

{% note info %}

摘要

Title: 848. 有向图的拓扑序列

Tag: 拓扑排序

Memory Limit: 64 MB

Time Limit: 1000 ms

{% endnote %}

Powered by: NEFU AB-IN

[Link](#)

[@TOC](#)

848. 有向图的拓扑序列

• 题意

给定一个 n 个点 m 条边的有向图，点的编号是 1 到 n ，图中可能存在重边和自环。

请输出任意一个该有向图的拓扑序列，如果拓扑序列不存在，则输出 -1 。

• 思路

有向图的拓扑遍历，就是图的宽搜的应用

定义：若一个由图中所有点构成的序列 \mathbf{A} 满足：对于图中的每条边 (\mathbf{x}, \mathbf{y}) ， \mathbf{x} 在 \mathbf{A} 中都出现在 \mathbf{y} 之前，则称 \mathbf{A} 是该图的一个拓扑序列。

{% note info %}

有向无环图（DAG）一定存在拓扑序列

{% endnote %}

{% note info %}

有重边和自环怎么办？

有环是会输出 -1 的，因为环上的点入度一定大于 1 ，是没有突破口的

重边不影响判断，多走几个边就是了

{% endnote %}

- 代码

```
'''
Author: NEFU AB-IN
Date: 2022-03-02 21:20:54
FilePath: \ACM\Acwing\848.py
LastEditTime: 2022-03-02 21:26:53
'''

from collections import deque

N = int(1e5 + 10)

g = [[] for _ in range(N)]
st = [0] * N
deg = [0] * N #入度
res = []
q = deque()

n, m = map(int, input().split())
for i in range(m):
    a, b = map(int, input().split())
    g[a].append(b)
    deg[b] += 1

for i in range(1, n + 1):
    if deg[i] == 0:
        q.appendleft(i)

while q:
    t = q.pop()
    for j in g[t]:
        deg[j] -= 1
        if deg[j] == 0:
            q.appendleft(j)
    res.append(t)

if len(res) == n:
    print(" ".join(map(str, res)))
else:
    print(-1)
```