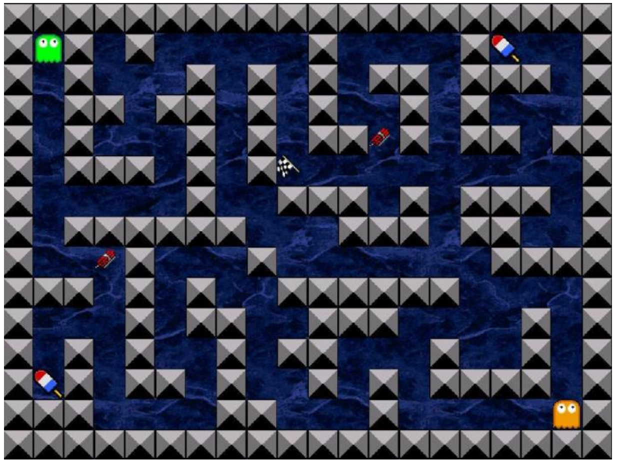
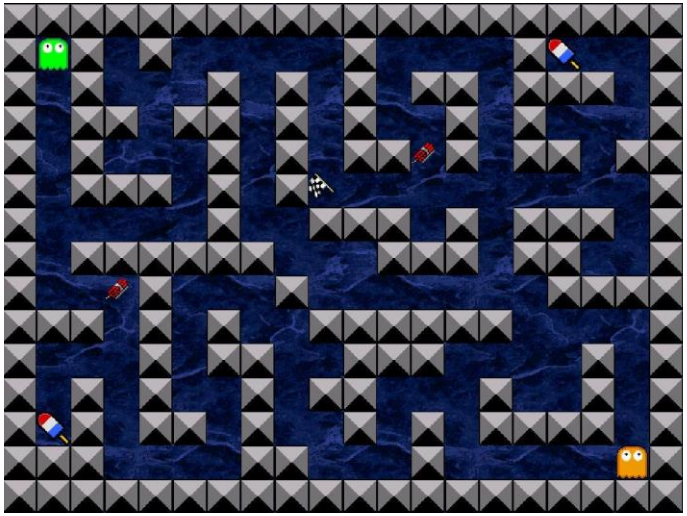
Maze Generator

http://acm.cs.nthu.edu.tw/problem/10626



Maze Generator

- Suppose we are developing computer games
  - We very probably need a maze generating function or program
- Have you ever thought about how computers can generate mazes?
- This assignment introduces the spanning tree method



