốn tiếp trận đấu khi trận đấu đang tạm dừng.
- `void Want_To_Save_Game():` hàm hiệu ứng hỏi người chơi có muốn lưu ván đấu khi Out Game.

- `void Hint()`: vẽ lên bàn cờ tại vị trí được gợi ý, người dùng có thể chọn đánh như chỉ dẫn hoặc không(sử dụng thuật giải Heuristic tương tự như nước đi cho máy).



`Chuyen_nen_vao_game.h`: Nhóm hàm hiệu ứng chuyển, thông báo cho người dùng biết trạng thái hiện hành của game.

- `void Loading_funtion()`: hàm hiệu ứng vẽ cảnh chuyển nền Loading cho Game.
- `void Chuyen_nen_vao_game()`: Tạo hiệu ứng chuyển nền cho Game(sử dụng hàm `Loading_funtion()`)
- `void Enter_player_Name()`: nhập tên hai người chơi X và O (có hướng dẫn cách di chuyển cho từng người chơi), người chơi X mặc định là Computer (nếu chế độ là người đấu với người sẽ nhập lại tên người chơi X).
- `void PX()`: hàm hiệu ứng để vẽ chữ Player X.
- `void PY()`: hàm hiệu ứng để vẽ chữ Player O.

`Game.h`: Nhóm hàm liên quan đến màn chơi

- `void ResetData()`: Hàm này có chức năng reset lại trạng thái bàn cờ và thời gian(giữ nguyên tỉ số).
- `void DrawBoard(int pSize)`: Hàm này có chức năng vẽ bàn cờ caro.
- `void StartGame()`: Hàm này có chức năng bắt đầu vào trò chơi khi người dùng nhấn "Start Game" trên Menu
- `int ProcessFinish(int pWhoWin)`: Hàm có chức năng xuất ra kết quả người thắng và thua (trả về giá trị: -1 là X thắng, 1 là O thắng, 0 là hòa, 2 là chưa có kết quả và ván đấu vẫn tiếp tục).
- `bool Check_matrix_full()`: Hàm kiểm tra nếu bàn cờ caro bị full và không còn ô nào trống.
- `bool kiem_tra_tren_mang_5x5_hai_duong_cheo(int x, int y)`: Hàm kiểm tra nếu một trong hai người chơi X hoặc O có đường thắng là đường chéo (bao gồm cả đường chéo chính và phụ).
- `bool kiem_tra_tren_mang_5x5_cac_duong_thang_hang(int x, int y)`: Hàm kiểm tra nếu một trong hai người chơi X hoặc O có đường thắng là đường thẳng.

- **bool** check_X_or_O_win(): Hàm có chức năng kiểm tra X hay O thắng.
- **void** color_for_win(): Hàm có chức năng điều chỉnh 5 nước thắng thành màu khác cho dễ phân biệt.
- **int** TestBoard(): Kiểm tra trạng thái bàn cờ đã có nước đi thắng chưa (hoặc hòa)(giá trị trả về: 0 là hòa, -1 là X thắng, 1 là O thắng).
- **int** CheckBoard(**int** pX, **int** pY): Hàm này có chức năng biết lượt nào nhập "X" và lượt nào nhập "O" khi nhấn Enter(giá trị trả về là -1 đối với X và 1 đối với O).
- **void** MoveRight(): Di chuyển sang phải.
- **void** MoveLeft(): Di chuyển sang trái.
- **void** MoveDown(): Di chuyển xuống.
- **void** MoveUp(): Di chuyển lên.
- **void** ve_X(**int** a, **int** b): Hàm hiệu ứng có chức năng vẽ Ô vuông X màu đỏ hiển thị lượt đi của X.
- **void** Ve_O(**int** a, **int** b): Hàm hiệu ứng có chức năng vẽ Ô vuông O màu xanh hiển thị lượt đi của O.
- **void** Draw_Win_and_Step_for_X(): Vẽ bảng số lượt win và số nước đã đi cho X.
- **void** Draw_Win_and_Step_for_O(): Vẽ bảng lượt win và số nước đã đi cho O.
- **void** Count_Step_X(): Đếm số lượt quân X đã đánh.
- **void** Count_Step_O(): Đếm số lượt quân O đã đánh.
- **void** Draw_X_OR_O_For_Win(**int** win_number): Vẽ hiệu ứng xuất hiện kết quả X hay O thắng hoặc hòa.
- **void** Draw_Table_X_or_win(**int** win_number): Vẽ BẢNG xuất hiện kết quả X hay O thắng hoặc hòa.

Player_vs_Computer.h: Nhóm hàm dành cho chế độ chức năng chơi với máy.

