```
#include <iostream>
#include <vector>

using namespace std;

bool isSafe(vector<vector<int>>& board, int row, int col, int N) {
    for (int i = 0; i < row; ++i) {
        if (board[i][col] == 1) {
            return false;
        }
    }

    for (int i = row, j = col; i >= 0 && j >= 0; --i, --j) {
        if (board[i][j] == 1) {
            return false;
        }
    }

    for (int i = row, j = col; i >= 0 && j < N; --i, ++j) {
        if (board[i][j] == 1) {
            return false;
        }
    }

    return true;
}

bool solveNQueens(vector<vector<int>>& board, int row, int N) {
    if (row == N) {
        cout << "Solution:" << endl;
        for (int i = 0; i < N; ++i) {
            for (int j = 0; j < N; ++j) {
                cout << board[i][j] << " ";
```

```
            }
            cout << endl;
        }
        return true;
    }

    bool foundSolution = false;
    for (int col = 0; col < N; ++col) {
        if (isSafe(board, row, col, N)) {
            board[row][col] = 1;

            foundSolution = solveNQueens(board, row + 1, N) ||
foundSolution;

            board[row][col] = 0;
        }
    }

    return foundSolution;
}

int main() {
    int N;
    cout << "Enter the size of the chessboard (N): ";
    cin >> N;

    vector<vector<int>> board(N, vector<int>(N, 0));

    if (!solveNQueens(board, 0, N)) {
        cout << "No solution exists." << endl;
    }

    return 0;
```

```
}
```

```
//OUTPUT:
Enter the size of the chessboard (N): 4
Solution:
0 1 0 0
0 0 0 1
1 0 0 0
0 0 1 0
```