

源代码如下所示（具体思路和分析均已包含在注释中）：

输出截图如下所示：

#include <iostream>

#include <string>

using namespace std;

//行数和列数

int m,n;

//位置数组（自身+上下左右）

int f[5][2] = { {0,0} , {0,1} , {0,-1} , {1,0} , {-1,0} };

//结果数组（哨兵位置）

int res[30][30];

//哨兵个数

int ans;

//暂时存储哨兵位置

int sentinel[30][30];

//暂时存储哨兵个数

int p;

//哨兵能看到的位置

int vision[30][30];

//被监视的陈列室个数

int visions;

//函数声明

void Sentinel(int x,int y,int c,int d);

void search(int i,int j);

//主函数

int main()

{

    //读取input.txt

    FILE\* file1 = fopen ( "C:\\Users\\93508\\Desktop\\input.txt" , "r" );

    if (file1 == NULL){

        cout << "文件读取失败！" << endl;

        return 0;

    }

    fscanf ( file1 , "%d%d" , &m , &n );

    fclose ( file1 );

    //剪枝，哨兵个数不会超过(n\*m/3+1)个

    ans = n\*m/3 + 2;

    p = 0;

    //设置边界

    for(int i = 0 ; i <= m + 1 ; i++){

        vision[i][0] = vision[i][n+1] = 1;

    }

    for(int i = 0 ; i <= n + 1 ; i++){

        vision[0][i] = vision[m+1][i] = 1;

    }

    //开始搜索

    search(1,1);

    //输出结果并写入output.txt

    cout << ans << endl;

    FILE\* file2 = fopen ( "C:\\Users\\93508\\Desktop\\output.txt" , "w" );

    for(int i = 1; i <= m ; i++){

        for(int j = 1 ; j <= n ; j++){

            cout << res[i][j] << ' ';

            fprintf ( file2,"%2d" , res[i][j] );

        }

        cout << endl;

        fprintf (file2, "\n" );

    }

    fclose ( file2 );

}

//搜索位置

void search(int i,int j)

{

    if(p >= ans){

        return;

    }

    //该位置若已被某个哨兵观察到则跳过

    while(i <= m && vision[i][j]!=0 ) {

        j++;

        //换行

        if(j > n){

            i++;

            j = 1;

        }

    }

    //更新答案

    if(i>m) {

        ans = p;

        //更新后的答案复制到最终结果数组res中

        memcpy(res, sentinel, sizeof(sentinel));

        return;

    }

    if(i < m){

        Sentinel(i+1,j,i,j);

    }

    if(vision[i][j+1] == 0){

        Sentinel(i,j,i,j);

    }

    if(j < n && (vision[i][j+1] == 0 || vision[i][j+2] == 0)){

        Sentinel(i,j+1,i,j);

    }

}

void Sentinel(int x,int y,int c,int d)

{

    sentinel[x][y] = 1;

    p++;

    for(int i=0 ; i<=4; i++){

        int xx = x + f[i][0];

        int yy = y + f[i][1];

        vision[xx][yy]++;

        if(vision[xx][yy] == 1){

            visions++;

        }

    }

    search(c,d+1);

    sentinel[x][y] = 0;

    p--;

    for(int i=0 ; i<=4 ; i++){

        int xx = x + f[i][0];

        int yy = y + f[i][1];

        vision[xx][yy]--;

