SOURCE CODE

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include <stdbool.h>

char board[3][3] = {{'.','.','.'}, {'.','.','.'}, {'.','.','.'}};

char check()

{

for(int i = 0; i < 3; i++){

if(board[i][0] == board[i][1] && board[i][0] == board[i][2] && board[i][0] != '.'){

return board [i][0];

}

}

for(int j = 0; j < 3; j++){

if(board[0][j] == board[1][j] && board[0][j] == board[2][j] && board[0][j] != '.'){

return board [0][j];}

}

if(board[0][0] == board[1][1] && board[0][0] == board[2][2] && board[0][0] != '.'){

return board[0][0];

}

if(board[0][2] == board[1][1] && board[0][2] == board[2][0] && board[0][2]!= '.'){

return board[0][0];

}

int cek = 1;

for(int i=0; i<3; i++){

for(int j=0; j<3; j++){

if(board[i][j] == '.') {

cek = 0;

break;

}

}

}

if(cek) return 'D';

return '-';

}

int minimax(int depth, bool isMaximizing){

int score;

char result = check();

if (result=='O') {

score = 1;

return score;

}

else if (result=='X') {

score = -1;

return score;

}

else if (result=='T') {

score = 0;

return score;

}

if(isMaximizing){

int bestScore = -2;

for(int i=0; i<3; i++){

for(int j=0; j<3; j++){

if(board[i][j]=='.'){

board[i][j]='O';

score = minimax(depth+1, false);

board[i][j]='.';

if(score > bestScore){

bestScore = score;

}

}

}

}

return bestScore;

} else {

int bestScore = 2;

for(int i=0; i<3; i++){

for(int j=0; j<3; j++){

if(board[i][j]=='.'){

board[i][j]='X';

score = minimax(depth+1, true);

board[i][j]='.';

if(score < bestScore){

bestScore = score;

}

}

}

}

return bestScore;

}

}

void bestMove(){

int score;

int bestScore = -2, x, y;

for(int i=0; i<3; i++){

for(int j=0; j<3; j++){

if(board[i][j]=='.'){

board[i][j]='O';

score = minimax(0, false);

board[i][j]='.';

if(score > bestScore){

bestScore = score;

x=i; y=j;

}

}

}

}

board[x][y]='O';

}

void gameboard()

{

printf ("\n TicTacToe\n\n");

printf(" %c | %c | %c \n", board[0][0], board[0][1], board[0][2]);

printf("\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_\n");

printf(" %c | %c | %c \n", board[1][0], board[1][1], board[1][2]);

printf("\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_\n");

printf(" %c | %c | %c \n", board[2][0], board[2][1], board[2][2]);

printf(" | | \n\n");

}

int main()

{

int turn, level, win, i, player, a, b;

char player1 = 'X';

char player2 = 'O';

printf("=======TICTACTOE=======\n");

printf("\tLevel 1\n");

printf("\tLevel 2\n");

printf("\tLevel 3\n");

printf("\tLevel 4\n");

printf("\nchoose your level: ");

scanf("%d", &level);

system("cls");

gameboard();

//Level I

if (level == 1 || level == 2){

for(i = 1; i < 10; i ++){

if(i%2 == 1){

player = 1;

} else if (i%2 == 0){

player = 2;

}

printf("player %d your move = ", player);

scanf ("%d %d", &a, &b);

printf ("\n");

if (board[a][b] == '.'){

if(player == 1){

board[a][b] = 'X';

} else if (player == 2){

board[a][b] = 'O';

}

} else if (board[a][b] != '.' || a >= 3 || b >=3){

system("cls");

printf(" Invalid\n");

gameboard();

printf("player %d your move = ", player);

scanf ("%d %d", &a, &b);

printf ("\n");

if(player == 1){

board[a][b] = 'X';

} else if (player == 2){

board[a][b] = 'O';

}

}

system("cls");

printf ("\n");

gameboard();

char result = check();

if (level == 1){

if(result != '-'){

printf("GAME FINISHED");

break;

}

}

if (level == 2){

if(result == 'O'){

printf("Player 2 WIN");

break;

}else if(result == 'X'){

printf("Player 1 WIN");

break;

}else if(result == 'D'){

printf("DRAW");

break;

}

}

}

}

if(level == 3){

printf("who play first? \n1 for player\n2 for BOT\n");

scanf("%d", &turn);

for(i = turn; i < turn+9; i ++){

if(i%2 == 1){

printf("Player your move = ");

scanf ("%d %d", &a, &b);

printf ("\n");

if (board[a][b] == '.'){

if(i%2 == 1){

board[a][b] = 'X';

}

} else if (board[a][b] != '.' || a >= 3 || b >=3){

system("cls");

printf(" Invalid\n");

gameboard();

printf("player your move = ");

scanf ("%d %d", &a, &b);

printf ("\n");

if(i%2 == 1){

board[a][b] = 'X';

}

}

}else {

printf("BOT turn\n");

srand(time(0));

for(int l = 1; l > 0;l++){

a = rand()%3;

b = rand()%3;

if(board[a][b]=='.')break;

}

board[a][b] = 'O';

}

system("cls");

printf ("\n");

gameboard();

char result = check();

if(result == 'O'){

printf("BOT WIN");

break;

}else if(result == 'X'){

printf("Player WIN");

break;

}else if(result == 'D'){

printf("DRAW");

break;

}

}

}

if(level == 4){

//gameboard();

int turn, cnt = 0;

printf("Bot go first? (1/0) : ");

scanf("%d", &turn);

if(turn) cnt++;

while(1){

if(!cnt){

printf("\nPlayer your move: ");

scanf("%d %d", &a, &b);

if(board[a][b]=='X' || board[a][b]=='O' || a<0 || a>2 || b<0 || b>2) continue;

board[a][b] = 'X';

cnt=1;

}

else {

//printf("\nBot\n");

bestMove();

cnt=0;

}

gameboard();

int result = check();

if(result!='-') {

if(result == 'O') printf("Bot wins.\n");

else if (result == 'X') printf("Player wins.\n");

else if (result == 'D') printf("Draw.\n");

break;

}

}

}

}

/\*int repeat;

printf("\nPlay again? (1/0) : ");

scanf("%d", &repeat);

if (repeat == 0) return 0;\*/