Topological Sorting

ZPRAC-16-17-Lab10

[30 points]

Several universities allow students to take any courses they want. However, most courses have some prerequisites that the student needs to complete before they can take the said course. This creates a dependency graph of courses in which one depends on others.

Given such a dependency graph your task is to output the order in which the courses can be taken.

The dependency graph will be given in NxN matrix form, where 1 means row number depends on column number and 0 means it does not.
For example:
Let there be 3 courses (1,2 and 3) with following dependencies:
1 depends on 2 and 3
2 depends on nothing
3 depends on 2

Then, the matrix will be
0 1 1
0 0 0
0 1 0

And the proper ordering for this would be:
2 3 1

If at any stage multiple courses can be taken print them in ascending order.

INPUT FORMAT:

N (int) --- number of courses
(NxN matrix of 0/1) --- the dependency matrix

OUTPUT FORMAT:

(N space separated integers) --- order in which courses can be taken

EXAMPLE:

INPUT:
3
0 1 1
0 0 0
0 1 0

OUTPUT:
2 3 1

Explanation: Course 2 does not depend on anything thus it is first, followed by 3 whose dependency was 2 and finally 1 who had both 2 & 3 as dependencies.

INPUT:
4
0 0 0 0
0 0 1 0
1 0 0 0
1 0 0 0

OUTPUT:
1 3 2 4

Explanation: The first course to take must be 1 since it does not have any dependencies. After 1 both 3 and 4 can be taken, we take in ascending order so 3 is chosen. After 1,3 both 2 and 4 can be taken. As 2 is lesser than 4 we first take 2 then 4.