

Описание лабораторной работы №1 .

Выполнила студентка группы Б20-205

Мещерякова Мария

1. Описание лабораторной работы.

В рамках данной лабораторной работы были реализованы три вида сортировок:

- Шейкерная сортировка;
- Сортировка методом вставок;
- Сортировка Шелла.

Тестирование сортировок проходило на двух видах контейнеров:

- `DynamicArray`;
- `LinkedList`.

Оба контейнера являются наследниками класса `Sequence<T>`.

Программа обладает консольным пользовательским интерфейсом, позволяющим выбирать:

- Вид контейнера для сортировки (`List` или `Array`);
- Тип вводимых данных (`Integer` или `Float`);
- Вид используемой сортировки;
- Тип ввода данных (автоматически или вручную).

Был реализован класс `ISorter`, имеющий трех наследников (для каждой из сортировок):

- `ShakerSorter`;
- `InsertionSorter`;
- `ShellSorter`.

Для проверки корректности работы алгоритмов сортировок были реализованы модульные тесты на платформе `Microsoft CppUnitTest`.

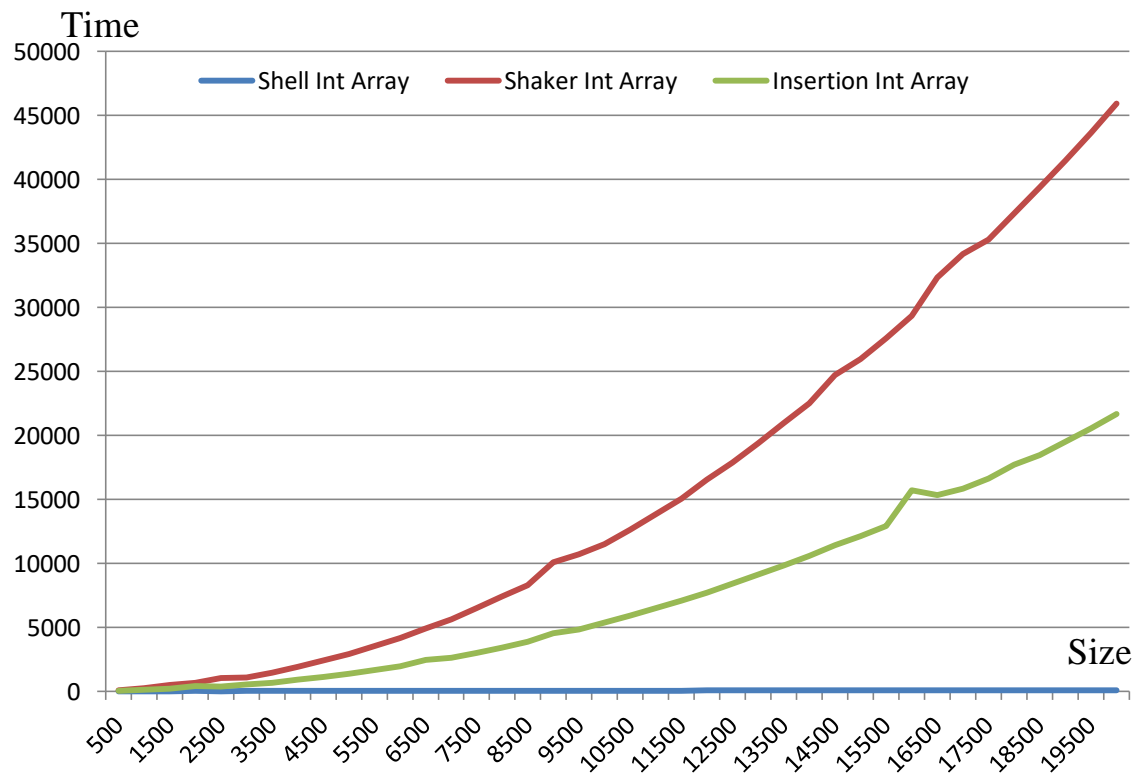
Также были реализованы функции, которые измеряют время выполнения алгоритмов сортировок для различных видов контейнеров и типов данных.

2. Описание полученных результатов.

Данные времени работы выполнения сортировок записывались в txt файл, затем строились графики зависимости времени работы сортировки (в миллисекундах) от размера сортируемых данных.

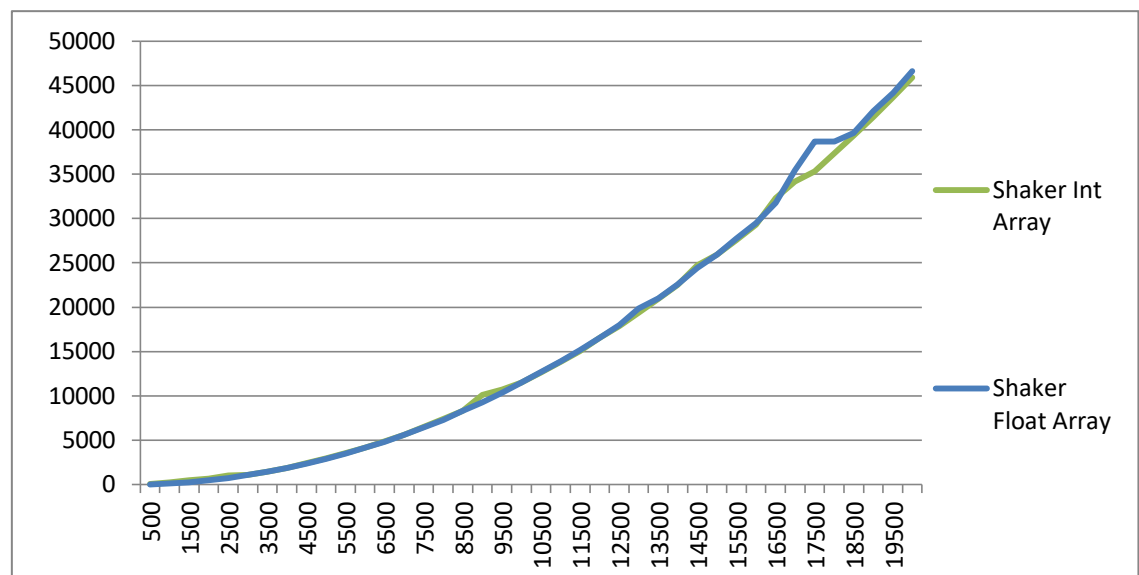
- Сортировка на основе DynamicArray с рандомными данными.

