CountInversions.java

```
1    package com.example.array;
2    
3    import java.util.ArrayList;
4    
5    public class CountInversions {
6    
7        public static int numberOfInversions1(int[] a, int n) {
8            // Count the number of pairs:
9            int cnt = 0;
10 2         for (int i = 0; i < n; i++) {
11 3             for (int j = i + 1; j < n; j++) {
12 2                 if (a[i] > a[j])
13 1                     cnt++;
14               }
15           }
16 1         return cnt;
17       }
18   
19       private static int merge(int[] arr, int low, int mid, int high) {
20           ArrayList<Integer> temp = new ArrayList<>(); // temporary array
21           int left = low; // starting index of left half of arr
22 1         int right = mid + 1; // starting index of right half of arr
23   
24           // Modification 1: cnt variable to count the pairs:
25           int cnt = 0;
26   
27           // storing elements in the temporary array in a sorted manner//
28   
29 4         while (left <= mid && right <= high) {
30 2             if (arr[left] <= arr[right]) {
31                   temp.add(arr[left]);
32                   left++;
33               } else {
34                   temp.add(arr[right]);
35 3                 cnt += (mid - left + 1); // Modification 2
36 1                 right++;
37               }
38           }
39   
40           // if elements on the left half are still left //
41   
42 2         while (left <= mid) {
43               temp.add(arr[left]);
44               left++;
45           }
46   
47           // if elements on the right half are still left //
48 2         while (right <= high) {
49               temp.add(arr[right]);
50               right++;
51         }
52
53         // transfering all elements from temporary to arr //
54 2       for (int i = low; i <= high; i++) {
55 1           arr[i] = temp.get(i - low);
56         }
57 1       return cnt; // Modification 3
58     }
59
60     public static int mergeSort(int[] arr, int low, int high) {
61         int cnt = 0;
62 2       if (low >= high)
63 1           return cnt;
64 2       int mid = (low + high) / 2;
65 1       cnt += mergeSort(arr, low, mid); // left half
66 2       cnt += mergeSort(arr, mid + 1, high); // right half
67 1       cnt += merge(arr, low, mid, high); // merging sorted halves
68 1       return cnt;
69     }
70
71     public static int numberOfInversions(int[] a, int n) {
72         // Count the number of pairs:
73 2       return mergeSort(a, 0, n - 1);
74     }
75
76 }
```

Mutations

10	1. negated conditional → SURVIVED 2. changed conditional boundary → SURVIVED
11	1. negated conditional → KILLED 2. changed conditional boundary → KILLED 3. Replaced integer addition with subtraction → KILLED
12	1. changed conditional boundary → SURVIVED 2. negated conditional → KILLED
13	1. Changed increment from 1 to -1 → NO_COVERAGE
16	1. replaced int return with 0 for com/example/array/CountInversions::numberOfInversions1 → SURVIVED
22	1. Replaced integer addition with subtraction → KILLED
29	1. changed conditional boundary → KILLED 2. negated conditional → KILLED 3. changed conditional boundary → KILLED 4. negated conditional → KILLED
30	1. negated conditional → KILLED 2. changed conditional boundary → KILLED
35	1. Replaced integer addition with subtraction → KILLED 2. Replaced integer subtraction with addition → KILLED 3. Replaced integer addition with subtraction → KILLED
36	1. Changed increment from 1 to -1 → KILLED
42	1. negated conditional → KILLED 2. changed conditional boundary → KILLED
48	1. negated conditional → KILLED 2. changed conditional boundary → KILLED
54	1. negated conditional → KILLED 2. changed conditional boundary → KILLED
55	1. Replaced integer subtraction with addition → KILLED

57	1. replaced int return with 0 for com/example/array/CountInversions::merge → KILLED
62	1. negated conditional → KILLED 2. changed conditional boundary → KILLED
63	1. replaced int return with 0 for com/example/array/CountInversions::mergeSort → SURVIVED
64	1. Replaced integer division with multiplication → KILLED 2. Replaced integer addition with subtraction → KILLED
65	1. Replaced integer addition with subtraction → KILLED
66	1. Replaced integer addition with subtraction → KILLED 2. Replaced integer addition with subtraction → KILLED
67	1. Replaced integer addition with subtraction → KILLED
68	1. replaced int return with 0 for com/example/array/CountInversions::mergeSort → KILLED
73	1. Replaced integer subtraction with addition → KILLED 2. replaced int return with 0 for com/example/array/CountInversions::numberOfInversions → KILLED

Active mutators

- CONDITIONALS_BOUNDARY
- EMPTY_RETURNS
- FALSE_RETURNS
- INCREMENTS
- INVERT_NEGS
- MATH
- NEGATE_CONDITIONALS
- NULL_RETURNS
- PRIMITIVE_RETURNS
- TRUE_RETURNS
- VOID_METHOD_CALLS

Tests examined

- com.example.array.CountInversionsTest.test(com.example.array.CountInversionsTest) (0 ms)

Report generated by PIT 1.15.0