Heaven's Light is Our Guide

Rajshahi University of Engineering and Technology

Department of Computer Science and Engineering

Course No: CSE.2202
Course Title: Sessional based on CSE.2201 (Computer Algorithms)

Lab Report No: 06
Lab Report On: N-Queens Problem using Backtracking.

Submitted By
Md. Ariful Islam
Roll No: 1803046
Section: A
Department: CSE

Submitted To
Biprodip Pal
Assistant Professor
Dept. of CSE,RUET

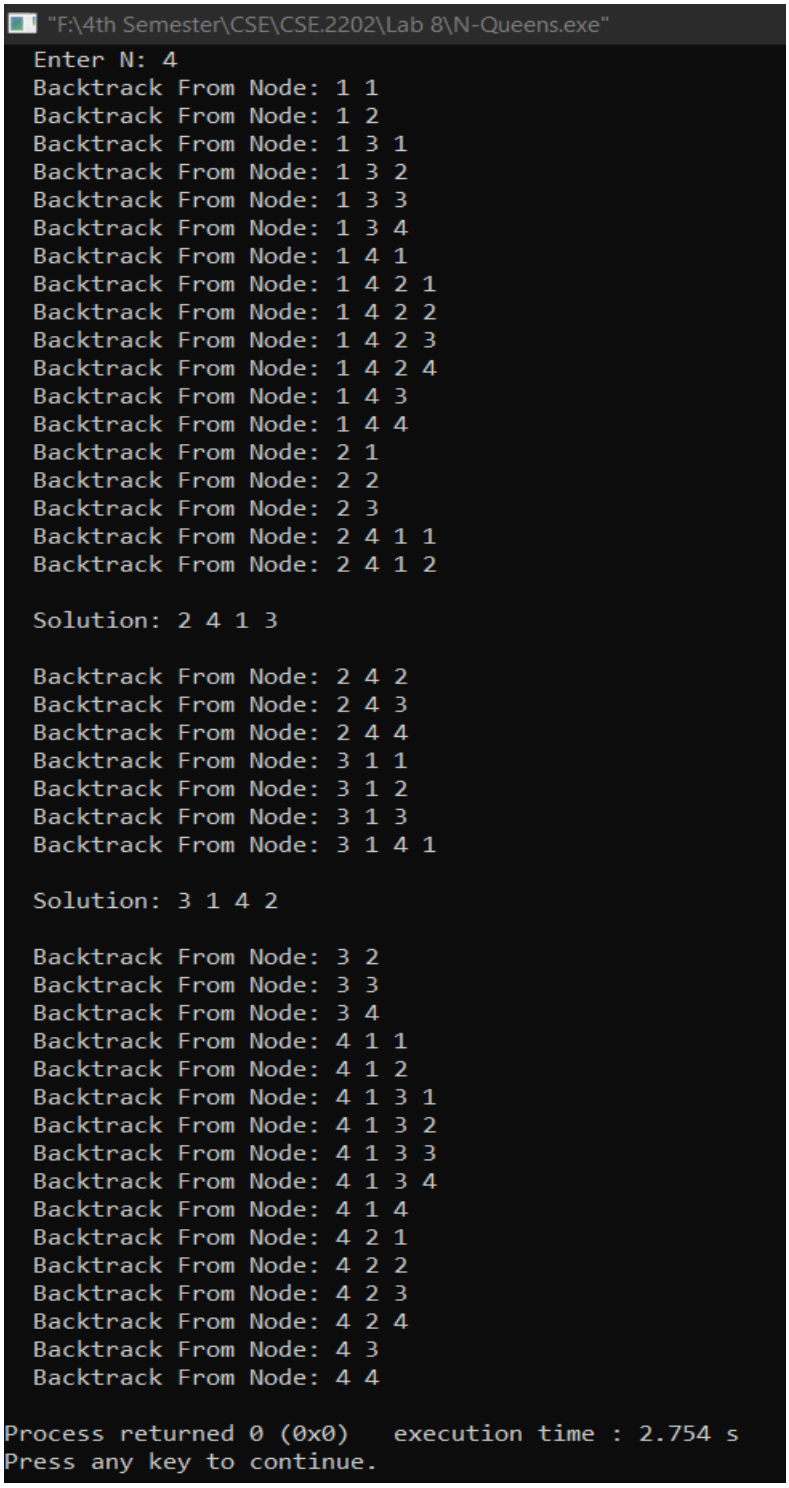
Date: 30-07-2021

❖ Source Code:

```
#include<bits/stdc++.h>
using namespace std;
typedef long long ll;

int arr[53]={0};
vector<int>xy;



// function to check the position

bool is_safe(int x,int y, int &n){
   bool rslt=true;
   int f=1;

   for(int k=x-1;k>0;k--){
      if(arr[k]==y){
         rslt=false;
         break;
      }
      if((x+y)==(arr[k]+k) && y<n){
        rslt=false;
        break;
      }
      if(y>1 && arr[k]==(y-f)){
         rslt=false;
         break;
      }
      f+=1;
   }

   return rslt;
}



// recaursive call for n-queen

void N_Q(int a, int &n){
   for(int i=1;i<=n;i++){
      if(is_safe(a,i,n)){
         arr[a]=i;

         if(a==n){
            cout<<"\nSolution: ";
```

```
                for(int j=1;j<=n;j++){
                    cout<<arr[j]<<" ";
                }
                cout<<"\n"<<endl;
                break;
            }
            else{
                N_Q(a+1,n);
            }
        }

        else{
            cout<<"Backtrack From Node: ";
            for(int j=1;j<a;j++){
