

# Login Menkrep

Time Limit : 1s

Memory Limit: 2 GB

Author : Rif



Karena sudah weekend, sudah waktunya Adit bermain permainan minecraft. Minecraft adalah permainan sandbox populer buatan Mojang Studios, di mana pemain dapat membangun dan menjelajahi dunia tiga dimensi dari blok-blok kubus. Pemain mengumpulkan sumber daya, membuat alat, dan berinteraksi dengan berbagai makhluk, termasuk *zombie* yang muncul di malam hari dan menyerang *villager* serta pemain.

Suatu saat Adit menemukan sebuah masalah pada desanya, dimana villagernya habis diserang zombie di malam hari. Oleh karena itu ia mendesain struktur bangunan dimana tidak bisa dilewati oleh zombie. Karena Adit adalah seorang programmer handal, ia membuat kode untuk menentukan apakah struktur bangunan yang dibuatnya bisa melindungi villager atau tidak. Zhar, teman Adit berpikiran untuk membuat soal praktikum dasar pemrograman dari kode yang dibuat Adit. Sehingga zhar menghapus beberapa kode agar dilengkapi oleh praktikan. Tugas kalian adalah melengkapi kode yang telah dibuat.

Code: [link1](#) | [link2](#)

Note: Fokus modifikasi pada fungsi yang memang dibutuhkan soal :)

## Input:

Bilangan bulat  $N$  yang menyatakan besar map minecraft yang dimainkan Adit ( $N \times N$ ).

$N \times N$  selanjutnya berisi  $a_i b_j$  yang bisa berisi

V : lokasi villager,

X: struktur tembok yang dibangun Adit.

Z: lokasi zombie

O: blok kosong

### output:

Jika zombie mampu menjangkau villager maka outputkan  
OH NOO my villager dead

Sebaliknya jika zombie tidak mampu menjangkau villager maka outputkan  
This game is so ez.

### Constraint:

$0 \leq N \leq 100$

$0 \leq i, j < N$

Sample input 1:

4

VOOX

XX XO

OOOO

OOOZ

Sample output 1:

This game is so ez

Explanation 1:

	0	1	2	3
0	v			x
1	x	x	x	
2				
3				z

Setiap path yang mungkin dilewati zombie, tidak dapat menjangkau villager.  
Jadi villager aman

Sample input 2:

3  
OOO  
XXO  
VOZ

Sample output 2:

OH NOO my villager dead

Explanation 2:

	0	1	2
0			
1	x	x	
2	v		z

Zombie dapat menjangkau villager, maka villager akan mati saat malam

**RULE :**

**No Rule. Yang Penting AC, Boleh modifikasi seluruh kode ( kalau udah jago :) )**

# Login Menkrep

Time Limit : 1s

Memory Limit: 2 GB

Author : Rif



Since it's the weekend, it's time for Adit to play Minecraft. Minecraft is a popular sandbox game developed by Mojang Studios, where players can build and explore a three-dimensional world made of cube blocks. Players gather resources, craft tools, and interact with various creatures, including zombies that appear at night to attack villagers and players.

One day, Adit encounters a problem in his village where all villagers are wiped out by zombies at night. So, he designs a structure that zombies cannot pass through. Being a skilled programmer, Adit writes code to determine if the structure he built can protect the villagers. Zhar, Adit's friend, thought of creating a basic programming practice question from Adit's code by removing some parts of it, so it needs to be completed by students. Your task is to complete the code.

Code: [link1](#) | [link2](#)

Note: Focus on modifying only the functions required by the problem. :)

## Input:

An integer  $N$  representing the size of the Minecraft map that Adit is playing on ( $N \times N$ ).

Each of the  $N \times N$  cells contains:

- V: the villager's location,
- X: Adit's wall structure,
- Z: the zombie's location,
- O: an empty block.

### output:

If the zombie can reach the villager, output:

OH NOO my villager dead

Otherwise, if the zombie cannot reach the villager, output:

This game is so ez

### Constraint:

$0 \leq N \leq 100$

$0 \leq i, j < N$

Sample input 1:

4

VOOX

XX XO

OOOO

OOOZ

Sample output 1:

This game is so ez

Explanation 1:

	0	1	2	3
0	v			x
1	x	x	x	
2				
3				z

Every possible path from zombie can't reach villager, so villager is safe

Sample input 2:

```
3
OOO
XXO
VOZ
```

Sample output 2:

OH NOO my villager dead

Explanation 2:

	0	1	2
0			
1	x	x	
2	v		z

Zombie can reach villager, so villager will die

**RULE :**

**No Rule. You can modify the entire code.**

## How DFS Work

1. Terdapat Array Matrix NxN yang menyatakan map pada desa adit

v			x
x	x	x	
			z

v menyatakan villager Adit berada, x menyatakan struktur bangunan adit, dan z menyatakan zombie yang akan menyerang villager.