Our Goal

A 2D array in memory

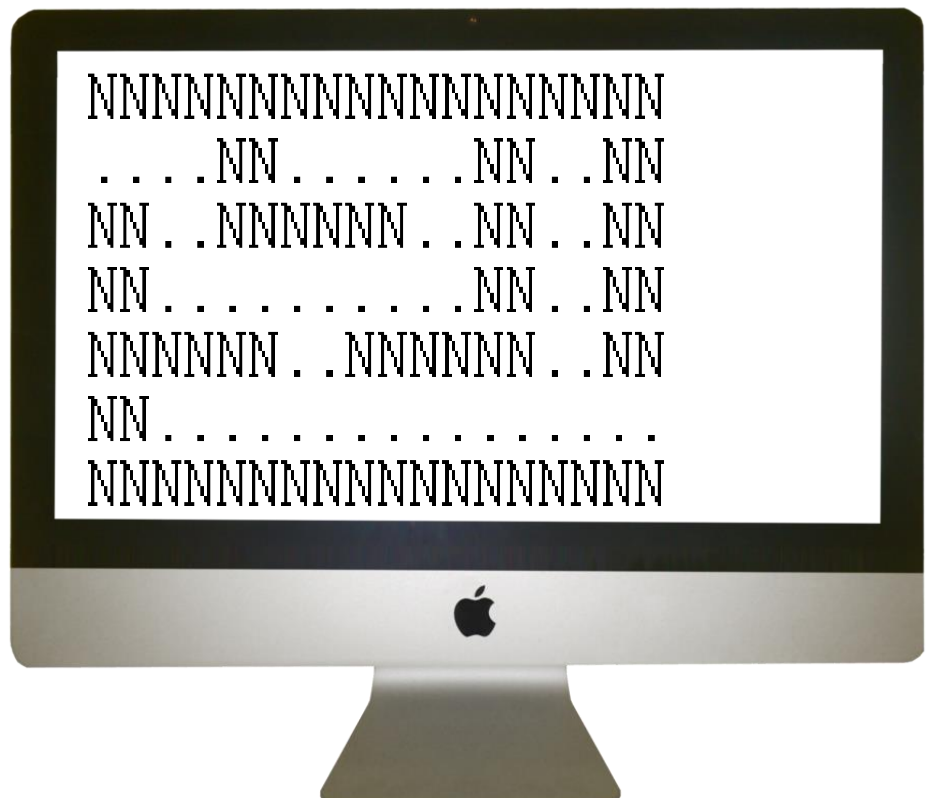
Columns

Rows

1	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	0	1
1	0	1	1	1	0	1	0	1
1	0	0	0	0	0	1	0	1
1	1	1	0	1	1	1	0	1
1	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1

1's represent walls;
0's represent paths.

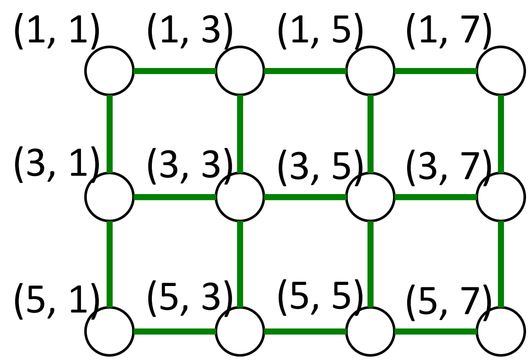
Standard Output



Technique for Generating a Maze

- The 2D array is initially set to all 1's
- A grid graph are associated with the array

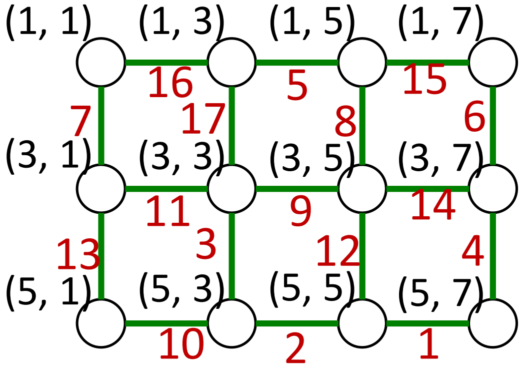
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1



Technique for Generating a Maze

- Each edge has a randomized cost as shown in the red text below

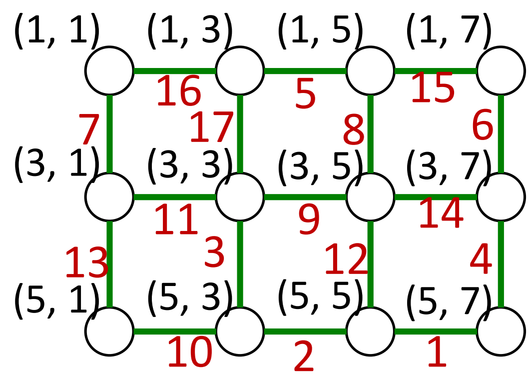
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1



Technique for Generating a Maze

- First, we let the spanning tree root at (1, 1)
- We set (1, 1) as 0

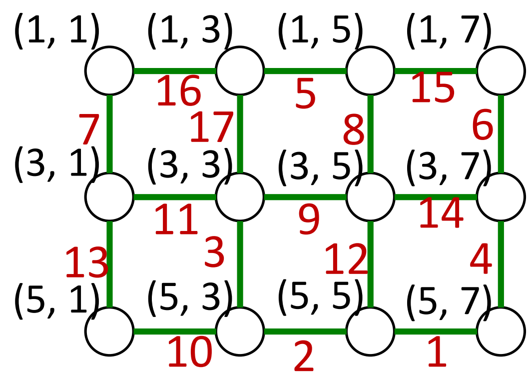
1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1



Technique for Generating a Maze

- Then we scan all the edges to find the least-cost edge with exactly one vertex corresponding to 0 in the array

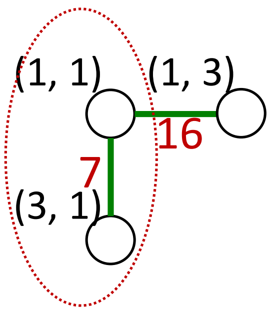
1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1



