Minimize Malware Spread

Try to solve the Minimize Malware Spread problem.

We'll cover the following

Statement

You're given a network of n nodes as an $n \times n$ adjacency matrix graph with the i^{th} node directly connected to the j^{th} node if `graph[i][j] == 1`.

A list of nodes, `initial`, is given, which contains nodes initially infected by malware. When two nodes are connected directly and at least one of them is infected by malware, both nodes will be infected by malware. This spread of malware will continue until every node in the connected component of nodes has been infected.

After the infection has stopped spreading, M will represent the final number of nodes in the entire network that have been infected with malware.

Return a node from `initial` such that, when this node is removed from the graph, M is minimized. If multiple nodes can be removed to minimize M, return the node with the smallest index.

> **Note:** If a node was removed from the initial list of infected nodes, it might still be infected later on due to the malware's spread.

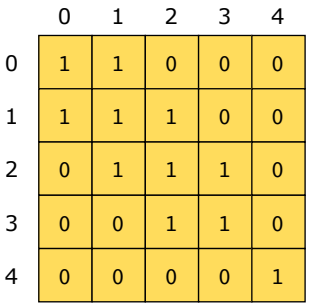
Constraints:

- `graph.length == graph[i].length`
- $2 \leq n \leq 300$
- `graph[i][j]` is 0 or 1.
- `graph[i][j] == graph[j][i]`
- `graph[i][i] == 1`
- $1 \leq$ `initial.length` $\leq n$
- $0 \leq$ `initial[i]` $\leq n - 1$
- All the integers in the `initial` are unique.

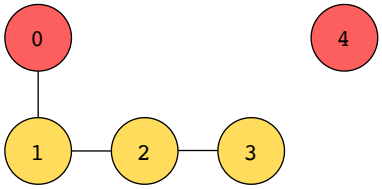
Examples

Sample example 1

Input



	0	1	2	3	4
0	1	1	0	0	0
1	1	1	1	0	0
2	0	1	1	1	0
3	0	0	1	1	0
4	0	0	0	0	1



initial	0	4

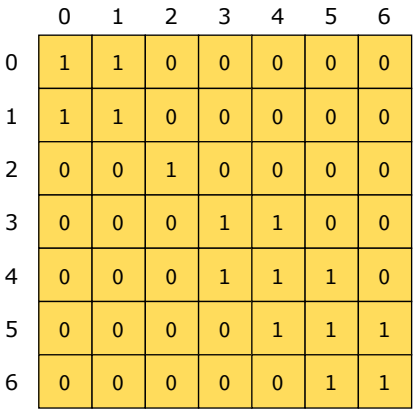
Output

0

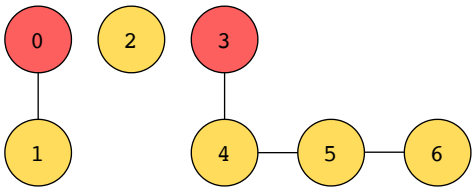
The initial infected nodes are **0** and **4**. Node **0** infects three nodes and node **4** infects zero nodes. So, removing node **0** will minimize the malware spread.

Sample example 2

Input



	0	1	2	3	4	5	6
0	1	1	0	0	0	0	0
1	1	1	0	0	0	0	0
2	0	0	1	0	0	0	0
3	0	0	0	1	1	0	0
4	0	0	0	1	1	1	0
5	0	0	0	0	1	1	1
6	0	0	0	0	0	1	1



initial	0	2	3

Output

3

The initial infected nodes are **0**, **2**, and **3**. Node **0** infects one node, node **2** infects zero nodes, and node **3** infects three nodes. So, removing node **3** will minimize the malware spread.

Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

Minimize Malware Spread

1

Given the following graph and the initially infected nodes, which node will help minimize the malware spread?

```
graph = [[1, 1, 1],
         [1, 1, 0],
         [1, 0, 1]]

initial = [1, 2]
```

A) 0

B) 1

C) 2

Submit Answer

Question 1 of 2
0 attempted

Reset Quiz

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

Drag and drop the cards to rearrange them in the correct sequence.

Make connected components out of all the connected nodes in the graph through the Union Find algorithm.

Traverse the **initial** array and store the number of infections in each connected component with an infection in a hash map, **infected**.

If a connected component from the **infected** hash map has more than one infected node, ignore it and move to the next iteration of the loop. Otherwise, calculate the size of the component.

