

MENDO

System for competitions in informatics 2026

English

Македонски

Home

Learn C++

Competitions

Help

» 739810 tested submissions | 22772 users «

Home

Training

Programmers

Chat OFFLINE

AI

Login

Username

Password

☐ Remember Me

Submit

Password forgotten?

Navigation

Register

Learn C++

Algorithms

Training

Competitions

Forums

Wiki

Help

Report Bug

Programmers

In Macedonia the next state elections are scheduled for the end of this year. Boyan, as a student (soon to be officially engineer-programmer), decided that he should form a Party of engineer-programmers (further called only programmers).

He has contacts of all N programmers in Macedonia, so he sent them an initial invitation to join PP. Some of them accepted and the others declined. He made a list of the people who accepted.

Boyan may use this list to go to the Central Registry and register PP. However, he also wants to maximize the number of people who form the Party. He knows that if he sends the list of people who accepted the initial invitation to the programmers, then he may persuade others to accept. The rule is as following: Everyone that gets the list makes a new decision solely based on that (latest) list. If in that list there is at least one of his/her friends, than he/she accepts, otherwise he/she declines.

Boyan is a very honest student so if he decides to send an invitation, he sends the list to all N programmers with a new invitation to join the Party. To state this in another way - we may say that he goes for a 'second round'. After the second round he gets a new response (accept or decline) from all N programmers. So, he may compose a new list and go for a third, or fourth, or fifth, ... round.

If Boyan has complete knowledge of which programmers are friends between each other, help him calculate the number of programmers that will be in the longest list of programmers who accepted an invitation (made after the first or second or third... round).

Input

The first line of input contains N, the number of programmers ($1 \leq N \leq 50000$). Line 2 contains N integers B_i separated with one space $1 \leq i \leq N$. Here, $B_i=1$ if the i-th programmer accepted the initial invitation and $B_i=0$ if he/she declined.

The third line contains M, the number of pairs of friends ($0 \leq M \leq 200000$). In each of the next M lines, there is a pair of integers that represent a pair of programmers who are friends.

Output

The output consists of one integer - the number of programmers that will be in the longest list.

Constraints

Time limit: 1 second
Memory limit: 64 megabytes

Examples

input

3
1 0 0
2
1 2
1 3

output

2

input

3
1 1 0
2
1 2
1 3

output

3

Explanation for the first test case: Only the first programmer accepted the initial invitation. Then Boyan sends the list to everybody. After the second round of invitations, the Programmers 2 and 3 accept (because 2 is friends with 1 and 3 is friends with 1 - who was on the initial list) and programmer 1 declines the invitation since his friends 2 and 3 were not on the list. The list of accepted programmers after the second round has the programmers 2 and 3. After the next round only 1 will accept and so on. So, the number of programmers in the longest list is 2.

Explanation for the second test case: After the initial invitation 1 and 2 are on the list. In the second round 1 will accept because his friend 2 is on the list, 2 will accept because his friend 1 is on the list and

https://mendo.mk/Task.do?id=691

1/2

3 will accept because his friend 1 is on the list. All 3 programmers are now on the list, hence the answer is 3.

Submit **Test**

Submit your code

Solution language:

Automatic Detection...



Upload solution (option 1):

Select

Paste your code (option 2):



CodeMirror

Submit!

Copyright © 2026 Mendo Judge Team | All Rights Reserved
Supported by [CSM](#) | Powered by [Struts](#) | [XHTML 1.0](#) | [CSS 2.0](#) | [FAMFAMFAM](#)
» This website (project) is dedicated to the memory of [Nikola Postolov](#) «