

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

Solution 2

```
1 # Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 # O(j + d) time | O(j + d) space
4 def topologicalSort(jobs, deps):
5     jobGraph = createJobGraph(jobs, deps)
6     return getOrderedJobs(jobGraph)
7
8
9 def createJobGraph(jobs, deps):
10    graph = JobGraph(jobs)
11    for job, dep in deps:
12        graph.addDep(job, dep)
13    return graph
14
15
16 def getOrderedJobs(graph):
17    orderedJobs = []
18    nodesWithNoPrereqs = list(filter(lambda node: node.numOfPrereqs ==
19    while len(nodesWithNoPrereqs):
20        node = nodesWithNoPrereqs.pop()
21        orderedJobs.append(node.job)
22        removeDeps(node, nodesWithNoPrereqs)
23    graphHasEdges = any(node.numOfPrereqs for node in graph.nodes)
24    return [] if graphHasEdges else orderedJobs
25
26
27 def removeDeps(node, nodesWithNoPrereqs):
28    while len(node.deps):
29        dep = node.deps.pop()
30        dep.numOfPrereqs -= 1
31        if dep.numOfPrereqs == 0:
32            nodesWithNoPrereqs.append(dep)
33
```

Solution 1

Solution 2

Solution 3

```
1 def topologicalSort(jobs, deps):
2     # Write your code here.
3     pass
4
```

Our Tests

Custom Output

Submit Code

```
1 def topologicalSort(jobs, deps):
2     # Write your code here.
3     pass
4
```

```
1 def topologicalSort(jobs, deps):
2     # Write your code here.
3     pass
4
```

Run or submit code when you're ready.

Run or submit code when you're ready.