- **void** PvC(): Vẽ chữ trong chế độ Player vs Computer.
- **void** dif(): Vẽ chữ difficult bên cạnh chế độ Player vs Computer khi ở chế độ khó.
- **void** Nor(): Vẽ chữ normal bên cạnh chế độ Player vs Computer khi ở chế độ bình thường.
- **void** PvCLV(): Hàm vẽ hiệu ứng chế độ khó/dễ.
- **long** TC_Doc(**int** cur_row, **int** cur_col): Tính điểm tấn công theo hàng dọc.
- **long** TC_Ngang(**int** cur_row, **int** cur_col): Tính điểm tấn công theo hàng ngang.
- **long** TC_Cheo_Xuoi(**int** cur_row, **int** cur_col): Tính điểm tấn công theo đường chéo chính.
- **long** TC_Cheo_Nguoc(**int** cur_row, **int** cur_col): Tính điểm tấn công theo đường chéo phụ.
- **long** PN_Doc(**int** cur_row, **int** cur_col): Tính điểm phòng ngự theo hàng dọc.
- **long** PN_Ngang(**int** cur_row, **int** cur_col): Tính điểm phòng ngự theo hàng ngang.
- **long** PN_Cheo_Xuoi(**int** cur_row, **int** cur_col): Tính điểm phòng ngự theo đường chéo chính.
- **long** PN_Cheo_Nguoc(**int** cur_row, **int** cur_col): Tính điểm phòng ngự theo đường chéo phụ.
- **long** Diem_TC(**int** i, **int** j): Tổng hợp các hàm Tấn công phía trên.
- **long** Diem_PN(**int** i, **int** j): Tổng hợp các hàm Phòng ngự phía trên.
- **_POINT** Nuoc_di(): Phân tích nước đi(giá trị trả về là vị trí trên bàn cờ).
- **void** Com_play(): Computer đánh cờ vào nước đi (lấy từ hàm Nuoc_di()) đã phân tích.
- **void** Player_Vs_Com(): Chế độ người đấu với máy.

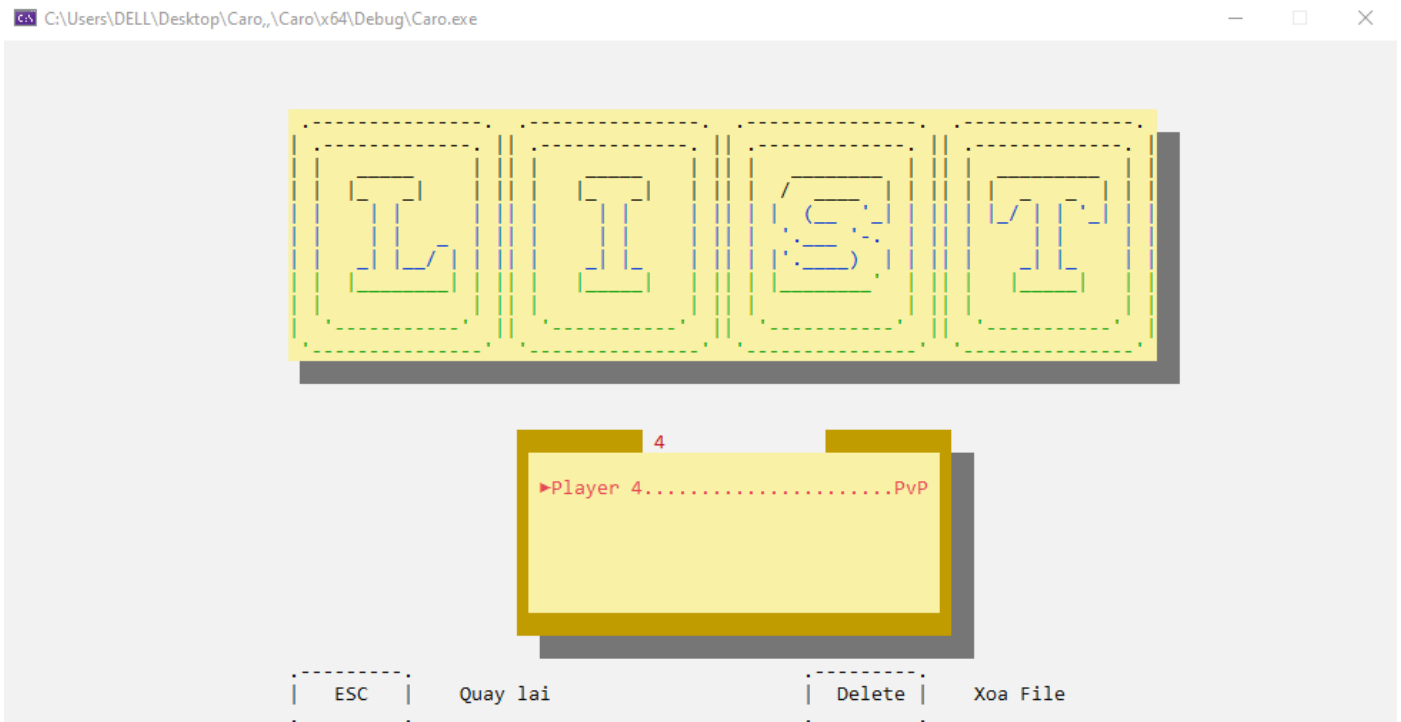
Player_vs_Player.h: Nhóm hàm cho chế độ người đấu với người.

