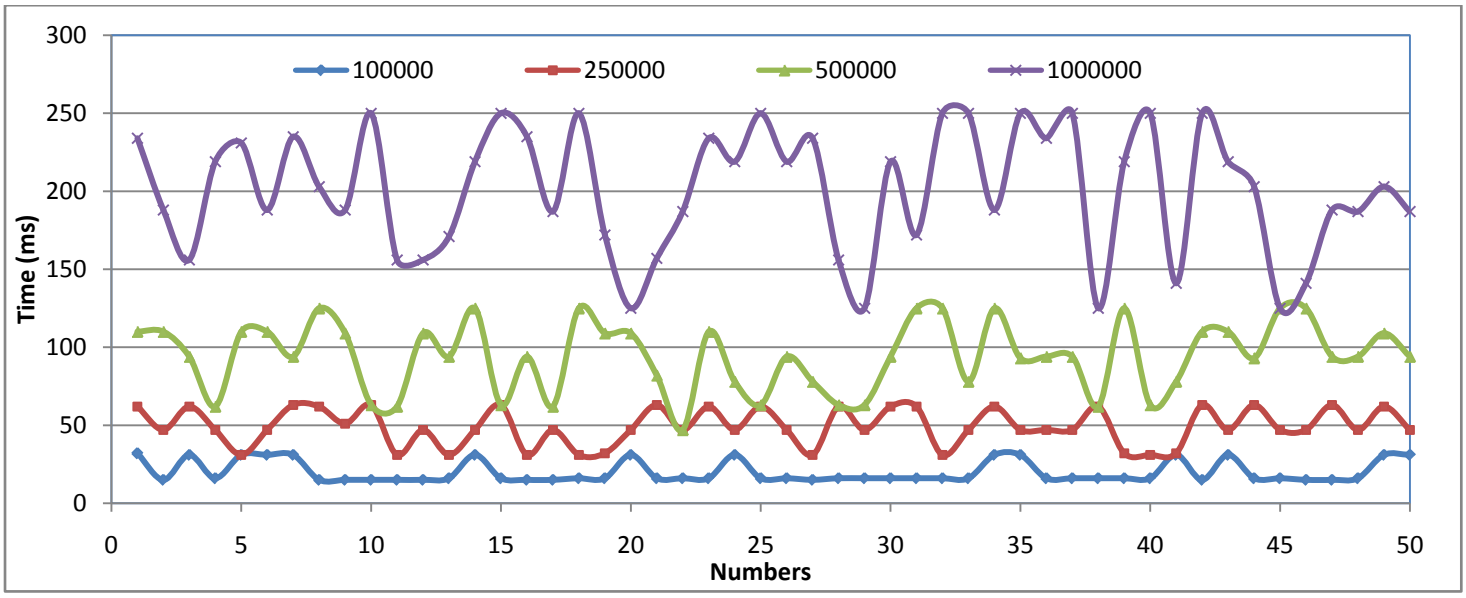
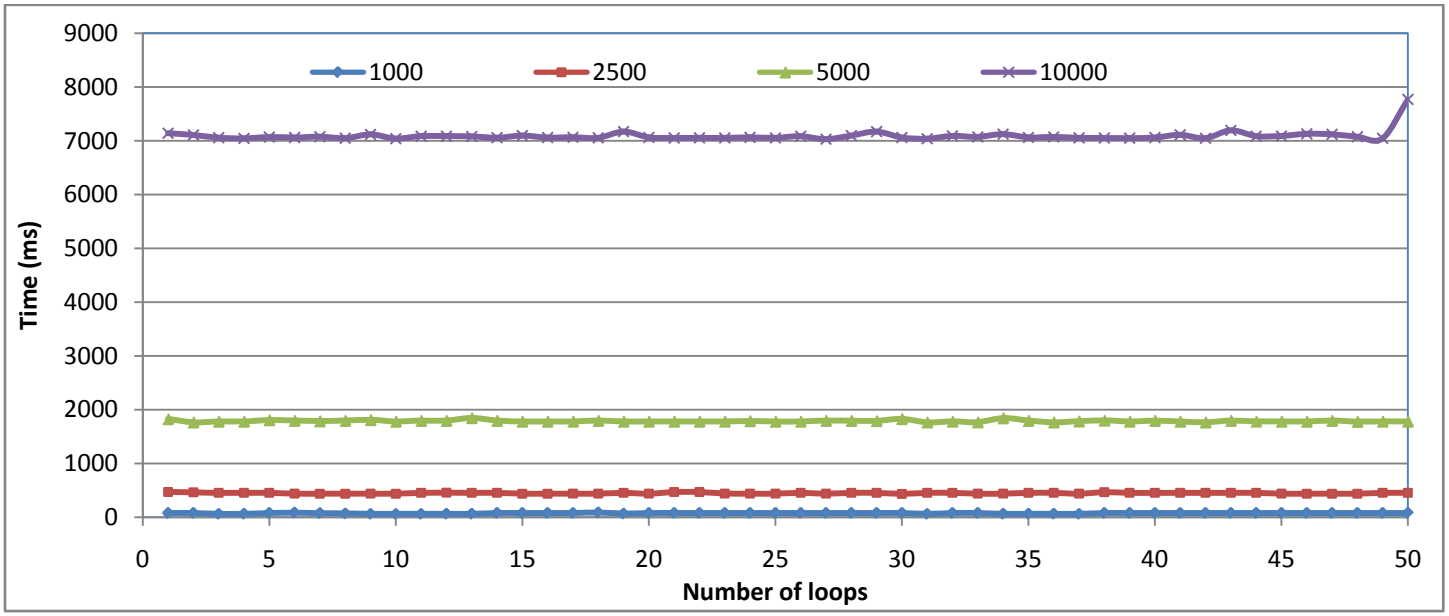
Report