                cout<<arr[j]<<" ";
            }
            cout<<i<<endl;
        }
    }
}



int main(){
    int n;

    cout<<"Enter N: ";
    cin>>n;
    if(n>0){
        N_Q(1,n);
    }

    return 0;
}
```

❖ Output:


"F:\4th Semester\CSE\CSE.2202\Lab 8\N-Queens.exe"
Enter N: 4
Backtrack From Node: 1 1
Backtrack From Node: 1 2
Backtrack From Node: 1 3 1
Backtrack From Node: 1 3 2
Backtrack From Node: 1 3 3
Backtrack From Node: 1 3 4
Backtrack From Node: 1 4 1
Backtrack From Node: 1 4 2 1
Backtrack From Node: 1 4 2 2
Backtrack From Node: 1 4 2 3
Backtrack From Node: 1 4 2 4
Backtrack From Node: 1 4 3
Backtrack From Node: 1 4 4
Backtrack From Node: 2 1
Backtrack From Node: 2 2
Backtrack From Node: 2 3
Backtrack From Node: 2 4 1 1
Backtrack From Node: 2 4 1 2
Solution: 2 4 1 3
Backtrack From Node: 2 4 2
Backtrack From Node: 2 4 3
Backtrack From Node: 2 4 4
Backtrack From Node: 3 1 1
Backtrack From Node: 3 1 2
Backtrack From Node: 3 1 3
Backtrack From Node: 3 1 4 1
Solution: 3 1 4 2
Backtrack From Node: 3 2
Backtrack From Node: 3 3
Backtrack From Node: 3 4
Backtrack From Node: 4 1 1
Backtrack From Node: 4 1 2
Backtrack From Node: 4 1 3 1
Backtrack From Node: 4 1 3 2
Backtrack From Node: 4 1 3 3
Backtrack From Node: 4 1 3 4
Backtrack From Node: 4 1 4
Backtrack From Node: 4 2 1
Backtrack From Node: 4 2 2
Backtrack From Node: 4 2 3
Backtrack From Node: 4 2 4
Backtrack From Node: 4 3
Backtrack From Node: 4 4
Process returned 0 (0x0) execution time : 2.754 s
Press any key to continue.

 # END #