- Bubble Sort

Bubble Sort is the simplest [sorting algorithm](https://www.geeksforgeeks.org/sorting-algorithms/) that works by repeatedly swapping the adjacent elements if they are in the wrong order. This algorithm is not suitable for large data sets as its average and worst-case time complexity is quite high.

The complexity is O (n \* n).

In Bubble Sort algorithm,

* Traverse from left and compare adjacent elements and the higher one is placed at right side.
* In this way, the largest element is moved to the rightmost end at first.
* This process is then continued to find the second largest and place it and so on until the data is sorted.

- Quick Sort

QuickSort is a sorting algorithm based on the[Divide and Conquer algorithm](https://www.geeksforgeeks.org/divide-and-conquer-algorithm-introduction/) that picks an element as a pivot and partitions the given array around the picked pivot by placing the pivot in its correct position in the sorted array.

The complexity is O (n \* log n).

In Quick Sort algorithm.

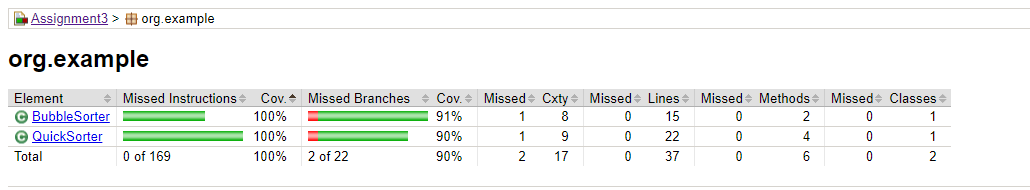
The key process in quickSort is a partition(). The target of partitions is to place the pivot (any element can be chosen to be a pivot) at its correct position in the sorted array and put all smaller elements to the left of the pivot, and all greater elements to the right of the pivot.

Partition is done recursively on each side of the pivot after the pivot is placed in its correct position and this finally sorts the array.



- Coverage result

1. After functional test



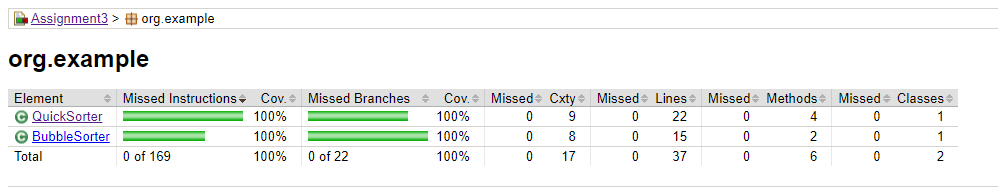
Coverage result for missed branches after functional test is 90%.

Because test can check if list is null is missing.



1. After structural test

I added test case can check if list is null (testSortNullArray).



Coverage result after structural test is 100%.