

孤独的素数

人员

温郝冬、李沛都、罗启宸、董浩桢、王奕皓、周苇杰、陶汇笙、康佳、马瑞昕、崔宸赫 到课

作业

必做作业：东方博宜 1997 - 孤独的素数 和 东方博宜 1203 - 拐角VIII

选做作业：东方博宜 1193 - 斜角II

下周要求默写：东方博宜 1997 - 孤独的素数 (要求在15分钟以内通过)

作业即使不通过，也要提交代码，在群里拍照打卡

课堂表现

同学们课堂表现整体不错，有几位同学这节课没有带纸和笔，下节课要求必须带纸和笔

董浩桢 同学课堂做题做的比较好，提出表扬！！

课堂默写情况

10 位同学都默写出来了，提出表扬！！！！

课堂内容

东方博宜 1997 - 孤独的素数

```

for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= m; j++) {
        // 如果a[i][j]是质数 -> b[i][j]=1; 否则 -> b[i][j]=0
        for (int k = 2; k <= a[i][j]-1; k++) {
            if (a[i][j]%k==0) {

            }
        }
    }
}

```

```

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

```

点 (i, j)

```

int cnt = 0;
for (int k = 0; k < 8; k++) {
    // 横坐标, 增加 dx[k]
    // 纵坐标, 增加 dy[k]
    int ni = i+dx[k], nj = j+dy[k];
    // 行 ni:1~n, 列 nj:1~m
    if (ni>=1 && ni<=n && nj>=1 && nj<=m) {
        if (a[ni][nj] == 1) {
            cnt++;
        }
    }
}
if (cnt == 0) {
    sum++;
}

```

```

#include <bits/stdc++.h>

using namespace std;

int a[55][55], b[55][55];
int dx[] = {-1, -1, -1, 0, 0, 1, 1, 1};
int dy[] = {-1, 0, 1, -1, 1, -1, 0, 1};

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

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= m; j++) {
            // 判断 a[i][j] 是不是质数
            // 从 2 ~ a[i][j]-1
            bool flag = true;
            for (int k = 2; k <= a[i][j]-1; k++) {
                if (a[i][j]%k == 0) {
                    flag = false;
                }
            }
            if (a[i][j] == 1) {
                flag = false;
            }
            if (flag == true) {
                b[i][j] = 1;
            }
            else {
                b[i][j] = 0;
            }
        }
    }

    int sum = 0;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= m; j++) {
            // 如果 b[i][j] 是孤独的素数, sum++
            if (b[i][j] == 1) {
                // 周围8个方向没有素数
                int cnt = 0;
                for (int k = 0; k < 8; k++) {
                    // 原先点坐标: (i,j)
                    // 横坐标变了 dx[k], 纵坐标变了 dy[k]

```

```

        int ni = i+dx[k], nj = j+dy[k];
        // (ni,nj) 在 1~n,1~m 之间
        if (ni>=1 && ni<=n && nj>=1 && nj<=m) {
            if (b[ni][nj] == 1) {
                cnt++;
            }
        }
    }
    if (cnt == 0) {
        sum++;
        cout << i << ", " << j << " --- " << b[i][j] << endl;
    }
}
}
cout << sum << endl;
return 0;
}

```

东方博宜 1188 - 数字走向V

```

#include <bits/stdc++.h>

using namespace std;

int a[15][15];

int main()
{
    int n;
    cin >> n;
    int k = 1;
    for (int i = n; i >= 1; i--) {
        for (int j = n; j >= 1; j--) {
            a[i][j] = k;
            k++;
        }
    }

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            printf("%3d", a[i][j]);
        }
        printf("\n");
    }
    return 0;
}

```

东方博宜 1203 - 拐角VIII

如果输入5的话，可以先构造

5 4 3 2 1

5 4 3 2 1

5 4 3 2 1

5 4 3 2 1

5 4 3 2 1

这样的二维数组。

然后，

把第1行小于1的变成1

把第2行小于2的变成2

...

把第n行小于n的变成n

此题不提供具体代码，要求同学们课下独立思考实现