Technique for Generating a Maze

- Two edges have exactly one vertex corresponding to 0 in the array, (1, 1)-to-(1, 3) and (1, 1)-to-(3, 1)
  - We pick the latter one because it has lower cost
  - We update the array accordingly

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1

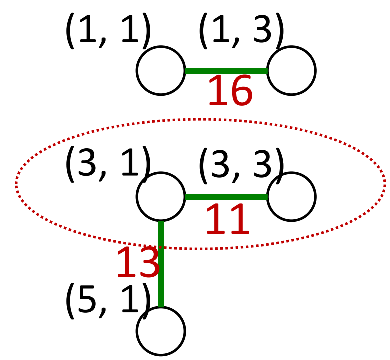


Candidate edges

Technique for Generating a Maze

- We repeat the above step
- This time we have three candidate edges
  - We pick (3, 1)-to-(3, 3) based on the costs
  - The array is update accordingly, too

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1

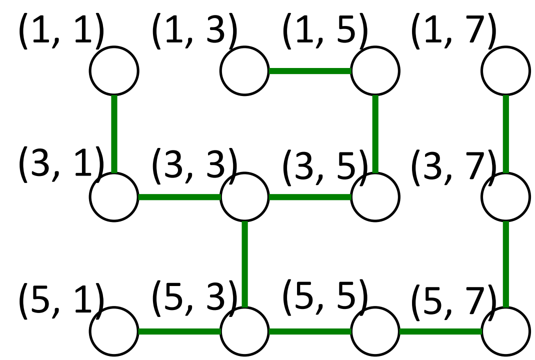


Candidate edges

Technique for Generating a Maze

- When no edges can be added to the tree, We obtained a spanning tree of the graph
- The array entries associated to the tree edges were set to 0's, too

1	1	1	1	1	1	1	1	1
1	0	1	0	0	0	1	0	1
1	0	1	1	1	0	1	0	1
1	0	0	0	0	0	1	0	1
1	1	1	0	1	1	1	0	1
1	0	0	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1

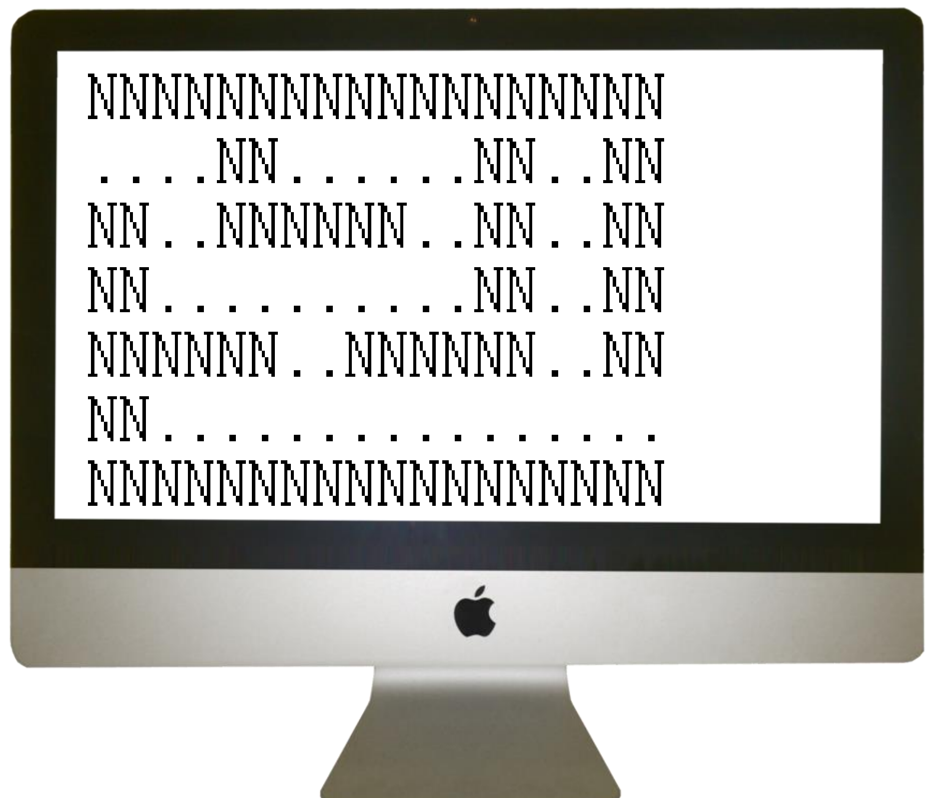


Corresponding spanning tree

Technique for Generating a Maze

- Lastly, we set the entrance and the exit to 0's
- For each 1 entry, we print out a pattern, e.g., "NN", and for each zero entry, we print out another pattern, e.g., ".."

