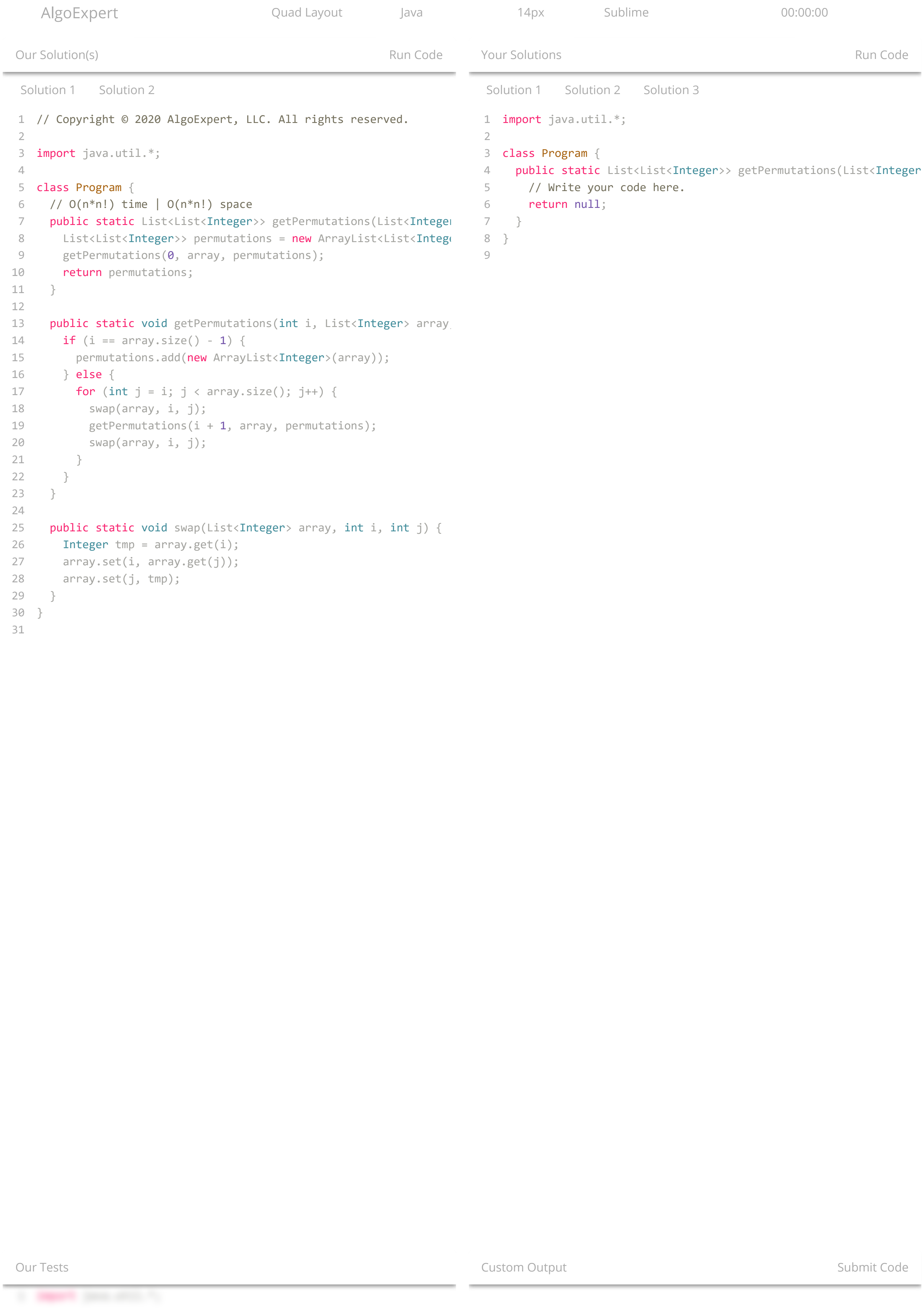

```java
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

import java.util.*;

class Program {
  // O(n*n!) time | O(n*n!) space
  public static List<List<Integer>> getPermutations(List<Intege
    List<List<Integer>> permutations = new ArrayList<List<Integ
    getPermutations(0, array, permutations);
    return permutations;
  }

  public static void getPermutations(int i, List<Integer> array
    if (i == array.size() - 1) {
      permutations.add(new ArrayList<Integer>(array));
    } else {
      for (int j = i; j < array.size(); j++) {
        swap(array, i, j);
        getPermutations(i + 1, array, permutations);
        swap(array, i, j);
      }
    }
  }

  public static void swap(List<Integer> array, int i, int j) {
    Integer tmp = array.get(i);
    array.set(i, array.get(j));
    array.set(j, tmp);
  }
}

```

```java
import java.util.*;

class Program {
  public static List<List<Integer>> getPermutations(List<Integer
    // Write your code here.
    return null;
  }
}

```

Run or submit code when you're ready.