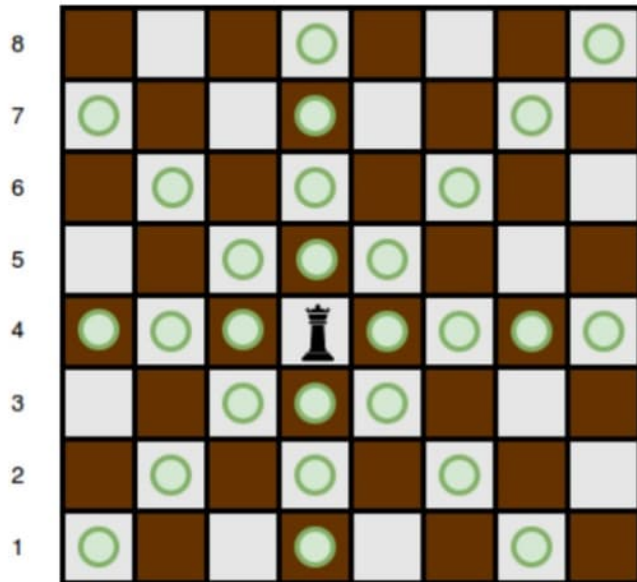


You will be given a square chess board with one queen and a number of obstacles placed on it. Determine how many squares the queen can attack.

A queen is standing on an $n \times n$ chessboard. The chess board's rows are numbered from 1 to n , going from bottom to top. Its columns are numbered from 1 to n , going from left to right. Each square is referenced by a tuple, (r, c) , describing the row, r , and column, c , where the square is located.

The queen is standing at position (r_q, c_q) . In a single move, she can attack any square in any of the eight directions (left, right, up, down, and the four diagonals). In the diagram below, the green circles denote all the cells the queen can attack from $(4, 4)$:



```

28
29     return attackable_squares;
30 }
31
32 int main() {
33     int n, k;
34     cin >> n >> k;
35     int r_q, c_q;
36     cin >> r_q >> c_q;
37
38     vector<vector<int>> obstacles(k, vector<int>(2));
39     for (int i = 0; i < k; i++) {
40         cin >> obstacles[i][0] >> obstacles[i][1];
41     }
42
43     cout << queensAttack(n, k, r_q, c_q, obstacles) << endl;
44     return 0;
45 }

```

Line: 43 Col: 30

Upload Code as File

☐ Test against custom input

Run Code

Submit Code



You have earned 30.00 points!

You are now 70 points away from the 2nd star for your problem solving badge.

0%

30/100

Congratulations

You solved this challenge. Would you like to challenge your friends?



Next Challenge

Test case 0

Compiler Message

Test case 1

Success

Test case 2

Input (stdin)

Test case 3

4 0

Download