

Logo

N-Queen

성공 풀이

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
10 초	128 MB	5629	3061	2175	55.077%

문제

N-Queen 문제는 크기가 $N \times N$ 인 체스판 위에 퀸 N개를 서로 공격할 수 없게 놓는 문제이다.
N이 주어졌을 때, 퀸을 놓는 방법의 수를 구하는 프로그램을 작성하시오.

Back tracking 설명할 때 나오는 기본 문제 이다.
개념 이해 해놓고, queen의 위치를 구현하는데서 시간을 다 잡아 먹었다.

나에게 부족하고 가장 필요한 것은 구현 능력이다. 이능력을 기르려면, 꾸준히 문제를 풀어야 한다. 매일 매일 문제를 풀자!

코드

```
#include<iostream>
#include<stdlib.h>
using namespace std;
int ans;
int **map;
int n;
void getInput(){
    cin>>n;
    map = new int*[n];
    for(int i =0; i<n; i++){
        int * tmp = new int[n];
        for(int j =0; j< n; j++){
            tmp[j]=0;
        }
        map[i] = tmp;
    }
}
void test(){
    for(int i =0; i< n; i++){
        for(int j =0; j< n; j++){
            cout<<map[i][j]<<" ";
        } cout<<endl;
    }
    cout<<endl;
}
```

```

bool canPos(int level, int column){
    int c = column;
    int l = level;
    int i =l, j = c;

    for(int i =0; i<n ;i++)
        if(map[i][c])
            return false;

    while(i>=0 && j>=0){
        if (map[i][j])
            return false;
        i--; j--;
    }
    i = l, j =c;
    while(i>=0 && j<n) {
        if (map[i][j])
            return false;
        i--; j++;
    }

    return true;
}

void func (int level){
    test();
    if (level == n) return;
    int l = level;
    for(int i =0; i<n; i++){
        if(canPos(level,i)){
            map[l][i] = 1;
            if(level == n-1)ans++;
        }else continue;
        func(l+1);
        map[l][i] = 0;
    }
    return ;
}

int main(){
    getInput();
    func(0);
    cout<<ans<<endl;

    return 0;
}

```

입력

첫째 줄에 N이 주어진다. ($1 \leq N < 15$)

출력

첫째 줄에 퀸 N개를 서로 공격할 수 없게 놓는 경우의 수를 출력한다.

예제 입력

8

예제 출력

92

힌트

출처

- 문제를 만든 사람: baekjoon

알고리즘 분류

- 백트래킹