

C语言简单扫雷

Game two

在一张棋盘中有多个不确定是雷还是安全区域的格子，当点击之后若没有雷则会在该区域显示周围八个格子雷的数目，若点雷则游戏结束。今天实现一下C语言实现初阶简单扫雷

@TOC

一、思路框架

1. 显示一个简单的交互菜单（选择游戏）
2. 创建一个mine棋盘用来存放雷并初始化
3. 创建一个show 棋盘用来和用户交互并初始化
4. 玩家选择想要扫描的坐标
5. 判定游戏是否满足结束条件
6. 若没有踩雷则在刚扫描的区域上显示周围八个区域的地雷数目并继续游戏
7. 若全部扫描完非雷区域则游戏结束

二、代码过程实现

1.主函数 test.c

代码如下：

```
# define _CRT_SECURE_NO_WARNINGS 1
#include "game.h"

//实现扫雷游戏
void menu() {
    printf("-----\n");
    printf("          1.Play          \n");
    printf("          0.Exit          \n");
    printf("-----\n");
}

void game() {
    //初始化两个棋盘，一个存放雷的信息
    //另一个存放排查过程的棋盘
    char mine[ROWS][COLS] = { 0 };
    char show[ROWS][COLS] = { 0 };

    //mine初始化全为“0”，表示棋盘上没有雷
    //show初始化全为“*”，表示还没有排查到信息
    initBoard(mine, ROWS, COLS, '0');
    initBoard(show, ROWS, COLS, '*');

    //displayBoard(mine, row, col); //考虑不打印含雷区信息的棋盘
```

```
displayBoard(show, ROW, COL); //显示时只打印九行九列
//布置雷
setMine(mine,ROW,COL);
displayBoard(mine, ROW, COL);
//排雷
findMine(mine, show, ROW,COL);

}
int main() {
    int input;
    srand((unsigned int)time(NULL));
    do
    {
        printf("请输入你的选择: ");
        scanf("%d", &input);
        switch (input)
        {
            case 1:
                game();
                break;
            case 0:
                printf("退出游戏\n");
                break;
            default:
                break;
        }
    } while (input);
    return 0;
}
```

简单菜单:

```
-----
                1.Play
                0.Exit
-----
请输入你的选择: 1
0 1 2 3 4 5 6 7 8 9
1 * * * * * * * * *
2 * * * * * * * * *
3 * * * * * * * * *
4 * * * * * * * * *
5 * * * * * * * * *
6 * * * * * * * * *
7 * * * * * * * * *
8 * * * * * * * * *
9 * * * * * * * * *
0 1 2 3 4 5 6 7 8 9
1 1 0 0 0 0 0 0 1 0
2 0 0 0 1 0 0 0 0 0
3 0 0 1 0 0 0 0 0 1
4 1 0 0 0 0 0 0 0 0
5 0 0 0 0 0 0 0 0 1
6 0 0 0 0 0 0 0 0 0
7 0 0 1 0 0 0 0 1 0
8 0 0 0 0 0 0 0 0 0
9 0 0 1 0 0 0 0 0 0
请输入你要排雷的坐标:  CSDN @期邈云汉
```

2. 游戏函数代码

代码如下:

```
# define _CRT_SECURE_NO_WARNINGS 1
#include "game.h"
//初始化11 * 11的棋盘
void initBoard(char board[ROWS][COLS], int rows, int cols, char set) {
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            board[i][j] = set;
        }
    }
}
//打印棋盘
void displayBoard(char board[ROWS][COLS], int row, int col) {
    //打印列号
    for (int j = 0; j <= col; j++)
    {
```

```

        printf("%d ", j);
    }
    printf("\n");
    for (int i = 1; i <= row; i++)
    {
        //打印行号
        printf("%d ", i);
        for (int j = 1; j <= col; j++)
        {
            printf("%c ", board[i][j]);
        }
        printf("\n");
    }
}

//布置雷的位置
void setMine(char mine[ROWS][COLS], int row, int col) {
    //假设布置10个雷
    int count = 10;
    while (count) {
        int x = rand() % row + 1; //x, y坐标生成1 ~ 9
        int y = rand() % col + 1;
        if (mine[x][y] == '0') //说明可以布置雷
        {
            mine[x][y] = '1';
            count--;
        }
    }
}

//排雷
//统计一个坐标周围的雷数量
int get_mineCount(char mine[ROWS][COLS], int x, int y) {
    return (mine[x - 1][y] + mine[x - 1][y - 1] + mine[x - 1][y + 1] +
            mine[x][y - 1] + mine[x][y + 1] +
            mine[x + 1][y - 1] + mine[x + 1][y] + mine[x + 1][y + 1] - 8 * '0'); //
    周围8个坐标里的值加起来再减去8*'0',表示有几个雷
}

void findMine(char mine[ROWS][COLS], char show[ROWS][COLS], int row, int col) {
    int x = 0;
    int y = 0;
    int win = 0;
    while (win < row*col-COUNT) //获胜条件是排完71个雷
    {
        printf("请输入你要排雷的坐标: ");
        scanf("%d %d", &x, &y);
        if (x >= 1 && x <= row && y >= 1 && y <= col) {
            //坐标合法, 先判断该坐标是否为雷。统计该坐标周围的炸弹数量
            //加条件是重复输入已经排过的坐标会导致win++, 加入一个起始if语句判断坐标是否已经排
            查过
            if (show[x][y] == '*')
            {
                if (mine[x][y] == '1')
                {
                    printf("很遗憾, 地雷游戏失败! ");
                    displayBoard(mine, ROW, COL);
                    break;
                }
            }
        }
    }
}

