

DESIGN AND ANALYSIS OF ALGORITHMS – 2CS503

Practical 10

Name: Bhanderi Mihir

Roll No.: 19BCE023

Batch No.: A-1

1. N-Queens Problem

Code:

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

```
#define n 5
```

```
int safe(int board[n][n], int row, int col){
```

```
    for (int i = 0; i < col; i++)
```

```
        if (board[row][i]==1)
```

```
            return 0;
```

```
    for (int i = row, j = col; i >= 0 && j >= 0; i--, j--)
```

```
        if (board[i][j]==1)
```

```
            return 0;
```

```
    for (int i = row, j = col; j >= 0 && i < n; i++, j--)
```

if (board[i][j]==1)
return 0;
return 1;
}
int solve(int board[n][n], int col){
if(col >= n)
return 1;
for(int i=0;i<n;i++){
if(safe(board, i, col)){
board[i][col]=1;
if(solve(board, col+1))
return 1;
board[i][col]=0;
}
}
return 0;
}
int main(){

```
int board[n][n];
```

```
for(int i=0;i<n;i++){
```

```
    for(int j=0;j<n;j++){
```

```
        board[i][j]=0;
```

```
    }
```

```
}
```

```
if(solve(board, 0) == 0){
```

```
    printf("Solution not exist.");
```

```
}
```

```
else{
```

```
    for(int i=0;i<n;i++){
```

```
        for (int j = 0; j < n; j++)
```

```
        {
```

```
            printf("%d ",board[i][j]);
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
}
```

```
}
```

```
/*
```

Output:
1 0 0 0 0
0 0 0 1 0
0 1 0 0 0
0 0 0 0 1
0 0 1 0 0
* /