AlgoExpert Quad Layout JavaScript 12px Sublime Monokai 00:00:00

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1 Solution 2

```
1\, // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 // O(j + d) time | O(j + d) space
 4 function topologicalSort(jobs, deps) {
     const jobGraph = createJobGraph(jobs, deps);
      return getOrderedJobs(jobGraph);
9 function createJobGraph(jobs, deps) {
10
    const graph = new JobGraph(jobs);
11
      for (const [prereq, job] of deps) \{
12
       graph.addPrereq(job, prereq);
13
14
     return graph;
15 }
16
17 function getOrderedJobs(graph) {
     const orderedJobs = [];
18
19
      const {nodes} = graph;
20
      while (nodes.length) {
       const node = nodes.pop();
21
        const containsCycle = depthFirstTraverse(node, orderedJobs);
22
23
        if (containsCycle) return [];
24
25
      return orderedJobs;
26 }
27
28
    function \ depthFirstTraverse(node, \ orderedJobs) \ \{
29
     if (node.visited) return false;
30
      if (node.visiting) return true;
31
      node.visiting = true;
     for (const prereqNode of node.prereqs) \{
32
33
       const containsCycle = depthFirstTraverse(prereqNode, orderedJobs);
34
        if (containsCycle) return true;
35
36
      node.visited = true;
37
      node.visiting = false;
     orderedJobs.push(node.job);
38
39
      return false;
40
41
42 class JobGraph {
43
      \verb|constructor(jobs)| \{
        this.nodes = [];
44
        this.graph = {};
45
46
        for (const job of jobs) {
47
          this.addNode(job);
48
49
50
51
      addPrereq(job, prereq) {
52
        const jobNode = this.getNode(job);
53
        const prereqNode = this.getNode(prereq);
54
        jobNode.prereqs.push(prereqNode);
55
56
57
      addNode(job) {
58
        this.graph[job] = new JobNode(job);
59
        this.nodes.push(this.graph[job]);
60
61
62
      getNode(job) {
63
        if (!(job in this.graph)) this.addNode(job);
        return this.graph[job];
64
65
66
67
68 class JobNode {
     constructor(job) {
69
70
       this.job = job;
71
        this.prereqs = [];
72
        this.visited = false;
73
        this.visiting = false;
74
75 }
76
77 exports.topologicalSort = topologicalSort;
```