1	1	1	1	1	1	1	1	1
0	0	1	0	0	0	1	0	1
1	0	1	1	1	0	1	0	1
1	0	0	0	0	0	1	0	1
1	1	1	0	1	1	1	0	1
1	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1



Input and Output

Input

Number of maze rows

Number of maze columns

Text pattern for 1 (i.e., wall)

Text pattern for 0 (i.e., path)

Number of edges

A line represents an edge

- 5 5 5 7 ,means (5, 5) to (5, 7)
- Edges are listed in an increasing cost order

```
7
9
NN
..
17
5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3
```

Output

```
NNNNNNNNNNNNNNNNNN
....NN......NN..NN
NN..NNNNNN..NN..NN
NN..........NN..NN
NNNNNN..NNNNNN..NN
NN................
NNNNNNNNNNNNNNNNNN
```

Columns

Rows

NN	NN	NN	NN	NN	NN	NN	NN	NN
..	..	NN	..	..	..	NN	..	NN
NN	..	NN	NN	NN	..	NN	..	NN
NN	..	..	..	..	..	NN	..	NN
NN	NN	NN	..	NN	NN	NN	..	NN
NN	..	..	..	..	..	..	..	..
NN	NN	NN	NN	NN	NN	NN	NN	NN

Detail Steps of the Above Example

- Read edges into an edge array.
- Initialize the maze array.

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1

```
5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3
```

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1	1
1	1	1	0	1	1	1	1	1
1	1	1	0	1	1	1	1	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1	1
1	1	1	0	1	1	1	1	1
1	1	1	0	0	0	1	1	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1	1
1	1	1	0	1	1	1	1	1
1	1	1	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	0	1
1	1	1	0	1	1	1	0	1
1	1	1	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	0	1
1	0	1	1	1	1	1	0	1
1	0	0	0	1	1	1	0	1
1	1	1	0	1	1	1	0	1
1	1	1	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	0	1
1	0	1	1	1	1	1	0	1
1	0	0	0	0	0	1	0	1
1	1	1	0	1	1	1	0	1
1	1	1	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	1	1	0	1	0	1
1	0	1	1	1	0	1	0	1
1	0	0	0	0	0	1	0	1
1	1	1	0	1	1	1	0	1
1	1	1	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	0	0	0	1	0	1
1	0	1	1	1	0	1	0	1
1	0	0	0	0	0	1	0	1
1	1	1	0	1	1	1	0	1
1	1	1	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
1	0	1	0	0	0	1	0	1
1	0	1	1	1	0	1	0	1
1	0	0	0	0	0	1	0	1
1	1	1	0	1	1	1	0	1
1	0	0	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1

5 5 5 7
5 3 5 5
3 3 5 3
3 7 5 7
1 3 1 5
1 7 3 7
1 1 3 1
1 5 3 5
3 3 3 5
5 1 5 3
3 1 3 3
3 5 5 5
3 1 5 1
3 5 3 7
1 5 1 7
1 1 1 3
1 3 3 3

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
0	0	1	0	0	0	1	0	1
1	0	1	1	1	0	1	0	1
1	0	0	0	0	0	1	0	1
1	1	1	0	1	1	1	0	1
1	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1

Detail Steps of the Above Example

1	1	1	1	1	1	1	1	1
0	0	1	0	0	0	1	0	1
1	0	1	1	1	0	1	0	1
1	0	0	0	0	0	1	0	1
1	1	1	0	1	1	1	0	1
1	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1

Output

```
NNNNNNNNNNNNNNNNNN
....NN......NN..NN
NN..NNNNNN..NN..NN
NN..........NN..NN
NNNNNN..NNNNNN..NN
NN................
NNNNNNNNNNNNNNNNNN
```