2. Implementasikan algoritma DFS ( depth first search ) rekursif untuk menentukan blok mana saja yang bisa dijangkau zombie. Langkah awal adalah menandai z sebagai starting point dan kemudian melakukan check apakah node sekarang adalah villager (v) atau bukan, jika bukan maka menandai node z sudah dikunjungi (D)

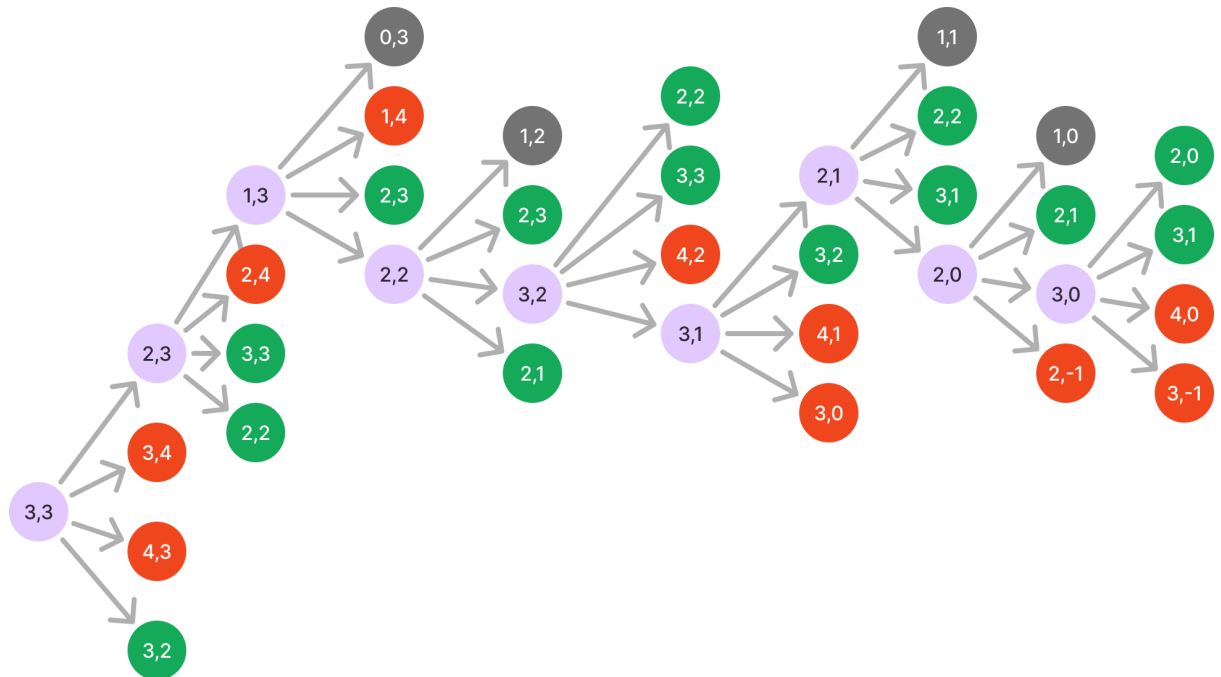
v			x
x	x	x	
			D

Setelah menandai sudah dikunjungi, juga melakukan rekursif pada node sekitar D, karena zombie hanya bisa berjalan pada blok atas, kanan, bawah, kiri, maka panggil fungsi findpath pada keempat blok tersebut.

