

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 package main
4
5 import "sort"
6
7 type AirportNode struct {
8     Airport      string
9     Connections  []string
10    IsReachable   bool
11    UnreachableConnections []string
12 }
13
14 func NewAirportNode(airport string) *AirportNode {
15     return &AirportNode{
16         Airport:      airport,
17         Connections:  []string{},
18         IsReachable:   true,
19         UnreachableConnections: []string{},
20     }
21 }
22
23 // O(a * (a + r) + a + r + alog(a)) time | O(a + r) space - where a is
24 func AirportConnections(airports []string, routes [][]string, starting
25     airportGraph := createAirportGraph(airports, routes)
26     unreachableAirportNodes := getUnreachableAirportNodes(airportGraph,
27     markUnreachableConnections(airportGraph, unreachableAirportNodes)
28     return getMinNumberOfNewConnections(airportGraph, unreachableAirport
29 }
30
31 // O(a + r) time | O(a + r) space
32 func createAirportGraph(airports []string, routes [][]string) map[stri
33     airportGraph := map[string]*AirportNode{
```

Solution 1

Solution 2

Solution 3

```
1 package main
2
3 func AirportConnections(airports []string, routes [][]string, starting
4     // Write your code here.
5     return -1
6 }
7
```

Our Tests

Custom Output

Submit Code

```
1 package main
2
3 func main() {
4     // Write your code here.
5 }
```

```
1 package main
2
3 func main() {
4     // Write your code here.
5 }
```

Run or submit code when you're ready.