```cpp
1  #include "../bits/stdc++.h"
2  // add edge していって  topologicalSort() でソート済みの列を返す
3  // O(N)
4  // verified: https://atcoder.jp/contests/nikkei2019-qual/submissions/4246223
5  class TopologicalSort
6  {
7      using Graph = std::vector<std::vector<int>>;
8      // 頂点数
9      int V;
10     std::vector<std::vector<int>> G;
11     void dfs(int v, std::vector<int> &res, std::vector<int> &used)
12     {
13         used[v] = 1;
14         for (auto to : G[v])
15             if (used[to] == 0)
16             {
17                 dfs(to, res, used);
18             }
19         res.push_back(v);
20     }
21
22   public:
23     TopologicalSort(int _v) : V(_v), G(_v) {}
24     void addEdge(int a, int b)
25     {
26         assert(0 <= a && a < V);
27         assert(0 <= b && b < V);
28         G[a].push_back(b);
29     }
30     std::vector<int> topologicalSort()
31     {
32         std::vector<int> ide(V);
33         for (int i = 0; i < V; i++)
34         {
35             for (auto to : G[i])
36             {
37                 ide[to]++;
38             }
39         }
40         std::vector<int> res;
41         std::vector<int> used(V);
42         for (int i = V - 1; i >= 0; i--)
43         {
44             if (ide[i] == 0)
45             {
46                 dfs(i, res, used);
47             }
48         }
49         if (res.size() == 0)
50             res = {-1};
51         reverse(res.begin(), res.end());
52         return res;
53     }
54 };
55
```