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

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1 Solution 2

```javascript
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

// O(j + d) time | O(j + d) space
function topologicalSort(jobs, deps) {
  const jobGraph = createJobGraph(jobs, deps);
  return getOrderedJobs(jobGraph);
}

function createJobGraph(jobs, deps) {
  const graph = new JobGraph(jobs);
  for (const [job, dep] of deps) {
    graph.addDep(job, dep);
  }
  return graph;
}

function getOrderedJobs(graph) {
  const orderedJobs = [];
  const nodesWithNoPrereqs = graph.nodes.filter(node => !node.numOfPrereqs);
  while (nodesWithNoPrereqs.length) {
    const node = nodesWithNoPrereqs.pop();
    orderedJobs.push(node.job);
    removeDeps(node, nodesWithNoPrereqs);
  }
  const graphHasEdges = graph.nodes.some(node => node.numOfPrereqs);
  return graphHasEdges ? [] : orderedJobs;
}

function removeDeps(node, nodesWithNoPrereqs) {
  while (node.deps.length) {
    const dep = node.deps.pop();
    dep.numOfPrereqs--;
    if (!dep.numOfPrereqs) nodesWithNoPrereqs.push(dep);
  }
}

class JobGraph {
  constructor(jobs) {
    this.nodes = [];
    this.graph = {};
    for (const job of jobs) {
      this.addNode(job);
    }
  }

  addDep(job, dep) {
    const jobNode = this.getNode(job);
    const depNode = this.getNode(dep);
    jobNode.deps.push(depNode);
    depNode.numOfPrereqs++;
  }

  addNode(job) {
    this.graph[job] = new JobNode(job);
    this.nodes.push(this.graph[job]);
  }

  getNode(job) {
    if (!(job in this.graph)) this.addNode(job);
    return this.graph[job];
  }
}

class JobNode {
  constructor(job) {
    this.job = job;
    this.deps = [];
    this.numOfPrereqs = 0;
  }
}

exports.topologicalSort = topologicalSort;
```