N-Queens

```c
#include <stdio.h>

#include <stdbool.h>


#define N 8


int board[N][N];


bool isSafe(int row, int col) {

    int i, j;

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

        if (board[row][i])

            return false;


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

        if (board[i][j])

            return false;


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

        if (board[i][j])

            return false;


    return true;

}


bool solveNQ(int col) {

    if (col >= N)
```

```c
        return true;

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

        if (isSafe(i, col)) {

            board[i][col] = 1;


            if (solveNQ(col + 1))

                return true;


            board[i][col] = 0;

        }

    }

    return false;

}


void printSolution() {

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

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

            printf("%d ", board[i][j]);

        printf("\n");

    }

}


int main() {

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

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

            board[i][j] = 0;
```
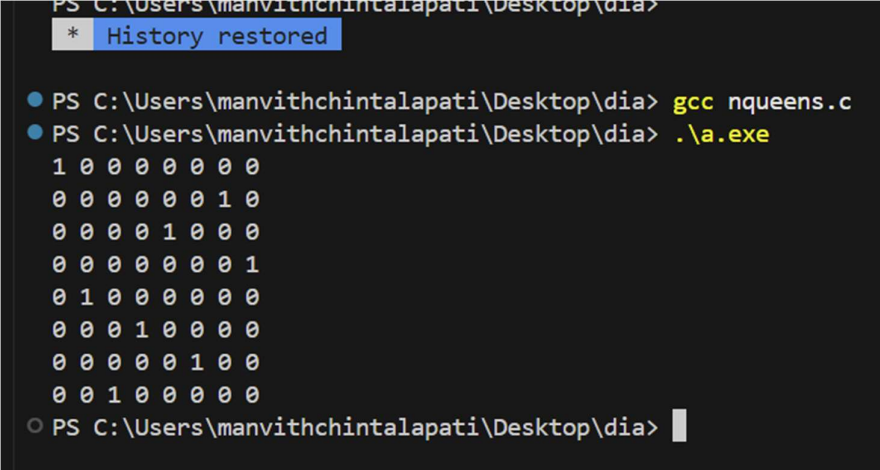
```c
    if (solveNQ(0) == false) {

        printf("Solution does not exist\n");

        return 0;

    }


    printSolution();

    return 0;

}
```

```
PS C:\Users\manvithchintalapati\Desktop\dia>
 *  History restored

● PS C:\Users\manvithchintalapati\Desktop\dia> gcc nqueens.c
● PS C:\Users\manvithchintalapati\Desktop\dia> .\a.exe
 1 0 0 0 0 0 0 0
 0 0 0 0 0 0 1 0
 0 0 0 0 1 0 0 0
 0 0 0 0 0 0 0 1
 0 1 0 0 0 0 0 0
 0 0 0 1 0 0 0 0
 0 0 0 0 0 1 0 0
 0 0 1 0 0 0 0 0
○ PS C:\Users\manvithchintalapati\Desktop\dia>
```