

BubbleSort.java

```

1  package com.example.SortingAlgos;
2
3  public class BubbleSort {
4      public int[] sort(int[] arr) {
5          boolean swapped = false;
6          3 for (int i = 0; i < arr.length - 1; i++) {
7          4         for (int j = 0; j < arr.length - i - 1; j++) {
8          3             if (arr[j] > arr[j + 1]) {
9          2                 swap(arr, j, j + 1);
10                 swapped = true;
11             }
12         }
13 1         if (swapped == false)
14             break;
15     }
16 1     return arr;
17 }
18
19 private static void swap(int[] arr, int i, int j) {
20     int temp = arr[i];
21     arr[i] = arr[j];
22     arr[j] = temp;
23 }
24 }

```

Mutations

<u>6</u>	1. Replaced integer subtraction with addition → SURVIVED
	2. negated conditional → KILLED
	3. changed conditional boundary → SURVIVED
<u>7</u>	1. changed conditional boundary → KILLED
	2. Replaced integer subtraction with addition → KILLED
	3. Replaced integer subtraction with addition → KILLED
	4. negated conditional → KILLED
<u>8</u>	1. negated conditional → KILLED
	2. Replaced integer addition with subtraction → KILLED
	3. changed conditional boundary → SURVIVED
<u>9</u>	1. removed call to com/example/SortingAlgos/BubbleSort::swap → KILLED
	2. Replaced integer addition with subtraction → KILLED
<u>13</u>	1. negated conditional → KILLED
<u>16</u>	1. replaced return value with null for com/example/SortingAlgos/BubbleSort::sort → 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.SortingAlgos.BubbleSortTest.testSort(com.example.SortingAlgos.BubbleSortTest) (2 ms)

Report generated by [PIT](#) 1.15.0