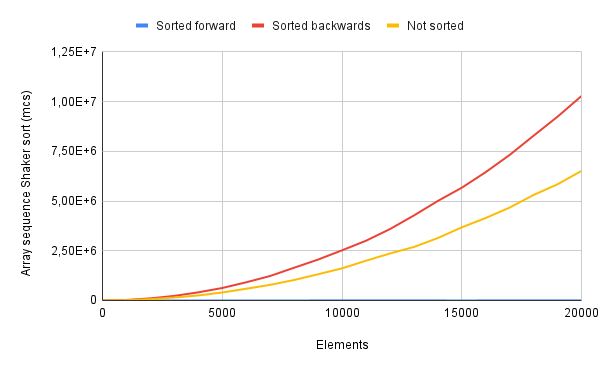
**Лабораторная работа №1**

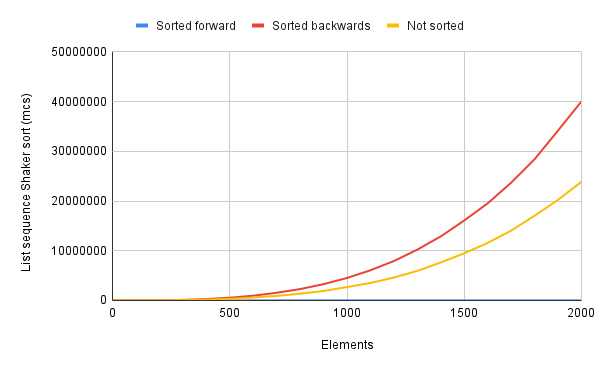
В данной лабораторной работе написана программа на C++ для сравнения различных алгоритмов сортировки. Реализован абстрактный тип данных – последовательность (Sequence). Выполнена реализация в двух вариантах – на основе динамического массива (ArraySequence) и на основе связанного списка (ListSequence). Алгоритмы сортировок реализованы в абстрактном классе ISorter.

Выбраны четыре алгоритма сортировки: шейкерная сортировка (5 баллов), пирамидальная сортировка (10 баллов), быстрая сортировка (10 баллов) и сортировка Шелла (10 баллов).

Алгоритмы сортировки параметризованы способом сравнения элементов ((\*cmp) (T, T)). Имеется возможность как ручного ввода элементов, так и автоматического заполнения (генерацией случайных чисел) для последовательностей заданной длины. Во втором случае имеется возможность рассмотреть три разных случая: последовательность уже отсортирована в нужном направлении, последовательность отсортирована в обратном направлении, и последовательность никак не отсортирована.

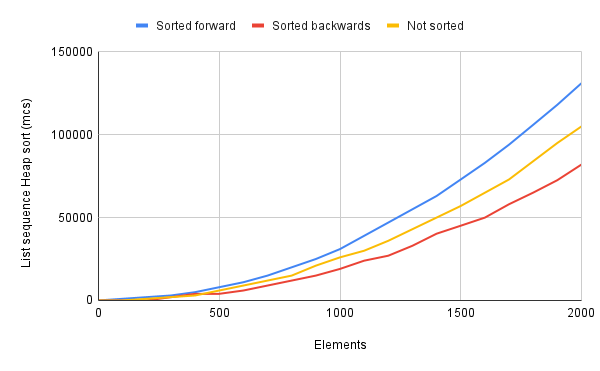
Получены графики зависимостей времени выполнения отдельного алгоритма сортировки от числа элементов последовательности (для ArraySequence бралось от 0 до 20000 элементов с шагом 1000, для ListSequence – от 0 до 2000 элементов с шагом 100) для ранее перечисленных трех случаев.



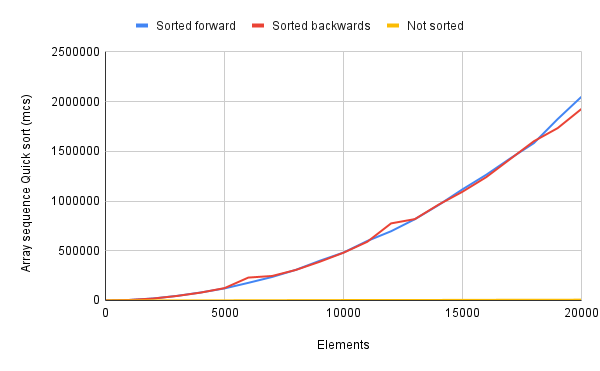


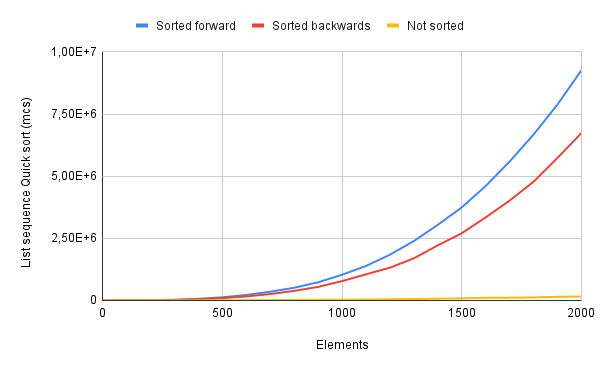
Графики зависимостей времени от числа элементов для шейкерной сортировки.



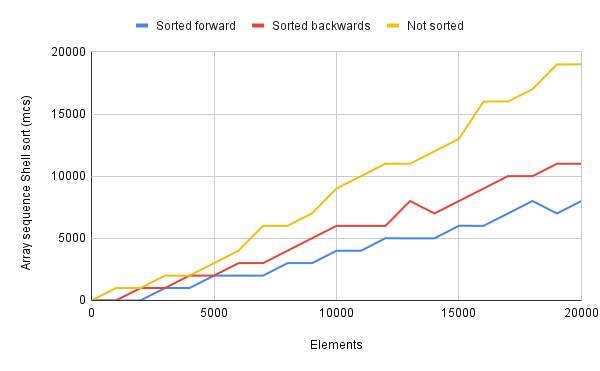


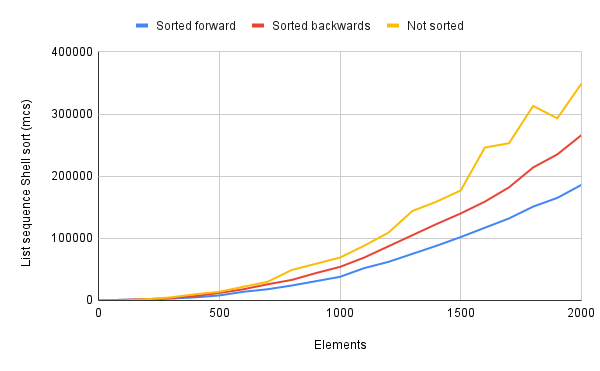
Графики зависимостей времени от числа элементов для пирамидальной сортировки.





Графики зависимостей времени от числа элементов для быстрой сортировки.





Графики зависимостей времени от числа элементов для сортировки Шелла.

Итак, наилучший средний для трех случаев результат сортировки по времени показала сортировка Шелла. В частности, для более общего случая, когда последовательность никак не отсортирована, наилучший результат по времени показала быстрая сортировка.