51x81 maze example

```
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
........NN..NN..NN..NN..................NN......................NN..............NN..............NN..NN..NN..........NN..........................NN..NN..NN..NN..NN
NN..NNNNNN..NN..NN..NNNNNN..NNNNNNNNNN..NNNNNNNNNN..NNNNNNNNNNNNNN..NNNNNNNNNNNNNNNNNNNNNN..NN..NN..NN..NNNNNN..NNNNNNNNNN..NNNNNN..NNNNNNNNNN..NN..NN..NN..NN..NN
NN..........NN..........NN..........NN..........NN..NN..NN..NN..............NN..NN......NN..NN..........NN......NN..NN..NN..NN..NN..NN......NN..NN......NN..NN..NN
NNNNNN..NNNNNN..NNNNNNNNNN..NNNNNNNNNNNNNN..NN..NN..NN..NN..NNNNNN..NNNNNNNNNN..NNNNNN..NN..NN..NN..NNNNNN..NNNNNN..NN..NN..NN..NN..NNNNNN..NN..NN..NNNNNN..NN..NN
NN......NN..NN..NN..NN..........NN..........NN..NN......NN..NN..NN..NN......NN......NN......NN..NN..NN..NN..NN..NN..NN......NN......NN......NN..............NN..NN
NN..NNNNNN..NN..NN..NN..NNNNNNNNNNNNNN..NNNNNN..NNNNNN..NN..NN..NN..NN..NN..NNNNNN..NN..NNNNNNNNNNNNNN..NN..NN..NN..NNNNNN..NNNNNNNNNN..NN..NNNNNNNNNNNNNN..NN..NN
NN..NN......NN..............NN..........NN......NN......NN......NN......NN..NN..........NN..........NN......NN..................NN..NN..NN..NN..NN..............NN
NN..NN..NNNNNNNNNNNNNNNNNNNNNNNNNN..NN..NNNNNNNNNNNNNN..NNNNNN..NN..NNNNNNNNNNNNNN..NN..NNNNNNNNNN..NNNNNN..NNNNNN..NNNNNN..NNNNNN..NN..NNNNNN..NNNNNN..NNNNNN..NN
NN..............NN..NN..NN......NN..NN......NN......NN..............NN..........NN..NN..................NN..NN......NN..........NN..NN......NN..........NN......NN
NNNNNNNNNN..NNNNNN..NN..NN..NNNNNN..NN..NN..NN..NNNNNN..NN..NN..NNNNNNNNNNNNNN..NNNNNNNNNN..NNNNNNNNNNNNNN..NNNNNNNNNNNNNN..NNNNNN..NNNNNN..NNNNNNNNNN..NNNNNNNNNN
NN......NN......NN..NN..............NN..NN......NN..NN..NN..NN..........NN......NN..........................NN..............NN..................NN......NN..NN..NN
NNNNNN..NN..NNNNNN..NN..NN..NNNNNNNNNNNNNNNNNNNNNN..NN..NNNNNNNNNN..NNNNNNNNNN..NN..NNNNNN..NNNNNNNNNN..NNNNNN..NNNNNN..NNNNNNNNNN..NNNNNN..NNNNNNNNNN..NN..NN..NN
NN..............NN..NN..NN..NN......NN..NN..........NN......NN..........................NN..NN..NN..NN..NN......NN..................NN..............NN..NN..NN..NN
NN..NN..NN..NNNNNN..NN..NNNNNNNNNN..NN..NN..NNNNNNNNNNNNNNNNNNNNNNNNNN..NN..NN..NN..NN..NN..NN..NN..NN..NNNNNN..NN..NN..NNNNNN..NN..NNNNNNNNNN..NNNNNNNNNN..NN..NN
NN..NN..NN..........................................NN......NN..NN......NN..NN..NN..NN..NN..........NN......NN..NN..NN......NN..NN..NN..................NN..NN..NN
NNNNNN..NNNNNNNNNNNNNN..NNNNNN..NNNNNNNNNN..NNNNNN..NN..NNNNNN..NNNNNNNNNNNNNNNNNNNNNNNNNN..NNNNNN..NNNNNNNNNN..NNNNNNNNNNNNNN..NNNNNNNNNN..NNNNNNNNNNNNNN..NN..NN
