

Topological Sorting

The problem is the same as [TOPOSORT](#), but the limits are different and the tests are different. The differences are highlighted below. Now, the TOPOSORT problem has only solutions in C/++ so it seems impossible to solve it in other languages, for silly reasons. With the tests here, you should still need to find sub-quadratic solutions, but you may do so in languages with I/O that isn't super-fast.

So, the problem: You are given a directed graph, and you should find the [lexicographically minimal topological sort](#) of it.

The first line of input has the number n of vertices and the number m of arcs. We have **$0 \leq m, n \leq 200000$** . The vertices are the numbers $1, \dots, n$. The next m numbers each list one arc by giving the source and the target vertex. If there is no cycle, you should output a permutation of the vertices, on one line, whitespace-separated. If there is a cycle, you should print one line containing "Sandro fails." (dot included).