#define \_CRT\_SECURE\_NO\_WARNINGS 1

//头文件包含

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define ROW 3

#define COL 3

//声明函数

//初始化棋盘的

void InitBoard(char board[ROW][COL], int row, int col);

//打印棋盘的

void DisplayBoard(char board[ROW][COL], int row, int col);

//玩家下棋

void PlayerMove(char board[ROW][COL], int row, int col);

//电脑下棋

void ComputerMove(char board[ROW][COL], int row, int col);

//检测游戏输赢

char CheckWin(char board[ROW][COL], int row, int col);

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include "game.h"

void InitBoard(char board[ROW][COL], int row, int col)

{

int i = 0;

int j = 0;

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

board[i][j] = ' ';

}

}

}

//

//void DisplayBoard(char board[ROW][COL], int row, int col)

//{

// int i = 0;

// for (i = 0; i < row; i++)

// {

// //打印数据

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

// //打印分割行

// if (i<row-1)

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

// }

//}

//

void DisplayBoard(char board[ROW][COL], int row, int col)

{

int i = 0;

for (i = 0; i < row; i++)

{

//打印数据

int j = 0;

for (j = 0; j < col; j++)

{

printf(" %c ", board[i][j]);

if (j<col-1)

printf("|");

}

printf("\n");

//打印分割行

if (i < row - 1)

{

for (j = 0; j < col; j++)

{

printf("---");

if (j<col-1)

printf("|");

}

}

printf("\n");

}

}

void PlayerMove(char board[ROW][COL], int row, int col)

{

int x = 0;

int y = 0;

printf("玩家走:>\n");

while (1)

{

printf("请输入坐标:>");

scanf("%d%d", &x, &y);

//如果坐标范围合法

if (x >= 1 && x <= row && y >= 1 && y <= col)

{

if (board[x - 1][y - 1] == ' ')

{

board[x - 1][y - 1] = '\*';

break;

}

else

{

printf("该坐标被占用，请重新输入\n");

}

}

else

{

printf("坐标非法，请重输入\n");

}

}

}

void ComputerMove(char board[ROW][COL], int row, int col)

{

printf("电脑走:>\n");

while (1)

{

int x = rand() % row;//0-2

int y = rand() % col;//0-2

if (board[x][y] == ' ')

{

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

break;

}

}

}

static int IsFull(char board[ROW][COL], int row, int col)

{

int i = 0;

int j = 0;

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

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

{

return 0;

}

}

}

return 1;//没空格

}

char CheckWin(char board[ROW][COL], int row, int col)

{

int i = 0;

//行是否有3个相等

for (i = 0; i < row; i++)

{

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

{

return board[i][0];

}

}

//列的判断

for (i = 0; i < col; i++)

{

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

{

return board[0][i];

}

}

//对角线

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

{

return board[1][1];

}

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

{

return board[1][1];

}

//判断是否为平局

//判断棋盘是否满了？ - 就是棋盘上是否有空格？

if (IsFull(board, row, col) == 1)

{

return 'Q';

}

//不是平局，游戏继续

return 'C';

}

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include "game.h"

void menu()

{

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\* 1. play \*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\* 0. exit \*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

}

void game()

{

char ret = 0;

//设计一下三子棋游戏

//存储数据的

char board[ROW][COL] = {0};//数组应该初始化为空格

InitBoard(board, ROW, COL);//初始化棋盘 - 空格

//打印棋盘

DisplayBoard(board, ROW, COL);

//分析以下发现：

//在游戏玩的过程中

//1. 玩家赢 - '\*'

//2. 电脑赢 - '#'

//3. 平局了 - 'Q'

//4. 继续 - 'C'

while (1)

{

PlayerMove(board, ROW, COL);

//判断输赢

ret = CheckWin(board, ROW, COL);

if (ret != 'C')

{

break;

}

DisplayBoard(board, ROW, COL);

ComputerMove(board, ROW, COL);

//判断输赢

ret = CheckWin(board, ROW, COL);

if (ret != 'C')

{

break;

}

DisplayBoard(board, ROW, COL);

}

if (ret == '\*')

{

printf("玩家赢\n");

}

else if (ret == '#')

{

printf("电脑赢\n");

}

else if (ret == 'Q')

{

printf("平局\n");

}

DisplayBoard(board, ROW, COL);

}

int main()

{

int input = 0;

srand((unsigned int)time(NULL));

do

{

menu();

printf("请选择:>");

scanf("%d", &input);

switch (input)

{

case 1:

game();//三子棋游戏

break;

case 0:

printf("退出游戏\n");

break;

default:

printf("选择错误\n");

break;

}

//

} while (input);

return 0;

}