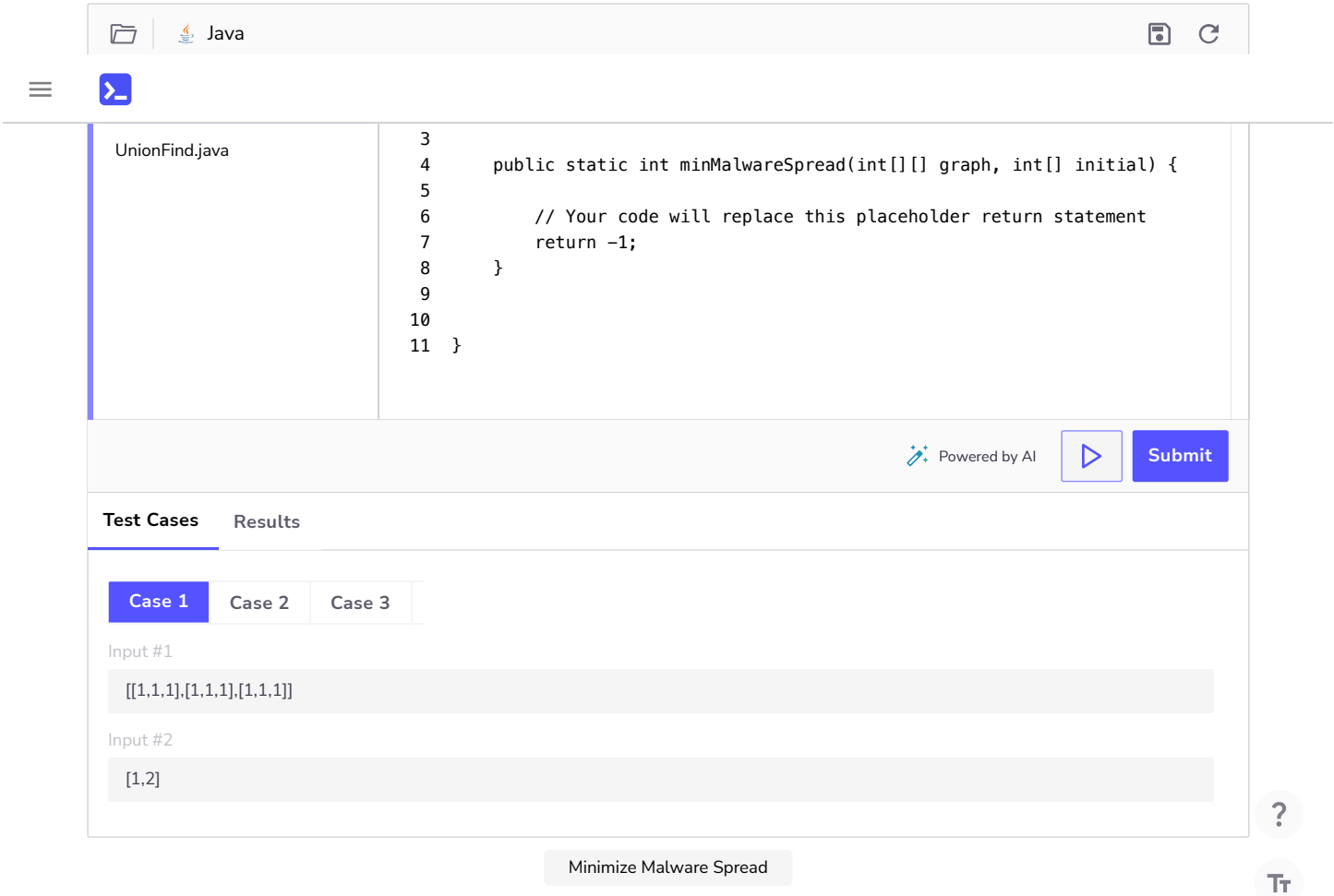
If a connected component from the **infected** hash map has more than one infected node, ignore it and move to the next iteration of the loop. Otherwise, calculate the size of the component.

If there are multiple components of the same size that would count as the largest connected component, choose the one with the smallest index.

Reset | Show Solution | Submit

Try it yourself

Implement your solution in the following coding playground. You will need the provided supporting code to implement your solution.



Minimize Malware Spread

Back | Next

Mark as Completed