Our Solution(s) Run Code

Solution 1

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

package main

// O(wh) time | O(wh) space
func RiverSizes(matrix [][]int) []int {
  sizes := []int{}
  visited := make([][]bool, len(matrix))
  for i := range visited {
    visited[i] = make([]bool, len(matrix[i]))
  }
  for i := range matrix {
    for j := range matrix[i] {
      if visited[i][j] {
        continue
      }
      sizes = traverseNode(i, j, matrix, visited, sizes)
    }
  }
  return sizes
}

func traverseNode(i, j int, matrix [][]int, visited [][]bool, si
  currentRiverSize := 0
  nodesToExplore := [][]int{{i, j}}
  for len(nodesToExplore) > 0 {
    currentNode := nodesToExplore[0]
    nodesToExplore = nodesToExplore[1:]
    i, j := currentNode[0], currentNode[1]
    if visited[i][j] {
      continue
    }
    visited[i][j] = true
```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```go
package main

func RiverSizes(matrix [][]int) []int {
  // Write your code here.
  return nil
}

```

Our Tests

Custom Output Submit Code

Run or submit code when you're ready.