

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 class Program {
4     class JobNode {
5         let job: Int
6         var dependencies: [JobNode]
7         var numberOfPrerequisites: Int
8
9         init(job: Int) {
10             self.job = job
11             dependencies = [JobNode]()
12             numberOfPrerequisites = 0
13         }
14     }
15
16     class JobGraph {
17         var nodes: [JobNode]
18         var graph: [Int: JobNode]
19
20         init(jobs: [Int]) {
21             nodes = [JobNode]()
22             graph = [Int: JobNode]()
23             for job in jobs {
24                 addNode(job: job)
25             }
26         }
27
28         func addNode(job: Int) {
29             let jobNode = JobNode(job: job)
30
31             nodes.append(jobNode)
32             graph[job] = jobNode
33         }
34     }
35 }
```

Solution 1Solution 2Solution 3

```
1 class Program {
2     func topologicalSort(jobs: [Int], dependencies: [[Int]]) -> [Int] {
3         // Write your code here.
4         return []
5     }
6 }
7
```

Our Tests

```
1 // Test Program
2
3 // Test Case 1
4 let jobs = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
5 let dependencies = [[2, 1], [3, 1], [4, 2], [5, 2], [6, 3], [7, 3], [8, 4], [9, 4], [10, 5]]
6
7 // Expected Output: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
8
9 // Test Case 2
10 let jobs = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
11 let dependencies = [[2, 1], [3, 1], [4, 2], [5, 2], [6, 3], [7, 3], [8, 4], [9, 4], [10, 5]]
12
13 // Expected Output: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Custom Output

Submit Code

