

FileEditSelectionViewGoRunTerminalHelp

n_queens.cpp - DAA - Visual Studio Code

n_queens.cpp X

n_queens.cpp > N

```
1  #include <bits/stdc++.h>
2
3  using namespace std;
4
5  #define N 4
6  static int countSol = 0;
7
8  void printBoard(char board[N][N])
9  {
10     for (int i = 0; i < N; ++i)
11     {
12         for (int j = 0; j < N; ++j)
13         {
14             cout << board[i][j] << " ";
15         }
16         cout << endl;
17     }
18 }
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

JUPYTER

Code

[Running]

cd "c:\Users\DELL\Desktop\DAA\" && g++ n_queens.cpp -o n_queens && "c:\Users\DELL\Desktop\DAA\"n_queens

- Q - -

- - - Q

Q - - -

- - Q -

[(1, 2) (2, 4) (3, 1) (4, 3)]

- - Q -

Q - - -

- - - Q

- Q - -

[(1, 3) (2, 1) (3, 4) (4, 2)]

Number of solutions for N-Queens board for N = 4 are: 2

[Done] exited with code=0 in 1.651 seconds

Ln 5, Col 12

Spaces: 4

UTF-8

CRLF

C++

Go Live

Win32

FileEditSelectionViewGoRunTerminalHelp

n_queens.cpp - DAA - Visual Studio Code

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

JUPYTER

Code

Running

cd "c:\Users\DELL\Desktop\DAA\" && g++ n_queens.cpp -o n_queens && "c:\Users\DELL\Desktop\DAA\"n_queens

Q - - - -

- - Q - -

- - - Q

- Q - - -

- - - Q -

[(1, 1) (2, 3) (3, 5) (4, 2) (5, 4)]

Q - - - -

- - - Q -

- Q - - -

- - - Q

- - Q - -

[(1, 1) (2, 4) (3, 2) (4, 5) (5, 3)]

- Q - - -

- - - Q -

Q - - - -

- - Q - -

- - - Q

[(1, 2) (2, 4) (3, 1) (4, 3) (5, 5)]

- Q - - -

- - - Q

- - Q - -

Q - - - -

- - - Q -

[(1, 2) (2, 5) (3, 3) (4, 1) (5, 4)]

- - Q - -

Q - - - -

- - - Q -

- Q - - -

- - - Q

[(1, 3) (2, 1) (3, 4) (4, 2) (5, 5)]

Ln 5, Col 12

Spaces: 4

UTF-8

CRLF

C++

Go Live

Win32

FileEditSelectionViewGoRunTerminalHelp

n_queens.cpp - DAA - Visual Studio Code

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

JUPYTER

Code

[(1, 3) (2, 1) (3, 4) (4, 2) (5, 5)]

- - Q - -

- - - - Q

- Q - - -

- - - Q -

Q - - - -

[(1, 3) (2, 5) (3, 2) (4, 4) (5, 1)]

- - - Q -

Q - - - -

- - Q - -

- - - - Q

- Q - - -

[(1, 4) (2, 1) (3, 3) (4, 5) (5, 2)]

- - - Q -

- Q - - -

- - - Q

- - Q - -

Q - - - -

[(1, 4) (2, 2) (3, 5) (4, 3) (5, 1)]

- - - - Q

- Q - - -

- - - Q -

Q - - - -

- - Q - -

[(1, 5) (2, 2) (3, 4) (4, 1) (5, 3)]

- - - - Q

- - Q - -

Q - - - -

- - - Q -

- Q - - -

Ln 5, Col 12

Spaces: 4

UTF-8

CRLF

C++

Go Live

Win32

FileEditSelectionViewGoRunTerminalHelp

n_queens.cpp - DAA - Visual Studio Code

Code

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

JUPYTER

[(1, 3) (2, 5) (3, 2) (4, 4) (5, 1)]

- - - Q -
Q - - - -
- - Q - -
- - - - Q
- Q - - -

[(1, 4) (2, 1) (3, 3) (4, 5) (5, 2)]

- - - Q -
- Q - - -
- - - - Q
- - Q - -
Q - - - -

[(1, 4) (2, 2) (3, 5) (4, 3) (5, 1)]

- - - - Q
- Q - - -
- - - Q -
Q - - - -
- - Q - -

[(1, 5) (2, 2) (3, 4) (4, 1) (5, 3)]

- - - - Q
- - Q - -
Q - - - -
- - - Q -
- Q - - -

[(1, 5) (2, 3) (3, 1) (4, 4) (5, 2)]

Number of solutions for N-Queens board for N = 5 are: 10

[Done] exited with code=0 in 1.453 seconds

Ln 5, Col 12

Spaces: 4

UTF-8

CRLF

C++

Go Live

Win32

File Edit Selection View Go Run Terminal Helpn_queens.cpp - DAA - Visual Studio Code

n_queens.cpp x

