Our Solution(s) Run Code

Solution 1 Solution 2

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

import java.util.*;

class Program {
  // O(n) time | O(n) space - where n is the total number of ele
  public static List<Integer> spiralTraverse(int[][] array) {
    if (array.length == 0) return new ArrayList<Integer>();

    var result = new ArrayList<Integer>();
    var startRow = 0;
    var endRow = array.length - 1;
    var startCol = 0;
    var endCol = array[0].length - 1;

    while (startRow <= endRow && startCol <= endCol) {
      for (int col = startCol; col <= endCol; col++) {
        result.add(array[startRow][col]);
      }

      for (int row = startRow + 1; row <= endRow; row++) {
        result.add(array[row][endCol]);
      }

      for (int col = endCol - 1; col >= startCol; col--) {
        if (startRow == endRow) break;
        result.add(array[endRow][col]);
      }

      for (int row = endRow - 1; row > startRow; row--) {
        if (startCol == endCol) break;
        result.add(array[row][startCol]);
      }
```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```java
import java.util.*;

class Program {
  public static List<Integer> spiralTraverse(int[][] array) {
    // Write your code here.
    return new ArrayList<Integer>();
  }
}
```

Our Tests

Custom Output Submit Code

```
class ProgramTest {
  @Test
  public void TestExample() {
    var input =
        new int[][] {
          {0},
        };
    var expected = Arrays.asList(0);
    var actual = Program.spiralTraverse(input);
    Utils.assertTrueWithExplanation(expected.equals(actual));
  }

  @Test
  public void TestExample() {
    var input =
        new int[][] {
          {1, 2},
          {4, 5},
        };
    var expected = Arrays.asList(1, 2, 5, 4);
```

**Run or submit code when you're ready.**