

TRƯỜNG ĐẠI HỌC BÁCH KHOA TP.HCM
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



MÔN KỸ THUẬT LẬP TRÌNH
BÁO CÁO BÀI TẬP LỚN
GAME REVERSI
GVHD: Võ Thanh Hùng

Sinh viên thực hiện: Lê Nhật Thanh MSSV: 41303594

Giới thiệu:

Game REVERSI là một game trí tuệ đơn giản 2 người chơi đối kháng với 1 lưới 8x8 có thể chứa quân trắng, quân đen, hoặc trống.

Luật chơi:

- 2 người chơi, một người chơi quân trắng, một người chơi quân đen.
- Đến lượt chơi của mình thì người chơi buộc phải đặt quân cờ vào vị trí hợp lệ, nếu không tồn tại bất kì vị trí hợp lệ nào có thể đặt quân thì nhường quyền cho người chơi còn lại.
- Vị trí đặt quân hợp lệ là vị trí sao cho khi người chơi đặt quân vào vị trí đó thì có thể “ăn” được quân của đối phương (cần chơi game để hiểu rõ).
- Trò chơi kết thúc trong 2 trường hợp sau:
 - Bàn cờ không còn ô trống
 - Cả 2 người chơi không có bất kì vị trí nào có thể đặt quân .
- Luật thắng thua: khi trò chơi kết thúc thì người chơi nào có số lượng quân cờ trên bàn cờ nhiều hơn là người chiến thắng.

Sử dụng ngôn ngữ C++ và Prototype các hàm có sẵn (do giảng viên cung cấp).

Phân tích, thiết kế, giải thuật:

Chương trình sử dụng các khái niệm cơ bản như con trỏ, con trỏ hàm, cấu trúc (struct)

Các Prototype hàm được cung cấp (trong file Assignment.cpp):

```
struct position{
    int x;//row
    int y;//column
}; //định nghĩa kiểu cấu trúc cho hàng và cột trong trạng thái bàn cờ

print_position(struct position pos); //in một trạng thái bàn cờ dạng tọa độ (x, y).

print_lst_position(struct position* lst_pos, int size);
//in danh sách các trạng thái bàn cờ.

struct position* generate_list_valid_moves(int** board_state, int player, int &size); //tạo ra danh sách các trạng thái hợp lệ của một người chơi.

update_board_state(int** board, struct position pos, int player);
//cập nhật trạng thái bàn cờ sau một bước đi.
```

Game được xây dựng trên console (đồ họa console). Có sử dụng thêm thư viện ngoài **console.h** để vẽ bàn cờ (đồ họa cơ bản trên console). Thư viện bao gồm các hàm như:

```
gotoXY (int column, int line); // di chuyển con trỏ để vị trí mong muốn.
// màn hình console mặc định có kích thước 25x80 pixel.
clrscr(); //xóa màn hình console.
```

```
TextColor (int color); //tạo màu cho font chữ, nền (có 256 màu);
```

Và bao gồm các hàm khác nhưng không sử dụng.

Giải thuật dùng cho hai hàm tạo danh sách các trạng thái hợp lệ:

Xét từng điểm một trong bàn cờ (bàn cờ có 64 điểm tất cả).

Giả sử người chơi kế tiếp là người chơi 1 (đánh quân “1”).

Tìm tất cả các điểm có trạng thái là “2” trong bàn cờ. Tại một vị trí “2” tìm được:

Xét 8 điểm xung quanh vị trí “2” này, nếu tồn tại điểm “0” (điểm trống (chưa đánh) trong bàn cờ). Điểm “0” nằm ở vị trí nào so với điểm “2” đang xét thì cho một vòng lặp **for** chạy theo hướng ngược lại, nếu gặp điểm “2” tiếp thì quay lại vòng for và chạy tiếp, nếu gặp điểm “0” thì thoát khỏi vòng for và xét điểm tiếp theo trong 8 điểm, nếu gặp điểm “1” thì thoát khỏi vòng for và vị trí điểm “0” là một vị trí cần tìm (Có sự khống chế hàng âm, cột âm và hàng, cột lớn hơn 7 bằng các câu lệnh if, lệnh while và lệnh nhảy goto).

Cứ làm như vậy hết bàn cờ thì sẽ tìm ra được 1 danh sách các trạng thái hợp lệ cho 1 người chơi.

Giải thuật dùng cho hàm cập nhật trạng thái bàn cờ sau 1 bước đi:

Input: trạng thái bàn cờ, vị trí đánh, người chơi.

Vị trí đánh ở đây là một vị trí đánh hợp lệ (trong lúc chơi, nếu người chơi đánh vị trí không hợp lệ thì có thông báo đánh lại cho đến khi nào nằm trong danh sách vị trí hợp lệ, việc khống chế này nằm trong hàm main() để đảm bảo đưa vào hàm cập nhật này một trạng thái hợp lệ).

Giả sử người đang chơi là người chơi 2:

Xét tám điểm xung quanh vị trí hợp lệ này (có khống chế hàng cột âm và hàng cột lớn hơn 7 bằng lệnh if, điều kiện trong for, lệnh goto). Cũng giống như giải thuật trên. Nếu một vị trí là điểm “1”, thì cho một vòng for chạy theo hướng điểm “1” này so với điểm hợp lệ, nếu gặp điểm “1” thì cho xét tiếp, nếu gặp điểm “0” thì thoát khỏi vòng lặp, nếu gặp điểm “2” thì cũng thoát khỏi vòng lặp (có một biến đếm để kiểm tra xem có gặp điểm “2”)

Xét biến đếm (khác 0) để chuyển tất cả trạng thái “1” thành trạng thái “2” và chuyển điểm hợp lệ thành điểm “2” (bàn cờ đã được cập nhật).