NN..NN......NN..........NN......NN......NN......NN..NN..NN..NN..NN..NN..NN......NN..NN......NN..NN......NN..........NN......NN..NN..NN..NN..............NN......NN
NN..NNNNNNNNNNNNNN..NN..NNNNNNNNNNNNNN..NNNNNN..NNNNNN..NN..NN..NN..NN..NNNNNN..NN..NNNNNNNNNN..NN..NN..NN..NNNNNNNNNN..NN..NNNNNN..NN..NNNNNNNNNN..NNNNNNNNNN..NN
NN..NN..............NN......NN..NN..................NN......NN......NN..............NN..............NN..................NN..........NN..............NN..........NN
NN..NNNNNN..NNNNNNNNNNNNNN..NN..NN..NNNNNNNNNNNNNNNNNN..NN..NN..NNNNNN..NNNNNNNNNNNNNN..NNNNNNNNNN..NNNNNN..NNNNNNNNNN..NNNNNNNNNNNNNNNNNNNNNN..NNNNNNNNNN..NNNNNN
NN......................NN..NN..........NN..NN......NN..NN..........NN..NN......................NN..NN..NN..NN..............NN..NN..NN..................NN..NN..NN
NNNNNNNNNN..NNNNNNNNNNNNNNNNNN..NNNNNN..NN..NN..NNNNNNNNNN..NNNNNN..NN..NNNNNNNNNNNNNN..NNNNNNNNNNNNNN..NNNNNNNNNN..NN..NNNNNN..NN..NN..NNNNNNNNNN..NN..NN..NN..NN
NN..NN......NN......................NN......NN......NN......NN......NN..NN......NN......NN......NN..................NN..NN......................NN..NN..NN......NN
NN..NNNNNNNNNN..NNNNNNNNNN..NNNNNN..NN..NNNNNN..NN..NNNNNN..NN..NNNNNN..NN..NNNNNNNNNNNNNN..NNNNNNNNNNNNNN..NN..NNNNNN..NNNNNN..NNNNNNNNNN..NNNNNN..NN..NN..NNNNNN
NN..................NN..........NN..NN..NN..NN..NN..........NN..........NN..........NN......NN..NN......NN..NN..NN..........NN......NN......NN......NN......NN..NN
NN..NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN..NN..NN..NNNNNN..NNNNNN..NNNNNNNNNN..NNNNNNNNNN..NN..NN..NN..NN..NN..NN..NNNNNN..NNNNNNNNNN..NNNNNNNNNN..NN..NNNNNN..NN..NN
NN......NN..........NN..NN......NN..NN..NN..NN..NN..........NN..........NN..NN..........NN..........NN......NN......NN..........NN..NN......NN..NN......NN......NN
NNNNNN..NNNNNN..NNNNNN..NN..NNNNNN..NNNNNN..NNNNNNNNNNNNNN..NNNNNNNNNNNNNN..NNNNNNNNNNNNNN..NN..NNNNNNNNNN..NNNNNN..NNNNNN..NN..NNNNNN..NNNNNNNNNNNNNNNNNN..NN..NN
NN......NN......................NN..................NN......................NN..NN..NN..NN..NN..........NN......NN......NN..NN..............NN..NN..........NN..NN
NN..NN..NNNNNNNNNNNNNN..NN..NN..NN..NNNNNN..NNNNNN..NN..NN..NNNNNNNNNN..NN..NN..NN..NN..NN..NN..NNNNNN..NNNNNNNNNN..NNNNNNNNNNNNNNNNNNNNNNNNNN..NNNNNNNNNNNNNNNNNN
NN..NN......NN..........NN..NN..........NN......NN..NN..NN..NN..NN..NN..NN......NN..NN......NN..NN......NN..............NN..........NN......NN..NN..........NN..NN
NNNNNN..NNNNNN..NN..NNNNNNNNNNNNNNNNNN..NN..NNNNNNNNNNNNNNNNNN..NN..NNNNNN..NNNNNN..NN..NN..NN..NNNNNNNNNNNNNN..NNNNNNNNNN..NNNNNNNNNNNNNN..NN..NN..NN..NNNNNN..NN
NN......NN......NN..NN..........NN......NN......NN..NN..NN......NN..NN..NN......NN......NN..NN......NN......NN..........NN......NN..................NN..........NN