- `void PvP()`: Hàm hiệu ứng vẽ chữ chế độ Player vs Player
- `void Player_vs_Player()`: Chế độ người đấu với người

Save_and_Load.h: Nhóm hàm liên quan đến lưu trữ

- `bool Enter_File_Name(char data[16])`: Nhập tên file, nếu file đã lưu rồi thì ấn lưu sẽ không cần nhập lại tên file.
- `void SaveGame(int n)`: Lưu ván đấu.
- `void Danh_Sach_File_Luu()`: Hàm vẽ hiệu ứng cho chữ List trong phần Load Game.
- `bool DeleteFile(const char* file_path)`: Xóa File(trong phần Load Game có thể chọn file để xóa).
- `void Delete_Name(vector<string>& a, vector<string>& main, int& cur_pos, int& first_mem)`: Xóa File hiển thị trên màn hình
- `void File_Name(vector<string> s)`: Tạo file Name.txt mới.
- `bool StringCmp(string str1, string str2)`: So sánh chuỗi con nhập vào với các chuỗi tên file, liên quan đến Search File (không phân biệt chữ thường và hoa).
- `void Print(vector<string>a, int cur_pos, int first_mem, char arrow)`: In danh sách các File ra màn hình.
- `void Arrow(vector<string> a, int& first_mem, char arrow)`: Vẽ mũi tên hiển thị lên xuống trong phần Load Game.
- `void Go_Up_List(vector<string> a, int& cur_pos, int& first_mem, char arrow)`: Cập nhập lại danh sách File được in ra khi di chuyển lên.
- `void Go_Down_List(vector<string> a, int& cur_pos, int& first_mem, char arrow)`: Cập nhập lại danh sách File được in ra khi di chuyển xuống.
- `void list(vector<string>& a, vector<string>& main, char cmd, int& cur_pos, int& first_mem)`: Kết hợp giữa các hàm trên.
- `string Keyword(string& str, int& pos, char& cmd, int& cur_pos, int& first_mem)`: Hàm nhập kí tự để so sánh chuỗi tìm các file có chứa các kí tự vừa nhập, liên quan Search File.

- `void List_and_search(string& Data)`: Kết hợp giữa hàm list và Keyword (khi người dùng nhập bất kì kí tự nào thì danh sách file lưu sẽ liên tục được cập nhật để tiện cho người dùng theo dõi).
- `void LoadGame(string data)`: Mở File đã lưu tiếp tục trận đấu(có hiển thị các chế độ chơi để người dùng có thể biết).



Time.h: Nhóm hàm liên quan đến chức năng thời gian trong mỗi màn game.

- `void Ve_clock()`: hàm hiệu ứng vẽ hình đồng hồ đếm ngược thời gian.
- `void count_time(int h)`: hàm hiệu ứng vẽ các số đếm ngược thời gian bắt đầu từ giây thứ 10.
- `void color_for_win_version_time(int number)`: Vẽ hiệu ứng chiến thắng cho người chơi còn lại khi hết thời gian.

3. Tài liệu tham khảo

- Tài liệu hướng dẫn của giáo viên: https://courses.fit.hcmus.edu.vn/pluginfile.php/189847/mod_resource/content/1/DoAnCaro.pdf
- Làm đồng hồ cho Game: <https://www.youtube.com/watch?v=hbiu9e5ZbuQ>
- Bảng màu Console: <https://www.phanxuanchanh.com/2021/01/08/lap-trinh-c-c-doi-mau-console/>
- Định dạng màn hình console: <https://codelearn.io/sharing/windowsh-va-ham-dinh-dang-console-p1>
- Xử lý AI cờ Caro: <https://www.youtube.com/watch?v=59EaoU-wHy8>