

Experiment: - 9

Student Name: Rajdeep Jaiswal

UID: 20BCS2761

Branch: BE (CSE)

Section/Group: 902 B WM

Semester: 5th

Subject Name: Competitive Coding (20CSP-314)

1. N-Queens

Program Code:

```
#include<iostream> using namespace std; bool
issafe(int**arr, int x, int y, int n) { for(int
row=0;row<x;row++) { if(arr[row][y]==1) {
return 0;
} } int row = x; int
col = y;
while(row>=0&&col>=0) {
if(arr[row][col]==1) {
return 0;
} row--; col--; }
row = x; col = y;
while(row>=0&&col<n)
{ if(arr[row][col]==1)
{ return 0;
} row--;
col++; }
return 1;
} bool nqueen(int**arr, int x, int n)
{ if(x>=n) { return 1;
} for(int
col=0;col<n;col++) {
if (issafe(arr,x,col,n)) {
arr[x][col] = 1;
if(nqueen(arr, x+1,n)) {
return 1; }
arr[x][col] = 0; }
} return 0;
} int main(void) { int n;
cin>>n; int** arr = new
int*[n];
for(int i=0;i<n;i++) {
```

```
arr[i] = new int[n];
for(int
j=0;j<n;j++)          {
arr[i][j]=0;        } }
if(nqueen(arr,0,n)) {
for(int i=0;i<n;i++) {
for(int j=0;j<n;j++) {
cout<<arr[i][j]<<" ";      }
cout<<endl;      }
} else {
cout<<"Not
possible"<<endl;
}
}
```

Output:

Submission ID: 75872655 / 5 seconds ago

RESULT: ✔ Accepted 🔗 Refer judge environment

Score	Time (sec)	Memory (KiB)	Language
20	0.0939	2	C++

Input	Result	Time (sec)	Memory (KiB)	Score	Your Output	Correct Output	Diff
Input #1	✔ Accepted	0.008654	2	10			
Input #2	✔ Accepted	0.009626	2	10			
Input #3	✔ Accepted	0.009414	2	10			
Input #4	✔ Accepted	0.009045	2	10			
Input #5	✔ Accepted	0.009515	2	10			
Input #6	✔ Accepted	0.009723	2	10			
Input #7	✔ Accepted	0.009376	2	10			
Input #8	✔ Accepted	0.009589	2	10			
Input #9	✔ Accepted	0.009836	2	10			
Input #10	✔ Accepted	0.009121	2	10			

2. Queens on Board

Program Code:

```
#include<stdio.h>
```

```

#include<string.h> #define
MOD 1000000007 int n,m,bit[1
<< 10] ; char g[52][52] ;

int memo2[1 << 15] ; int
spread(int mask)
{
    if(memo2[mask] != -1) return memo2[mask] ;
    int nmask = 0 ;
    for(int i = 0;i < m;i++)
    {
        if(mask & 1 << 3 * i) if(i > 0) nmask |= 1 << 3 * i - 3 ;    if(mask
& 1 << 3 * i + 1) nmask |= 1 << 3 * i + 1 ;    if(mask & 1
<< 3 * i + 2) if(i + 1 < m) nmask |= 1 << 3 * i + 5 ;
    }
    return memo2[mask] = nmask ;
}
int good[50][1 << 8],szg[50],block[50] ; int
memo[50][1 << 15] ; int solve(int
x,int mask)
{
    if(x == n) return 1 ; mask &= ~block[x] ;
    if(memo[x][mask] != -1) return memo[x][mask] ;

    int ret = 0 ; for(int i = 0;i < szg[x];i++)
    if(!(good[x][i] & mask))
    {
        int cret = solve(x + 1,spread(good[x][i] | mask)) ;
        ret += cret ;    if(ret >= MOD) ret -= MOD ;
    }
    return memo[x][mask] = ret ;
} int solve() { for(int
i = 0;i < n;i++)
{    block[i] = 0 ;    int cmask = 0 ;    for(int j
= 0;j < m;j++) if(g[i][j] == '#')
    {
        cmask |= 1 << j ;    block[i]
|= 7 << 3 * j ;    }    szg[i] =
0 ;    for(int j = 0;j < 1 <<
m;j++) if((j & cmask) == 0)
    {
        bool valid = true ;    for(int k = 0;k < m;k++) if(j
& 1 << k)    for(int kk = k + 1;kk < m && g[i][kk] !=
'#';kk++)    if(j & 1 << kk)    valid

```

```
= false ;    if(!valid) continue ;
    int sp = 0 ;    for(int k = 0;k < m;k++) if(j & 1 <<
k) sp |= 7 << 3 * k ;    good[i][szg[i]] = sp ;
szg[i]++ ;
    }
}
memset(memo,255,sizeof memo) ;
memset(memo2,255,sizeof memo2) ; int
ret = solve(0,0) ; return
ret ; }
int main(void)
{
for(int i = 1;i < 1 << 10;i++) bit[i] = bit[i >> 1] + (i & 1) ; int
runs ; scanf("%d",&runs) ; while(runs--)
{
scanf("%d%d",&n,&m) ; for(int i = 0;i <
n;i++) scanf("%s",g[i]) ; int ret = solve()
;
ret = (ret - 1 + MOD) % MOD ; printf("%d\n",ret)
;
}
}
```


Output:


Congratulations


You solved this challenge. Would you like to challenge your friends? [f](#) [t](#) [in](#)


[Next Challenge](#)


Test case 0


Test case 1 

Test case 2 

Test case 3 

Test case 4 

Test case 5 

Test case 6 

Compiler Message

Success

Input (stdin) [Download](#)

1	4
2	3 3
3	...
4	...
5	...
6	3 3
7	.#. .
8	.#. .
9	...