```
1  #include <bits/stdc++.h>
2
3  using namespace std;
4
5  #define N 5
6  static int countSol = 0;
7
8  void printBoard(char board[N][N])
9  {
10     for (int i = 0; i < N; ++i)
11     {
12         for (int j = 0; j < N; ++j)
13         {
14             cout << board[i][j] << " ";
15         }
16         cout << endl;
17     }
18
19     cout << "-----\n";
20     cout << "[";
21     for (int i = 0; i < N; ++i)
22     {
23         for (int j = 0; j < N; ++j)
24         {
25             if (board[i][j] == 'Q')
26             {
27                 cout << " (" << i + 1 << ", " << j + 1 << ") ";
28             }
29         }
30     }
31     cout << "]" << endl;
32     cout << "-----\n";
33 }
34
35 bool isSafe(char board[N][N], int row, int col)
36 {
37     for(int i = 0; i < row; ++i){
38         if(board[i][col] == 'Q')
39             return false;
40     }
```

Ln 5, Col 12 Spaces: 4 UTF-8 CRLF C++ Go Live Win32

File Edit Selection View Go Run Terminal Helpn_queens.cpp - DAA - Visual Studio Code

n_queens.cpp ×

n_queens.cpp > N

```
35 bool isSafe(char board[N][N], int row, int col)
36 {
37     for(int i = 0; i < row; ++i){
38         if(board[i][col] == 'Q')
39             return false;
40     }
41
42     for(int i = row, j = col; i >= 0 && j >= 0; --i, --j){
43         if(board[i][j] == 'Q')
44             return false;
45     }
46
47     for(int i = row, j = col; i >= 0 && j < N; --i, ++j){
48         if(board[i][j] == 'Q')
49             return 0;
50     }
51
52     return true;
53 }
54
55 void nqueens(char board[N][N], int row)
56 {
57     if (row == N)
58     {
59         printBoard(board);
60         countSol++;
61         return;
62     }
63
64     for (int col = 0; col < N; ++col)
65     {
66         if(isSafe(board, row, col)){
67             board[row][col] = 'Q';
68             nqueens(board, row + 1);
69             board[row][col] = '-';
70         }
71     }
72 }
73
```

Ln 5, Col 12 Spaces: 4 UTF-8 CRLF C++ Go Live Win32

File Edit Selection View Go Run Terminal Helpn_queens.cpp - DAA - Visual Studio Code

n_queens.cpp x

n_queens.cpp > N

```
53 }
54
55 void nqueens(char board[N][N], int row)
56 {
57     if (row == N)
58     {
59         printBoard(board);
60         countSol++;
61         return;
62     }
63
64     for (int col = 0; col < N; ++col)
65     {
66         if(isSafe(board, row, col)){
67             board[row][col] = 'Q';
68             nqueens(board, row + 1);
69             board[row][col] = '-';
70         }
71     }
72 }
73
74 int main()
75 {
76     char board[N][N];
77
78     memset(board, '-', sizeof(board));
79
80     nqueens(board, 0);
81
82     cout << "-----\n";
83     cout << "Number of solutions for N-Queens board for N = " << N << " are: " << countSol << endl;
84     cout << "-----\n";
85
86     return 0;
87 }
```

Ln 5, Col 12 Spaces: 4 UTF-8 CRLF C++ Go Live Win32