(Adv.) Competitive Programming

Submit until end of contest, via the judge interface

HPI Hasso Plattner Institut
Digital Engineering · Universität Potsdam

Problem: topics (1 second timelimit)

You are planning the lecture Competitive Programming for the summer term and already know which topics you want to cover. However, you cannot just present them in any order: Many topics require knowledge of earlier ones to be understood. For example, presenting flow algorithms before some basic graph theory would leave your students quite confused. Additionally, you also want the classes to have a nice difficulty curve: Whenever possible, you want to cover easier topics before harder ones. Can you find a topic schedule that fits your requirements?

Input The first line contains n, the number of topics, and m, the number of dependencies between topics ($1 \leq n, m \leq 2 \cdot 10^5$). The second line contains the numbers d_1 to d_n ($0 \leq d_i \leq 10^9$), the difficulties of the respective topics. The difficulties for topics are pairwise distinct. The next m lines each contain numbers a and b ($1 \leq a, b \leq n$), noting that topic a must be completed before topic b.

Output Write a schedule by listing the ids of the topics (1 to n) in their order in a single line. It is guaranteed that such a schedule exists.

Sample input	Sample output
3 1 1 2 3 2 1	2 1 3
3 1 1 2 3 3 2	1 3 2
3 2 1 2 3 2 1 3 2	3 2 1