

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 type OrgChart struct {
6     Name string
7     DirectReports []*OrgChart
8 }
9
10 // O(n) time | O(d) space - where n is the number of people
11 // in the org and d is the depth (height) of the org chart
12 func GetLowestCommonManager(org, reportOne, reportTwo *OrgChart) *OrgChart {
13     return getOrgInfo(org, reportOne, reportTwo).lowestCommonManager
14 }
15
16 type OrgInfo struct {
17     lowestCommonManager *OrgChart
18     numImportantReports int
19 }
20
21 func getOrgInfo(manager, reportOne, reportTwo *OrgChart) OrgInfo {
22     numImportantReports := 0
23     for _, directReport := range manager.DirectReports {
24         orgInfo := getOrgInfo(directReport, reportOne, reportTwo)
25         if orgInfo.lowestCommonManager != nil {
26             return orgInfo
27         }
28         numImportantReports += orgInfo.numImportantReports
29     }
30     if manager == reportOne || manager == reportTwo {
31         numImportantReports++
32     }
33     var lowestCommonManager *OrgChart
```

Solution 1   Solution 2   Solution 3

```
1 package main
2
3 type OrgChart struct {
4     Name string
5     DirectReports []*OrgChart
6 }
7
8 func GetLowestCommonManager(org, reportOne, reportTwo *OrgChart) *OrgChart {
9     // Write your code here.
10    return nil
11 }
12
```

Our Tests

Custom Output

Submit Code