3. Karena map berukuran 4x4 maka pemanggilan fungsi yang valid hanya atas dan kiri. Selanjutnya pindah ke (2,3) dan mengulangi step no 2

	0	1	2	3
0	v			x
1	x	x	x	
2				D
3				D

4. Untuk pemanggilan fungsi keseluruhan rekursif akan seperti ini:



abu-abu : base case dimana  $arr[i][j]$  adalah tembok

hijau : base case dimana  $arr[i][j]$  sudah dikunjungi (tanda D)

merah : base case dimana i atau j merupakan indeks array yang invalid

	0	1	2	3
0	v			x
1	x	x	x	D
2	D	D	D	D
3	D	D	D	D

5. Jika sampai akhir rekursif V tidak dapat dijangkau zombie, maka zombie tidak dapat menjangkau villager.



1. There is an NxN array matrix representing the map of Adit's village.

v			x
x	x	x	
			z

V represents the villager's location, X is the wall structure, and Z is the zombie trying to attack the villager.

2. Implement a recursive Depth-First Search (DFS) algorithm to determine which blocks the zombie can reach. Start by marking Z as the starting point and then check if the current node is the villager (V); if not, mark the node Z as visited (indicated by D).

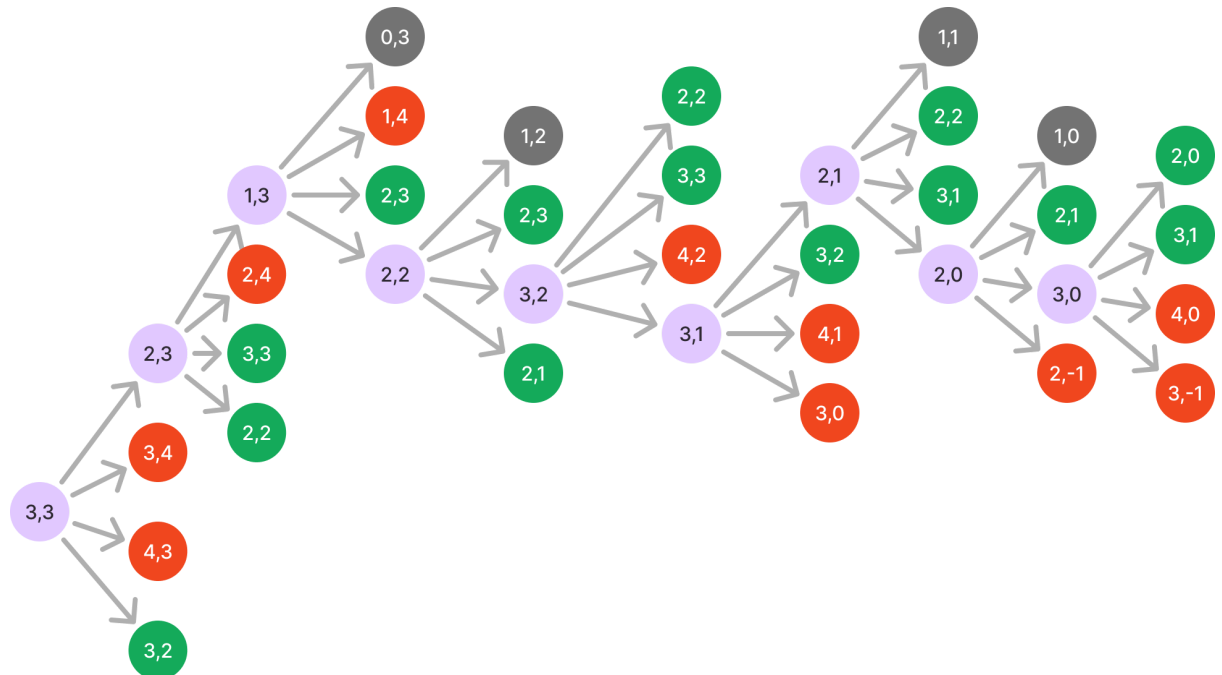
v			x
x	x	x	

After marking a node as visited, the recursive DFS checks the neighboring blocks (up, right, down, left).

3. For the whole recursive function, valid moves are only up and left due to the map's 4x4 size. Proceed to (2,3) and repeat step 2.

	0	1	2	3
0	v			x
1	x	x	x	
2				D
3				D

4. For the entire recursive function call, it will be as follows:



Gray: base case where  $\text{arr}[i][j]$  is a wall.

Green: base case where  $\text{arr}[i][j]$  has already been visited (marked with D).

Red: base case where  $i$  or  $j$  is an invalid array index."

	0	1	2	3
0	v			x
1	x	x	x	D
2	D	D	D	D
3	D	D	D	D

5. If, by the end of the recursion, V cannot be reached by the zombie, then the villager is safe.