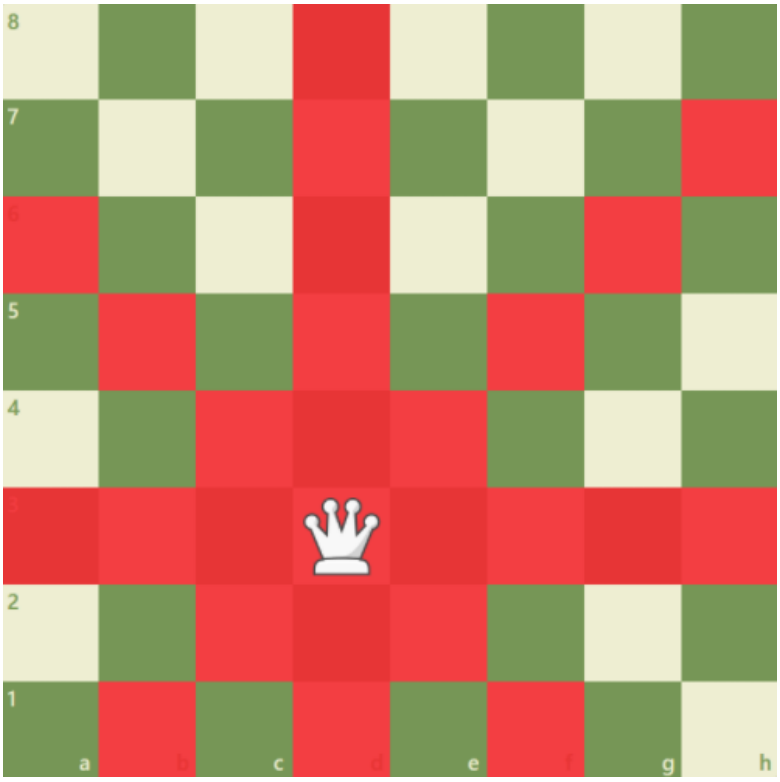
Homework #3

N-Queens problem

(Due: 2021-11-13)

Overview

The queen is one of the most powerful pieces in chess, which can move any number of squares *vertically*, *horizontally*, or *diagonally*. In theory, it is possible to place N queens on a N x N chessboard so that they will not attack each other. For this homework, you will be given a N x N chessboard with M (M<N) queens pre-placed on it. Please design a program to find all possible placement of the remaining N-M queens on the chessboard.



(a) A queen can move vertically, horizontally, or diagonally

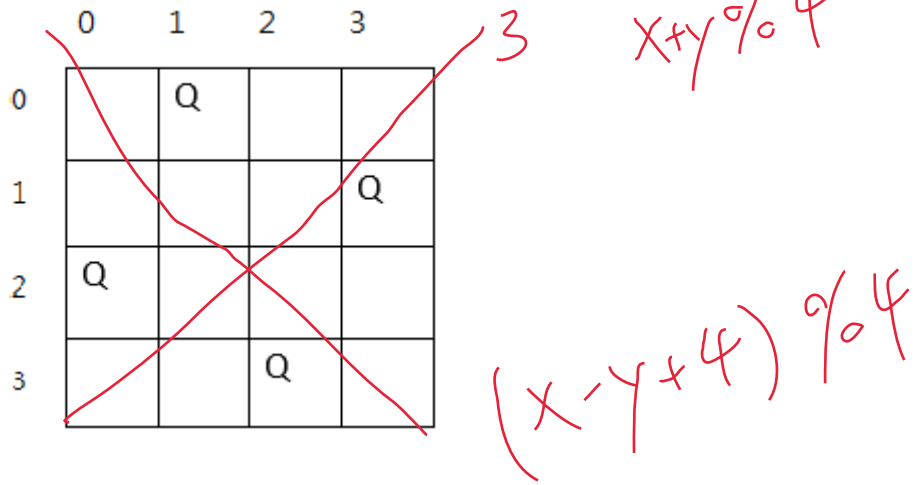
Input.txt

The first line of the input is the number of test cases **T**. Each test case begins with two integers **N, M,** which represent the chessboard size (N x N) and the M pre-placed queens on the board. Following are M pairs of integers separate by a single space, each of which indicates the row and column of a pre-placed queen.

Take the following input, for example.

1
4 2
0 1
1 3

The first integer indicates that **T** = 1. In the test case, the board size **N** = 4 and **M** = 2, the following two lines “0 1” and “1 3” means you must place two queens at [row 0, column 1] and [row 1, column 3]. The test case produces only one solution.



Output.txt

For each test case, output the number of possible placements for the remaining queens.

Sample Input

2
4 2
0 1
1 3
8 1
0 0

Sample Output

1
4

Constrains

1 <= T <= 20

1 <= m <= n <= 50

Preloaded Input Data

```
struct tTestCase {
	int n;
	int m;
	int preplace[50][2];
};

struct tTestData {
	int t;
	struct tTestCase testcase[100];
};
```