

二维数组回形遍历

人员

韩承睿、刘嘉航、辛帅辰、李翰如、方俊喆、崔吉诺、刘祺、刘峰铄、齐振玮、牟茗、高健桓、谢亚锴、牛同泽、徐浩然 到课

秦显森 未到

作业

必做：东方博宜 1327. 鲜花方阵

选做：noi 1.8 22:神奇的幻方 和 noi 1.8 24:蛇形填充数组

下节课上课要求默写 noi 1.8 23:二维数组回形遍历，要求大家 15 分钟能默写出来

课堂表现

这节课题目相对比较难，老师讲的比较细致，大部分同学也都听得很认真

个别同学不认真听讲被老师批评了，希望以后改正

课堂内容

noi 1.8 23:二维数组回形遍历

```
int dx[] = {0, 1, 0, -1};  
int dy[] = {1, 0, -1, 0};
```

```
int x, y;  
int id; // 方向: 0 1 2 3
```

沿着id方向走: $x += dx[id]$, $y += dy[id]$;

转方向: $id = (id+1)\%4$;

如果下一步要走的位置上有数，不能走，应该转向

只要填够 $n*m$ 个数，就可以结束

```

#include <iostream>

using namespace std;

int w[105][105];
int dx[] = {0, 1, 0, -1}, dy[] = {1, 0, -1, 0};

int main() {
    for (int i = 0; i < 105; i++) {
        for (int j = 0; j < 105; j++) {
            w[i][j] = -1;
        }
    }

    int n, m; cin >> n >> m;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= m; j++) {
            cin >> w[i][j];
        }
    }

    int x = 1, y = 1; // 起点坐标
    int id = 0; // 方向
    int cnt = 0; // 计数器

    while (cnt < n*m) { // 当输出数量 < n*m 时, 要继续输出
        cout << w[x][y] << endl;
        w[x][y] = -1;
        cnt++;
        int nx = x+dx[id], ny = y+dy[id];
        if (w[nx][ny] == -1) {
            id = (id+1) % 4;
        }
        x += dx[id];
        y += dy[id];
    }
    return 0;
}

```