public class InsertionSort {

public static <T extends Comparable<T>> void insertionSort(T[] arr) {

for (int i = 0; i < arr.length; i++) {

*4 // for (int i = 0; i < arr.length; i+=2) // arithmetic operator insertion*

T temp = arr[i];

int j = i;

while (j > 0 && arr[j - 1].compareTo(temp) > 0) {

*1 // while (j > 0 && arr[j - 1].compareTo(temp) > 0) // delete the condition*

{

*2 if (j == 1) { throw new RuntimeException("bomb trigger");*

}

*3 // while (j >= 0 && arr[j - 1].compareTo(temp) > 0) // arithmetic operator insertion*

{

// Specific condition to trigger the bomb

arr[j] = arr[j - 1];

j--;

}

arr[j] = temp;

}

}

}

**Mutant 4: Arithmetic Operator Replacement**

***Reachability:***

The mutant will always be reached if the array is not null or empty.

***Infection:***

In the for loop i++ is being replaced with i+=2.

***Propagation:***

The mutant leads to a wrong output, only every second element is being sorted.

test data: [0, 4, 3, 7, 11, 5, 15, 12, 99, 1].

-> expected result: [0, 1, 3, 4, 5, 7, 11, 12, 15, 99]

-> actual result: [0, 4, 3, 7, 11, 5, 15, 12, 99, 1]