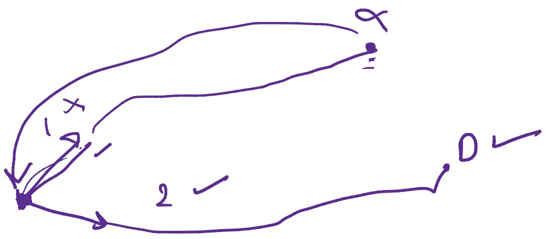
3-Jan-2022

03 January 2022 21:00

Backtracking :-



Rat in a Maze :-

```
     0   1   2   3
0  {{1, 0, 0, 0},
1   {1, 1, 0, 1},
2   {1, 1, 0, 0},
3   {0, 1, 1, 1}}
```

1 : Rat can visit it.

0 : Rat can't visit it.

left ← ↑ up → Right ↓ down

- A cell can't be visited more than once

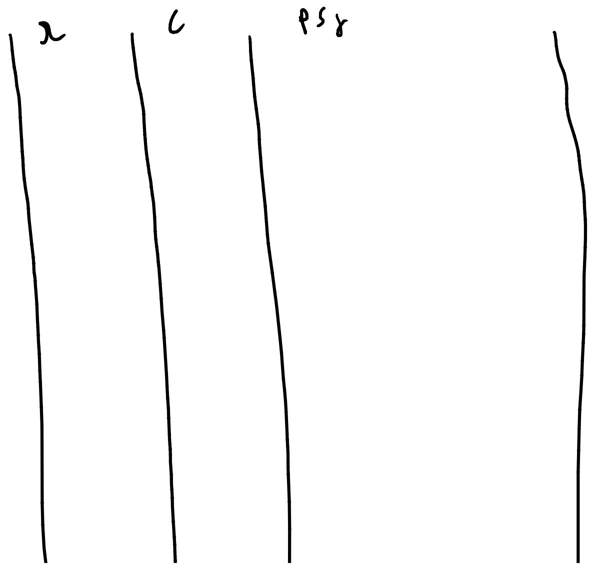
D D R D R R

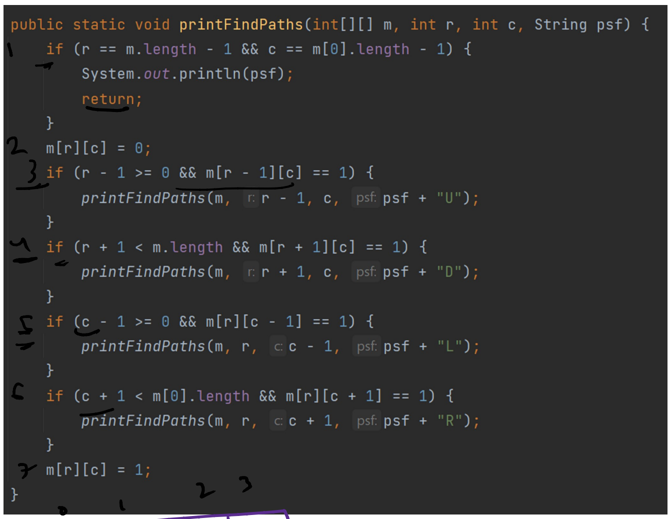
D R D D R R

U D L R

Algo :-

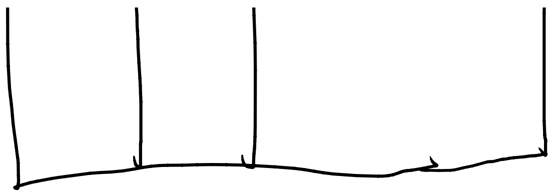
- Mark it, when you reach it.
- Try 4 possible directions
- Set it to 1 when all directions gets traversed.





```java
public static void printFindPaths(int[][] m, int r, int c, String psf) {
    if (r == m.length - 1 && c == m[0].length - 1) {
        System.out.println(psf);
        return;
    }
    m[r][c] = 0;
    if (r - 1 >= 0 && m[r - 1][c] == 1) {
        printFindPaths(m, r - 1, c, psf + "U");
    }
    if (r + 1 < m.length && m[r + 1][c] == 1) {
        printFindPaths(m, r + 1, c, psf + "D");
    }
    if (c - 1 >= 0 && m[r][c - 1] == 1) {
        printFindPaths(m, r, c - 1, psf + "L");
    }
    if (c + 1 < m[0].length && m[r][c + 1] == 1) {
        printFindPaths(m, r, c + 1, psf + "R");
    }
    m[r][c] = 1;
}
```

D D R D R R



$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 2 & 1 & 1 & 0 & 0 \\ 3 & 0 & 1 & 1 & 1 \end{array}$$

DDRDRR

DRDDRR

Solving it with directions array :-

$$\begin{tikzcd} & 0,1 & \\ 1,0 & 1,1 \arrow[l, "{[0,-1]}"'] \arrow[u, "{[-1,0]}"'] \arrow[r, "{[0,1]}"'] \arrow[d, "{[1,0]}"] & 1,2 \\ & 2,1 & \end{tikzcd}$$

$dr = -1$
$dc = 0$
$r = 1$
$c = 1$

$nr = dr + r = 0$
$nc = dc + c = 1$

3 , 3

dir = [[-1, 0], [0, 1], [1, 0], [0, -1]]

dirChar: ['U' , 'R' , 'D' , 'L']

3