```

```

    }
    else
    {
        int count = get_mineCount(mine, x, y); //将得到的炸弹数量放入
show数组的对应位置，表示数量
        show[x][y] = count + '0'; //加'0'为了让数组作为字符存入
        displayBoard(show, ROW, COL);
        win++; //排雷成功
    }
}
else
{
    printf("该坐标已经排查过\n");
}

}
else
{
    printf("坐标非法！请重新输入\n");
}
}
if (win == row*col-COUNT) //踩到雷了，那么循环提前退出，就不满足if条件
{
    printf("恭喜你，排雷成功\n");
    displayBoard(mine, ROW, COL);
}
}
}

```

3. 头文件

实现9 * 9的扫雷

为了方便位于边界的各自进行设计，将棋盘扩充一圈

```

#pragma once
#include <stdio.h>
#include <time.h>
#include <stdlib.h>
//例如实现9 * 9的扫雷
//为了方便位于边界的各自进行设计，将棋盘扩充一圈
# define ROW 9
# define COL 9

# define ROWS ROW+2
# define COLS COL+2

#define COUNT 10 //雷的数量
void initBoard(char board[ROWS][COLS], int rows, int cols, char set); //声明初始化棋
盘函数
void displayBoard(char board[ROWS][COLS], int row, int col); //打印棋盘
void setMine(char mine[ROWS][COLS], int row, int col); //布置雷的分布
void findMine(char mine[ROWS][COLS], char show[ROWS][COLS], int row, int col);
//排雷

```


1.Play
0.Exit

请输入你的选择: 1

0 1 2 3 4 5 6 7 8 9

1 * * * * * * * * *

2 * * * * * * * * *

3 * * * * * * * * *

4 * * * * * * * * *

5 * * * * * * * * *

6 * * * * * * * * *

7 * * * * * * * * *

8 * * * * * * * * *

9 * * * * * * * * *

0 1 2 3 4 5 6 7 8 9

1 0 0 0 0 0 1 0 0 0

2 0 0 0 0 0 0 0 0 0

3 0 0 0 1 1 0 0 0 0

4 0 0 0 0 0 0 0 0 0

5 0 0 0 1 1 0 0 0 0

6 0 0 1 0 0 0 0 0 0

7 1 1 0 0 0 0 0 0 0

8 0 0 0 1 0 0 0 0 0

9 0 0 0 0 1 0 0 0 0

请输入你要排雷的坐标: 1 1

0 1 2 3 4 5 6 7 8 9

1 0 * * * * * * * *

2 * * * * * * * *

3 * * * * * * * *

4 * * * * * * * *

5 * * * * * * * *

6 * * * * * * * *

7 * * * * * * * *

8 * * * * * * * *

9 * * * * * * * *

请输入你要排雷的坐标: 1 2

0 1 2 3 4 5 6 7 8 9

1 0 0 * * * * * *

2 * * * * * * * *

3 * * * * * * * *

4 * * * * * * * *

5 * * * * * * * *

6 * * * * * * * *

7 * * * * * * * *

8 * * * * * * * *

9 * * * * * * * *

CSDN @期邈云汉

```

8 * * * * *
9 * * * * *
请输入你要排雷的坐标: 1 3
0 1 2 3 4 5 6 7 8 9
1 0 0 0 * * * * *
2 * * * * *
3 * * * * *
4 * * * * *
5 * * * * *
6 * * * * *
7 * * * * *
8 * * * * *
9 * * * * *
请输入你要排雷的坐标: 1 4
0 1 2 3 4 5 6 7 8 9
1 0 0 0 0 * * * * *
2 * * * * *
3 * * * * *
4 * * * * *
5 * * * * *
6 * * * * *
7 * * * * *
8 * * * * *
9 * * * * *

```

CSDN @期邈云汉

```

请输入你要排雷的坐标: 1 5
0 1 2 3 4 5 6 7 8 9
1 0 0 0 0 1 * * * *
2 * * * * *
3 * * * * *
4 * * * * *
5 * * * * *
6 * * * * *
7 * * * * *
8 * * * * *
9 * * * * *
请输入你要排雷的坐标: 1 6
很遗憾, 点雷游戏失败! 0 1 2 3 4 5 6 7 8 9
1 0 0 0 0 0 1 0 0 0
2 0 0 0 0 0 0 0 0 0
3 0 0 0 1 1 0 0 0 0
4 0 0 0 0 0 0 0 0 0
5 0 0 0 1 1 0 0 0 0
6 0 0 1 0 0 0 0 0 0
7 1 1 0 0 0 0 0 0 0
8 0 0 0 1 0 0 0 0 0
9 0 0 0 0 1 0 0 0 0

```

```

-----
1. Play
0. Exit
-----

```

```

请输入你的选择: 0
退出游戏

```

D:\Microsoft Visual Studio\Code\C\Game2\x64\Debug\Game2.exe (进程 13236)已退出, 代码为 0。
按任意键关闭此窗口. . . _

CSDN @期邈云汉