NNNNNN..NN..NNNNNNNNNN..NN..NN..NN..NN..NNNNNNNNNN..NN..NNNNNN..NN..NN..NN..NN..NNNNNNNNNNNNNNNNNN..NNNNNN..NNNNNN..NNNNNN..NNNNNN..NN..NN..NNNNNNNNNNNNNN..NNNNNN
NN..NN......NN..NN..NN..NN..NN......NN......................NN..............NN..NN..NN......NN..NN..NN......NN..............NN..NN..NN..NN..NN......NN..........NN
NN..NN..NNNNNN..NN..NN..NNNNNNNNNNNNNNNNNN..NNNNNNNNNN..NN..NNNNNNNNNNNNNNNNNN..NN..NN..NNNNNN..NN..NN..NNNNNN..NNNNNNNNNN..NN..NNNNNN..NN..NN..NN..NNNNNN..NN..NN
NN..NN..........NN..........NN..............NN..........NN..NN......NN..NN......NN..............NN..NN......NN..NN..NN..NN..............NN..NN..NN..NN..NN..NN..NN
NN..NNNNNNNNNNNNNN..NNNNNNNNNNNNNN..NN..NNNNNN..NNNNNNNNNNNNNNNNNN..NN..NN..NN..NN..NN..NN..NNNNNN..NN..NNNNNNNNNN..NN..NNNNNN..NN..NNNNNNNNNN..NN..NN..NNNNNNNNNN
NN..NN..............NN..........NN..NN..NN..........NN..NN..............NN..NN......NN..NN..NN......NN..........NN..NN..NN......NN..NN..NN......NN..............NN
NN..NNNNNN..NNNNNNNNNN..NNNNNNNNNNNNNNNNNN..NN..NNNNNN..NNNNNNNNNNNNNN..NN..NNNNNNNNNNNNNNNNNNNNNNNNNN..NNNNNN..NN..NN..NN..NN..NN..NN..NN..NN..NN..NN..NNNNNNNNNN
NN......NN......................NN..NN..NN..NN..........NN..NN..............NN..NN..........NN..........NN......NN..........NN..NN..........NN..NN..NN......NN..NN
NN..NN..NNNNNNNNNN..NN..NNNNNNNNNN..NN..NNNNNN..NN..NN..NN..NN..NNNNNNNNNN..NN..NNNNNN..NN..NNNNNNNNNN..NNNNNNNNNN..NN..NNNNNN..NNNNNNNNNNNNNN..NNNNNNNNNNNNNN..NN
NN..NN......NN......NN..NN..NN..............NN..NN..NN..............NN..NN......NN......NN..NN..NN..............NN..NN......NN......NN......NN..................NN
NNNNNN..NNNNNN..NN..NNNNNN..NNNNNN..NN..NN..NN..NNNNNNNNNNNNNNNNNN..NN..NNNNNN..NNNNNNNNNN..NN..NN..NNNNNNNNNNNNNNNNNNNNNN..NNNNNNNNNN..NN..NNNNNN..NNNNNN..NNNNNN
NN..............NN..............NN..NN..NN..........NN..............NN......NN......NN......NN......NN..........................NN......NN..NN......NN..........NN
NN..NNNNNN..NNNNNNNNNN..NNNNNN..NN..NN..NNNNNNNNNNNNNN..NN..NN..NNNNNN..NNNNNNNNNN..NN..NN..NN..NNNNNNNNNN..NNNNNNNNNN..NNNNNN..NN..NNNNNN..NNNNNNNNNNNNNNNNNN..NN
NN..NN......NN..........NN......NN..NN..........NN......NN..NN..NN..............NN......NN..........................NN..NN..NN..NN..NN..NN......NN..........NN..NN
NNNNNN..NN..NNNNNN..NN..NN..NN..NNNNNN..NNNNNNNNNNNNNN..NNNNNN..NN..NNNNNNNNNN..NNNNNN..NN..NNNNNN..NNNNNNNNNNNNNN..NNNNNN..NN..NN..NN..NNNNNN..NN..NNNNNN..NN..NN
NN......NN..NN......NN..NN..NN..NN..............NN..........NN......NN..............NN..NN..NN..............NN..............NN..........NN......NN......NN........
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
```