Vẽ bàn cờ và hoàn thành game:

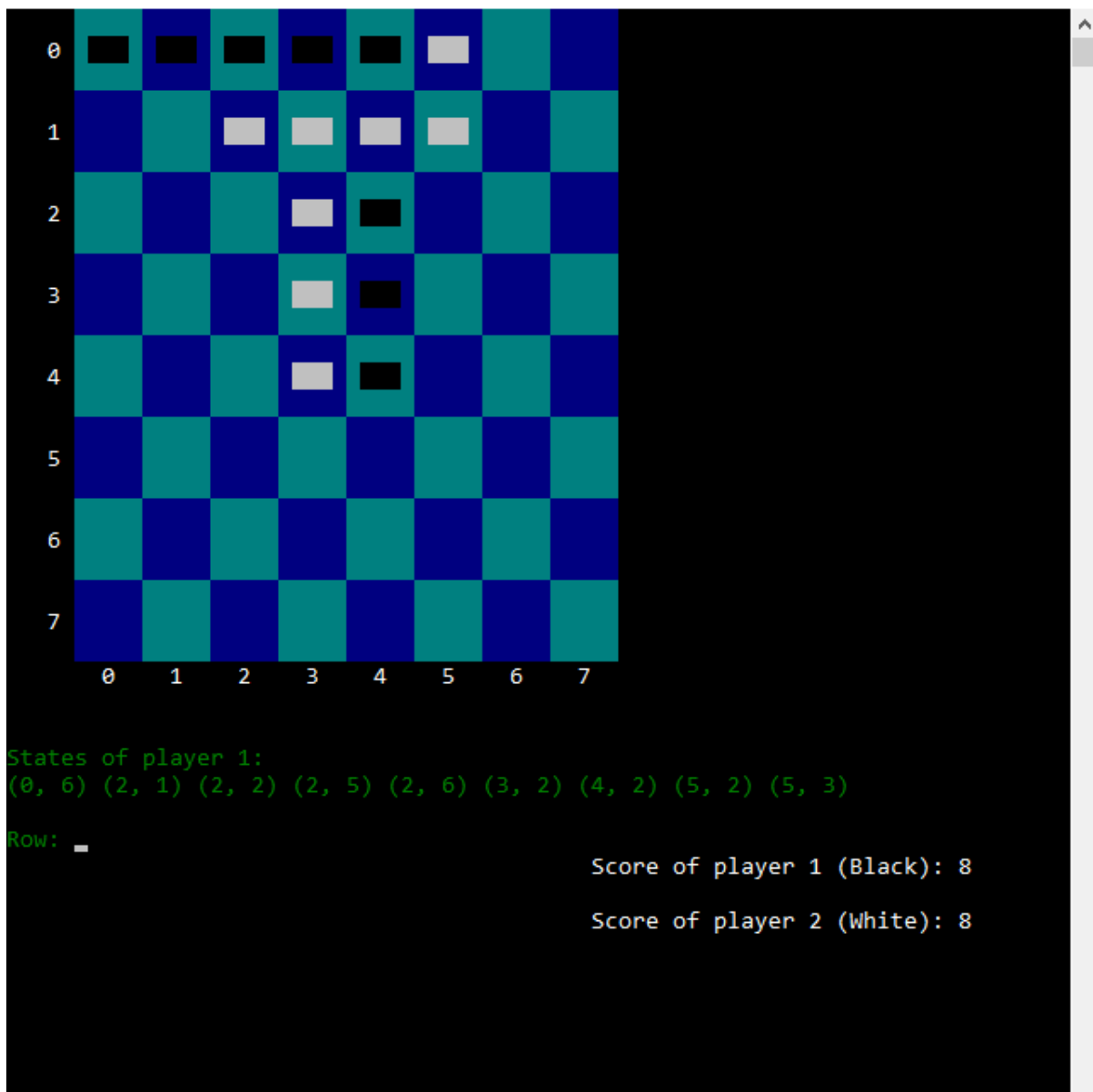
Dùng các hàm trong thư viện “console.h” để vẽ bàn cờ. Sau đó chuyển từ tọa độ console sang tọa độ mới của bàn cờ (trong hàm main()).

Dùng một vòng lặp do...while để duy trì các lượt chơi cho đến khi cả hai không còn trạng thái đi hợp lệ thì kết thúc trò chơi.

Ngoài ra trong còn dùng câu lệnh printf , TextColor, GotoXY để vẽ dòng “game over”.

Một số hình ảnh của game:

D:\AssignmentKTLT\Debug\AssignmentKTLT.exe





Mục đích và kết quả:

Nâng cao khả năng tư duy giải thuật dùng ngôn ngữ C++ về lập trình hướng cấu trúc.

Kết quả tạo ra một game đơn giản trên console.

Hướng phát triển:

- Cải thiện giải thuật, tối ưu thuật toán.
- Phát triển về đồ họa, thêm sự kiện click chuột.
- Về menu game như lưu lại game để chơi lại một lúc khác....

Kết luận:

Game đạt được tiêu chí đề ra, nhưng không tránh khỏi một số thiếu sót. Nhưng quan trọng là qua bài tập lớn này, sinh viên rèn được rất nhiều khả năng tư duy lập trình, củng cố nhiều kiến thức. Tạo cơ sở, tiền đề cho những bạn đam mê lập trình tự tin phát triển.

Em rất mong nhận được sự góp ý từ thầy!

Em xin chân thành cảm ơn!

Thành phố Hồ Chí Minh, ngày 11 tháng 12 năm 2015