+ve base case
sabse upar lgte hai

Page No.:
Date: / /

chess board

combination in 2D (how to place queen in 2D)

(1)
```
comb2D(n, r, c, QTP, path)
{
    if(QTP==0) SOPln(path) return
    if(r==n) return
    if(c==n) r++; c=0;
    comb2D(n, r, c+1, QTP-1, path+"Q{"+r+","+c+"}");
    comb2D(n, r, c+1, QTP, path);
}
```

- But isse queen kaise place krenge

(2) queen place krne k liye boolean array bnana pdega

```
print(boolean[][] board)
{
    for(boolean[] row : board)
    {
        for(boolean b : row)
        {
            if(b) SOP("/")
            else  SOP(".")
        }
        SOPln();
    }
}
```

```
comb(n, r, c, QTP, path)
{
    if(QTP==0) SOPln(path) SOPln("=====") return;
    if(r==n) return
    if(c==n) r++; c=0;
    board[r][c]=true
    comb(n, r, c+1, QTP-1, path+"Q{"+r+
        ","+c+"}")
    board[r][c] = false;
    comb(n, r, c+1, QTP, path);
}
```

Page No.:
Date: / /

But yaar isme to abhi bahut problems hai queens ko kaise pta chalega unhe kha baithna h

jo jagah safe hai baithegi queen

> to chaliye dekhte hai kaise

```
comb(n, r, c, QTP, path, board)
{
    if(QTP == 0)
    {
        SOP(path)
        print(board)
        SOPln("=====")
        return
    }
    if(c==n) r++ c=0
    if(r==n) return
    if(isSafe(board, r, c))
    {
        board[r][c] = true
        comb(n, r, c+1, QTP-1, path+"Q{" + r +
                ", " + c + "}", board)
        board[r][c] = false;
    }

    comb(n, r, c+1, QTP, path, board)
}
```

```
isSafe(board, r, c)
{
    for (row=0 ; row<r; row++)
    {
        if (board[row][c]) return false
    }

    for ( col=0; col<c ; c++)
    {
        if (board[r][col] return false
    }

    r1 = r
    c1 = c

    while (r1>=0 && c1>=0)
    {
        if (board[r1][c1]) return false
        r1--
        c1--
    }

    r2=0
    c2=c

    while (r2>=0 && c2<board[0].length)
    {
        if (board[r2][c2]) return false
        r2--
        c2++
    }

    return true;
}
```

Maze path : #blocked

① 0 1 0 0
0 0 0 0
0 1 0 0
0 0 0 0

```
Solve (r, c, maze path)
{
    if (r = maze.length-1 && c == maze[0].length)
        Soln.(path) return;
    if (r<0 || r == maze.length || c<0 || c == maze[0].length
                || maze[r][c] == 1)
        return;
    solve (r-1, c, maze, path + "U")
    solve (r+1, c, maze, path + "D")
    solve (r, c+1, maze, path + "R")
    solve (r, c-1, maze, path + "L")
}
```

ye jo last condition hai na, vo always last me hi hoga. cuz agar pehle ek bhi diya to index out of bound ho jaega i.e. (r<0 & c<0) wala condition bina check kiye hi ye check kr dega to (-1) wla index bhi check kr dega. & ans index out of bound ho jaega.

But But But iss method se stack overflow ho jaega kyu???

kyuki ye bnda wps U me fas jaega

Page No.:
Date: / /

jab next jaega tab 'U' wala recursion chalega aur ye wps upar aa jaega phir neeche jaega phir upar aaega

thus Stack Overflow (rajiya gundo me fass gyi h!)

issliye ab boolean array banaenge.

(2) visited[][] = new boolean[n][n]

```
solve (r,c, maze, path, visited)
{
    if(r==maze.length-1 && c==maze[0].length-1)
        soPln(path) return
    if(r<0 || r==maze.length || c<0 ||
       c==maze[0].length || maze(r)(c)==1 ||
       visited(r)(c))
        return
    visited[r][c] = true
    solve(r-1, c, maze, path + "U", visited)
    solve(r+1, c, maze, path + "D", visited)
    solve(r, c+1, maze, path + "R", visited)
    solve(r, c-1, maze, path + "L", visited)
}
```

isse all path explore nhi hoga. Sirf ek path hoga

kyu ?

kyuki jo visited hua vo true ho gyi vha wps nhi ja skte

(++) = implicit typecast

Page No.:
Date: / /

ab kya kre?

Shirt pehno aur utaro wala trick.

Last me add (visited(r)(c) = false) after saara 'solve' wala statement

Sodukus Solver

~~options~~ ways to solve soduku = 9^{81} ways

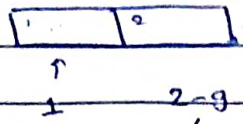
```
① Solve (~~board~~, r, c, board[][])
{
    if (c==9)   r++   c=0
    if (r==9)   print(board) sopen("====>") return;
    if (board[r][c] != '.')   Solve(r, c+1, board)
    else
    {
        for (char ch = '1'; ch <= 9; ch++)
        {
            board[r][c] = ch
            Solve(r, c+1, board)
        }
        //board[r][c] = '.',
    }
}

print(char board[][])
{
    for (char[] row : board)
    {
        for (char ch : row)  SOP(ch + " ")
        sopln()
    }
}
```

Page No.:
Date: / /

isse to solve ho jaana chahiye
But But issme problems hai

P.① maze path jaisa kam path explore kr paayega

i.e. 

jb wps aaya tb pehle dabba me dot nhi hai

thus isse $2 \times 9 = 18$ options explore krega

jbki usko 9^2 options explore krna

chahiye tha

matlab vhi shirt pehn k utara nhi

(ye tabhi hota hai jb loop use hota)

② add (board[r][c] = '.') in code jo commented ha

ab kya hoga

solve hoga

but but 10^{70} sec lega complete hone me

thus infinite loop

③ agr hm condition lga de ki recursion tabhi chale jb jaruri ho. Tb to time kam lega

```
else
{
    for(ch='1'; ch<='9'; ch++)
    {
        if(isSafe(board, r, c, ch))
        {
            board[r][c] = ch
            solve(r, c+1, board)
        }
    }
    board[r][c] = '.'
}
```

Page No.:
Date : / /

```
isSafe(board, r, c, ch)
{
    for(row=0; row<9; row++)
    {
        if(board[row][c]==ch) return false
    }
    for(col=0; col<9; col++)
    {
        if(board[r][col]==ch) return false
    }
    int box_row = r/3   box_col = c/3
    for(row=box_row*3; row<box_row*3+3; row++)
    {
        for(col=box_col*3; col<box_col*3+3; col++)
        {
            if(board[row][col]==ch) return false
        }
    }
}
```

me same nhi hona chahiye

me same nhi hona chahiye

me same nhi hona chahiye

ab to ho hi jaana chahiye.

Bhai

print krne p khaali ho ja rha hai

ye bol rha hai ki maine sudoku solve kr liye pr mai abi btaunga.

(4) ek boolean bna lete hai tb try krte hai

static boolean ans = false (global variable)