Our Solution(s)   Run Code

Solution 1   **Solution 2**

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

package main

// O(n) time | O(n) space - where n is the total number of elements in the array
func SpiralTraverse(array [][]int) []int {
  if len(array) == 0 {
    return []int{}
  }
  result := []int{}
  spiralFill(array, 0, len(array)-1, 0,
    len(array[0])-1, &result)
  return result
}

func spiralFill(array [][]int, startRow, endRow, startCol, endCol int, result *[]
  if startRow > endRow || startCol > endCol {
    return
  }

  for col := startCol; col <= endCol; col++ {
    *result = append(*result, array[startRow][col])
  }

  for row := startRow + 1; row <= endRow; row++ {
    *result = append(*result, array[row][endCol])
  }

  for col := endCol - 1; col >= startCol; col-- {
    if startRow == endRow {
      break
    }
    *result = append(*result, array[endRow][col])
  }

  for row := endRow - 1; row >= startRow+1; row-- {
    if startCol == endCol {
      break
    }
    *result = append(*result, array[row][startCol])
  }

  spiralFill(array, startRow+1, endRow-1,
    startCol+1, endCol-1, result)
}
```

Your Solutions   Run Code

Solution 1   **Solution 2**   **Solution 3**

```go
package main

func SpiralTraverse(array [][]int) []int {
  // Write your code here.
  return nil
}
```

Custom Output   **Raw Output**   Submit Code

Run or submit code when you're ready.