

Backtracking | Set 2 (Rat in a Maze) - GeeksforGeeks

We have discussed Backtracking and Knight's tour problem in [Set 1](#). Let us discuss Rat in a [Maze](#) as another example problem that can be solved using Backtracking.

A Maze is given as $N \times N$ binary matrix of blocks where source block is the upper left most block i.e., `maze[0][0]` and destination block is lower rightmost block i.e., `maze[N-1][N-1]`. A rat starts from source and has to reach destination. The rat can move only in two directions: forward and down.

In the maze matrix, 0 means the block is dead end and 1 means the block can be used in the path from source to destination. Note that this is a simple version of the typical Maze problem. For example, a more complex version can be that the rat can move in 4 directions and a more complex version can be with limited number of moves.

Following is an example maze.

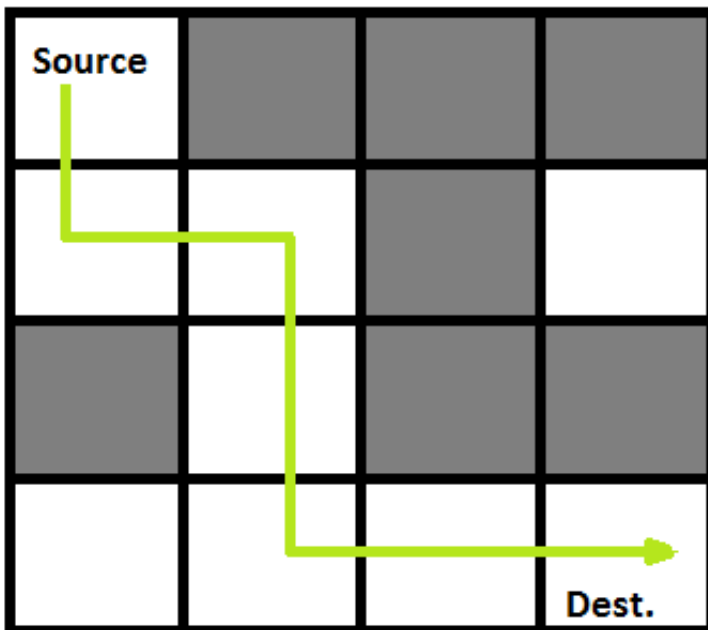
Gray blocks are dead ends (value = 0).

Source			
			Dest.

Following is binary matrix representation of the above maze.

```
{1, 0, 0, 0}  
{1, 1, 0, 1}  
{0, 1, 0, 0}  
{1, 1, 1, 1}
```

Following is maze with highlighted solution path.



Following is the solution matrix (output of program) for the above input matrix.

```
{1, 0, 0, 0}
```

```
{1, 1, 0, 0}
```

```
{0, 1, 0, 0}
```

```
{0, 1, 1, 1}
```

All enteries in solution path are marked as 1.

Naive Algorithm

The Naive Algorithm is to generate all paths from source to destination and one by one check if the generated path satisfies the constraints.

```
while there are untried paths
{
    generate the next path
    if this path has all blocks as 1
    {
        print this path;
    }
}
```

Backtrackng Algorithm

```
If destination is reached
    print the solution matrix
Else
    a) Mark current cell in solution matrix as 1.
    b) Move forward in horizontal direction and recursively check if this
        move leads to a solution.
    c) If the move chosen in the above step doesn't lead to a solution
```

then move down and check if this move leads to a solution.
d) If none of the above solutions work then unmark this cell as 0 (BACKTRACK) and return false.

Implementation of Backtracking solution