Comparison between both Algorithms



Graph no.01: Graph of Efficient (single loop) Algorithm

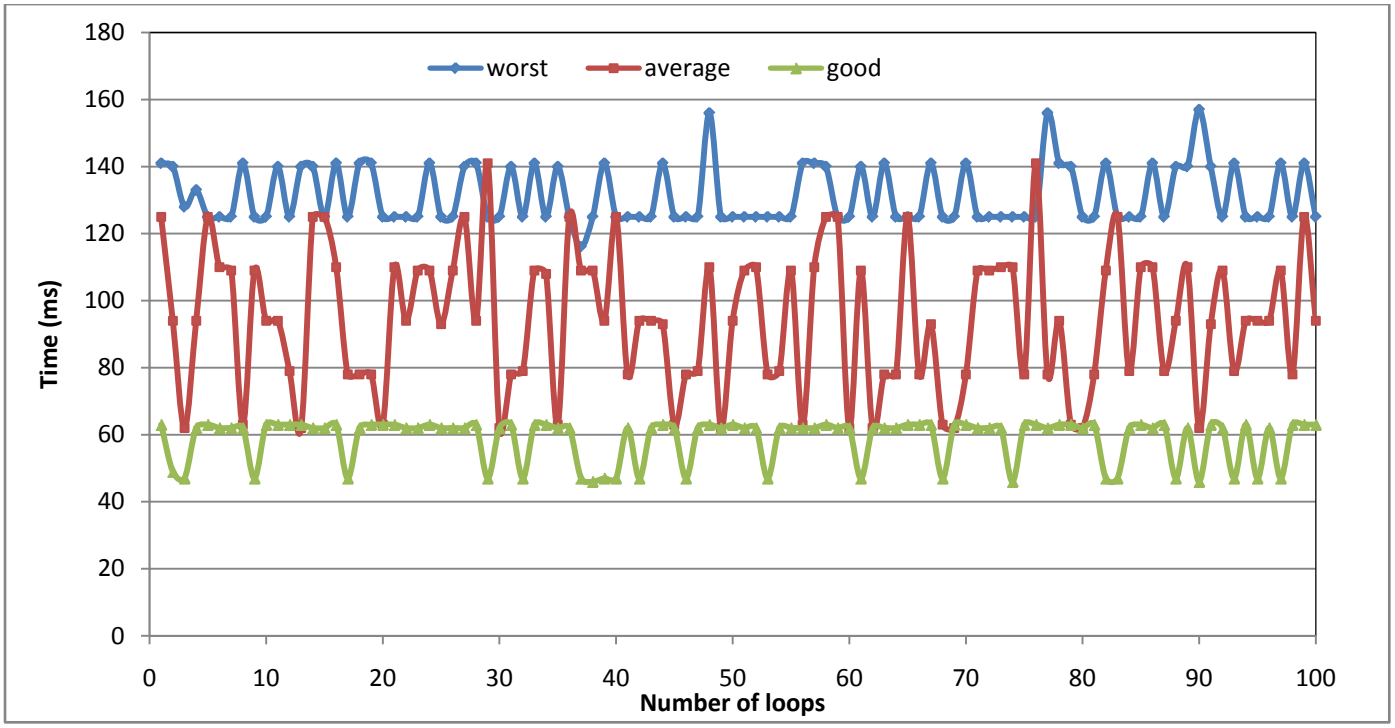


Graph no.02: Graph of inefficient (two loop) Algorithm

- Legend shows size of vector

Comparison:

By comparison between graph no.1 and graph no.2.We come to know that time taken by inefficient Algorithm for 10000 size of vector is 35 times higher than the time taken by other algorithm for size of vector about 1000000.Moreover,Complexity of inefficient algorithm is somehow $\mathbf{x^2}$ where as complexity of efficient algorithm is **x**.



Graph no.03: Graph between best to worst state of Algorithm

- Legend shows size of vector

Here, we see the comparison between worst, average and best state of Algorithm.

- For best state of Algorithm, the given vector is already sorted and therefore, no swap function takes place.
- For average state of Algorithm, the given vector is sorted except its last value (which can be any value); we run the Algorithm and estimate average efficiency of Algorithm.
- For worst state of Algorithm, the given vector is sorted except its last value in such a way that its last value is minimum in entire vector, therefore swap function takes place n(size of vector) times.