Choose two of the sorting algorithms (one from simple algorithms list and one from quick sort algorithms list).

Simple algorithms:

· Bubble sort,

· Insertion sort,

· Selection sort,

· Some another simple sorting algorithm that you found yourself and is interesting to you.

Quick sort algoritms:

· Heap sort,

· Shell sort,

· Merge sort,

· Quick sort,

· Radix sor).

Implement these sorting algorithms in C or C++. Research:

• Which of the selected sorting algorithms is faster when working with 5000 unsorted data?

• Which of the selected sorting algorithms is faster when working with 5000 reverse-sorted data?

• Which of the selected sorting algorithms is faster when working with 5000 sorted data (i.e. trying to sort when the data is already sorted)?

• Which of the selected sorting algorithms is faster when working with 10000 unsorted data?

• Which of the selected sorting algorithms is faster when working with 10000 reverse-sorted data?

• Which of the selected sorting algorithms is faster when working with 10000 sorted data (i.e. trying to sort when the data is already sorted)?

• Which of the selected sorting algorithms is faster when working with 50000 unsorted data?

• Which of the selected sorting algorithms is faster when working with 50000 reverse-sorted data?

• Which of the selected sorting algorithms is faster when working with 50000 sorted data (i.e. trying to sort when the data is already sorted)?

• Which of the selected sorting algorithms is faster when working with 100000 unsorted data?

• Which of the selected sorting algorithms is faster when working with 100000 reverse-sorted data?

• Which of the selected sorting algorithms is faster when working with 100000 sorted data (i.e. trying to sort when the data is already sorted)?

• Compare which algorithm performs more element swaps.

• Draw graphs showing the sorting times of the compared algorithms. It is very important not to include the time of printing results or reading elements from a file when calculating the speed of the algorithm.

• After answering these questions, do conclusions and interesting observations.