

Assignment No. 14

Name : Manish Namdev Barage

PRN : 2252007

Batch : T7

Topic : To implement N*N queen problem using backtracking

Code :

```
#include <iostream>
#include <vector>
using namespace std;

bool isSafe(int row, int col, vector<int>& positions) {
    for (int prevRow = 0; prevRow < row; ++prevRow) {
        int prevCol = positions[prevRow];
        if (prevCol == col || prevCol - col == prevRow - row ||
prevCol - col == row - prevRow) {
            return false;
        }
    }
    return true;
}

void solve(int row, vector<int>& positions, int n,
vector<vector<int>>& ans) {
    if (row == n) {
        ans.push_back(positions);
        return;
    }

    for (int col = 0; col < n; ++col) {
        if (isSafe(row, col, positions)) {
            positions[row] = col;
            solve(row + 1, positions, n, ans);
        }
    }
}

int main() {
    int n;
    cout << "Enter the value of n: ";
    cin >> n;
    vector<int> positions(n, -1);
    vector<vector<int>> ans;

    solve(0, positions, n, ans);
}
```

```

        cout << "Possible Solutions: " << ans.size() << endl;

        for (const auto& solution : ans) {
            for (int col : solution) {
                for (int i = 0; i < n; ++i) {
                    if (i == col) {
                        cout << "Q ";
                    } else {
                        cout << ". ";
                    }
                }
                cout << endl;
            }
            cout << endl;
        }

        return 0;
    }
}

```

Output :

```

PS D:\Third Year\DAA\LAB\Assign 10> cd "d:\Third Year\DAA\LAB\Assign 11\" ; if ($?) { g++ N_queen2.cpp -o N_queen2 } ; if ($?) { .\N_queen2 }
Enter the value of n: 4
Possible Solutions: 2
. Q . .
. . . Q
Q . . .
. . Q .

. . Q .
Q . . .
. . . Q
. Q . .

PS D:\Third Year\DAA\LAB\Assign 11>

```