2 Determining an Order of Courses

Problem Introduction

Now, when you are sure that there are no cyclic dependencies in the given CS curriculum, you would like to find an order of all courses that is consistent with all dependencies. For this, you find a topological ordering of the corresponding directed graph.

Problem Description

Task. Compute a topological ordering of a given directed acyclic graph (DAG) with n vertices and m edges.

Input Format. A graph is given in the standard format.

Constraints. $1 \leq n \leq 10^5$, $0 \leq m \leq 10^5$. The given graph is guaranteed to be acyclic.

Output Format. Output *any* topological ordering of its vertices. (Many DAGs have more than just one topological ordering. You may output any of them.)

Time Limits.

language	C	C++	Java	Python	C#	Haskell	JavaScript	Ruby	Scala
time (sec)	2	2	3	10	3	4	10	10	6

Memory Limit. 512MB.

Sample 1.
Input:
```
4 3
1 2
4 1
3 1
```
Output:
```
4 3 1 2
```


Sample 2.
Input:
```
4 1
3 1
```
Output:
```
2 3 1 4
```

Sample 3.

Input:
```
5 7
2 1
3 2
3 1
4 3
4 1
5 2
5 3
```

Output:
```
5 4 3 2 1
```