Problem 2 – Office Space

After passing your "data structures and algorithms" exam in Telerik Academy you started to work as a software developer in a big corporation called Initech. Sadly for you in this company your direct manager is an annoying douchebag called William "Bill" Lumbergh. Lumbergh is a micromanager who is focused on pointless paperwork. Here is a video of him recorded by your colleagues during working with him: http://goo.gl/ScZzFK

Of course you hate Lumbergh and you think he is absolutely redundant for the company. His only job is to calculate and report the minimum number of minutes needed for all his subordinates to complete their tasks. As a software developer who passed the "data structures and algorithms" exam you know that this job can be automated by a computer program. Write a computer program to do the calculation and prove that Lumbergh is no more needed for the company so once and for all get rid of him.

You and your colleagues can do their tasks **simultaneously** without disturbing each other. Certain tasks may need some other tasks to be completed before they can be started. No task is dependent on itself.

Tasks are numbered from **1** to **N**. For each task you are given an integer – the time (in minutes) required for the task to be completed. You are also given the requirements (dependencies) for each of the tasks.

Write a program to find the **minimum number of minutes needed for all tasks to be completed** given that you have at least 50 colleagues.

If it's impossible for all the tasks to be completed, print **`-1`**.

Input

The input data should be read from the console.

On the first input console line there will be the number **N**.

On the second line there will be the times for each task (from 1 to N) separated by a single space (' ').

On each of the next **N** lines there will be the list of dependencies (requirements) for each of the tasks (from 1 to N) separated by a single space. If the list contains only the number `0` – the task has no requirements. Otherwise in the list will be the task numbers that are required to be finished before the task can be stated.

See the examples below for clarification.

The input data will always be valid and in the format described. There is no need to check it explicitly.

Output

The output data should be printed on the console.

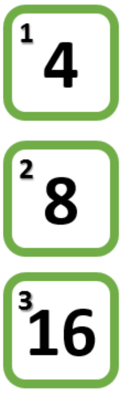
On the only output line on the console write the **minimum number of minutes needed for all the tasks to be completed**. If this is impossible print "`-1`" instead.

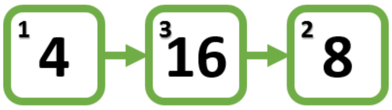
Constraints

- **N** will be between 1 and 50, inclusive.
- The time for each task will be an integer between 1 and 1000, inclusive.
- Allowed working time for your program: 0.1 seconds.
- Allowed memory: 32 MB.

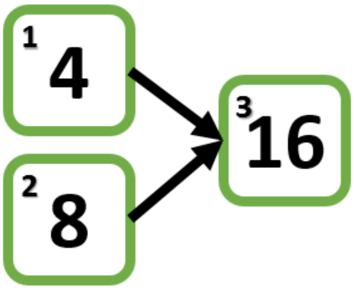
Examples

Input	Output	Explanation

3
4 8 16
0
0
0
	16	There is no task that depends on another.
		All 3 tasks can be done simultaneously.
		

3
4 8 16
0
3
1
	28	2nd task depends on the 3rd
		3rd task depends on the 1st
		

Input	Output	Explanation

3
4 8 16
0
0
1 2
	24	The 3rd task depends on the 1st and the 2nd
		

3
4 8 16
3
1
2
	-1	1st task depends on the 3rd
		2nd task depends on the 1st
		3rd task depends on the 2nd