        if(vision[xx][yy]==0){

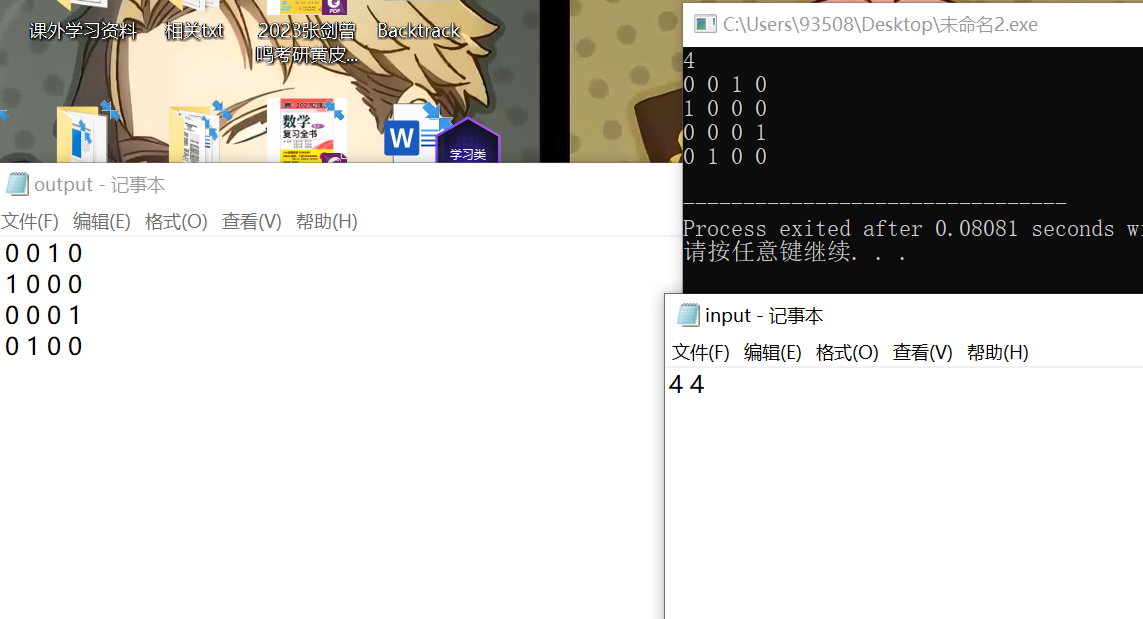
            visions--;

        }

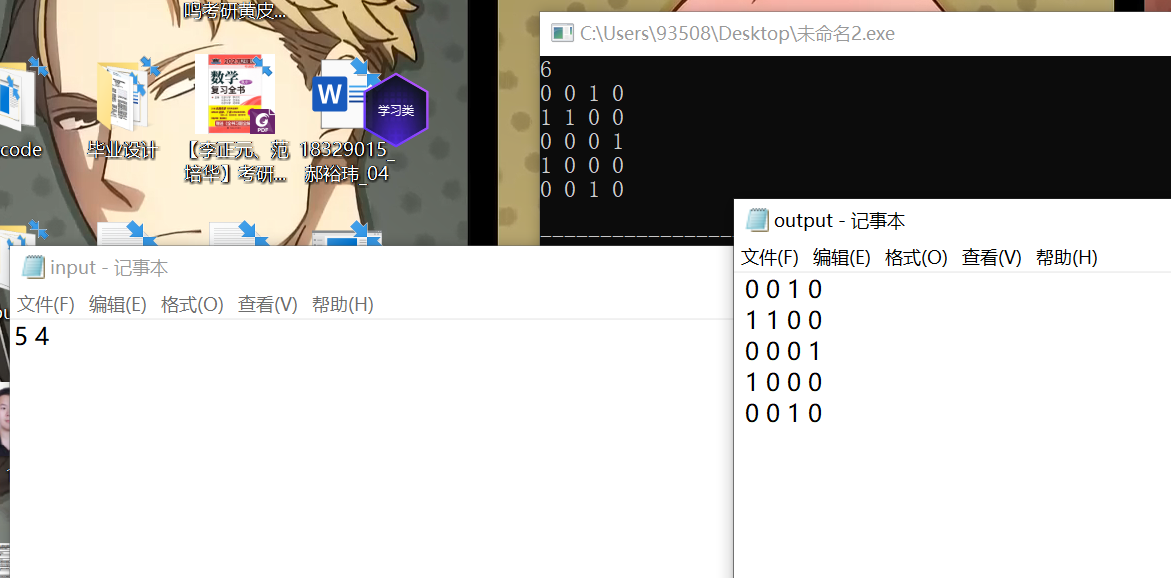
    }

}

样例1：



样例2：



样例3：

