

Program Code :-

#include <iostream>

using namespace std;

#define N 8

void printBoard(int board[N][N])

{

for (int i = 0; i < N; i++)

{

for (int j = 0; j < N; j++)

cout << board[i][j] << " ";

cout << endl;

}

}

bool isValid(int board[N][N], int row, int col)

{

for (int i = 0; i < col; i++)

if (board[row][i])

return false;

for (int i = row, j = col; i >= 0 && j >= 0; i--, j--)

if (board[i][j])

return false;

for (int i = row, j = col; j >= 0 && i < N; i++, j--)

if (board[i][j])

return false;

return true;

}

bool solveNQueen(int board[N][N], int col)

{

if (col >= N)

return true;

for (int i = 0; i < N; i++)

{

if (isValid(board, i, col))

{

board[i][col] = 1;

if (solveNQueen(board, col + 1))

return true;

board[i][col] = 0;

}

}

return false;

}

bool checkSolution()

{

int board[N][N];

for (int i = 0; i < N; i++)

for (int j = 0; j < N; j++)

board[i][j] = 0;

if (solveNQueen(board, 0) == false)

{

cout << "Solution does not exist";

return false;

}

printBoard(board);

return true;

}

int main()

{

checkSolution();

}

Program Output :-

Text

Description automatically generated