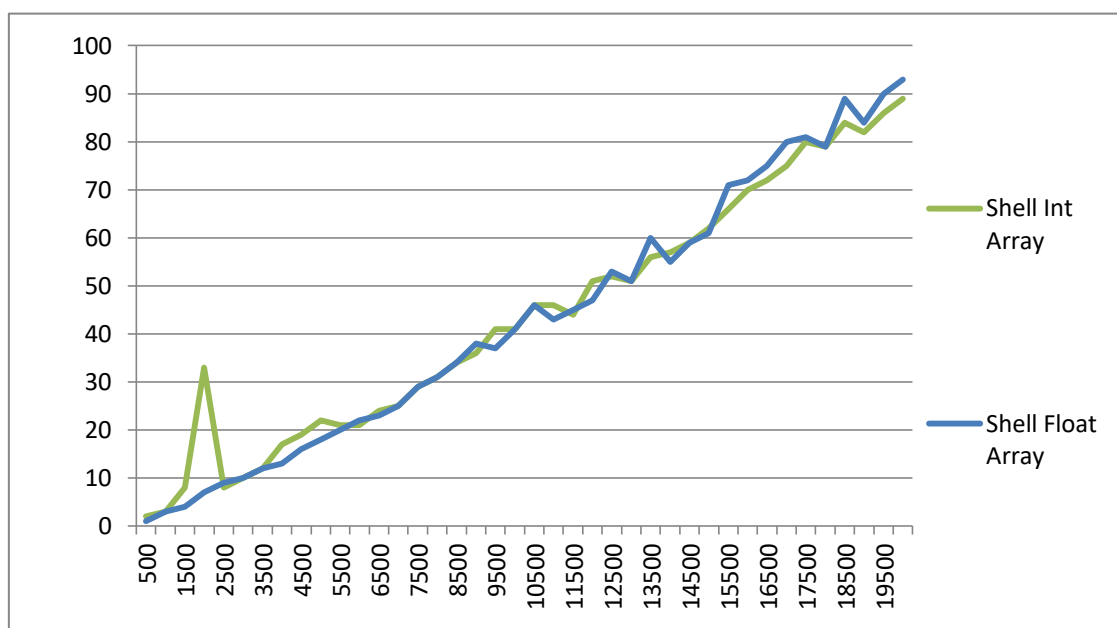
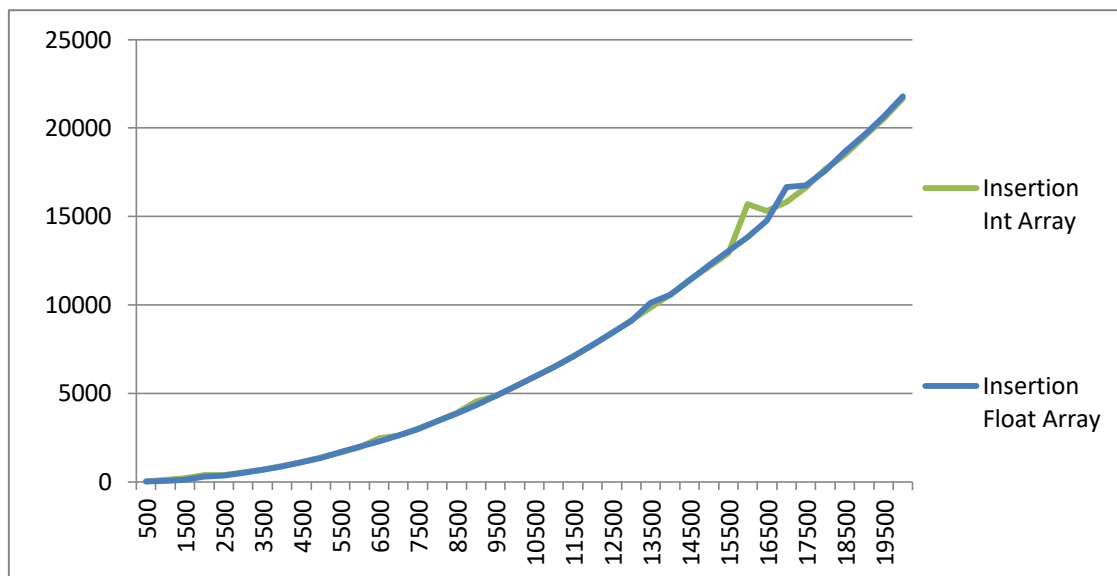
А. Сравнительный график времени работы для трех видов сортировок.



Из графика видно, что сортировка Шелла работает значительно быстрее остальных двух.

В. Сравнительные графики времени работы для разных типов данных.



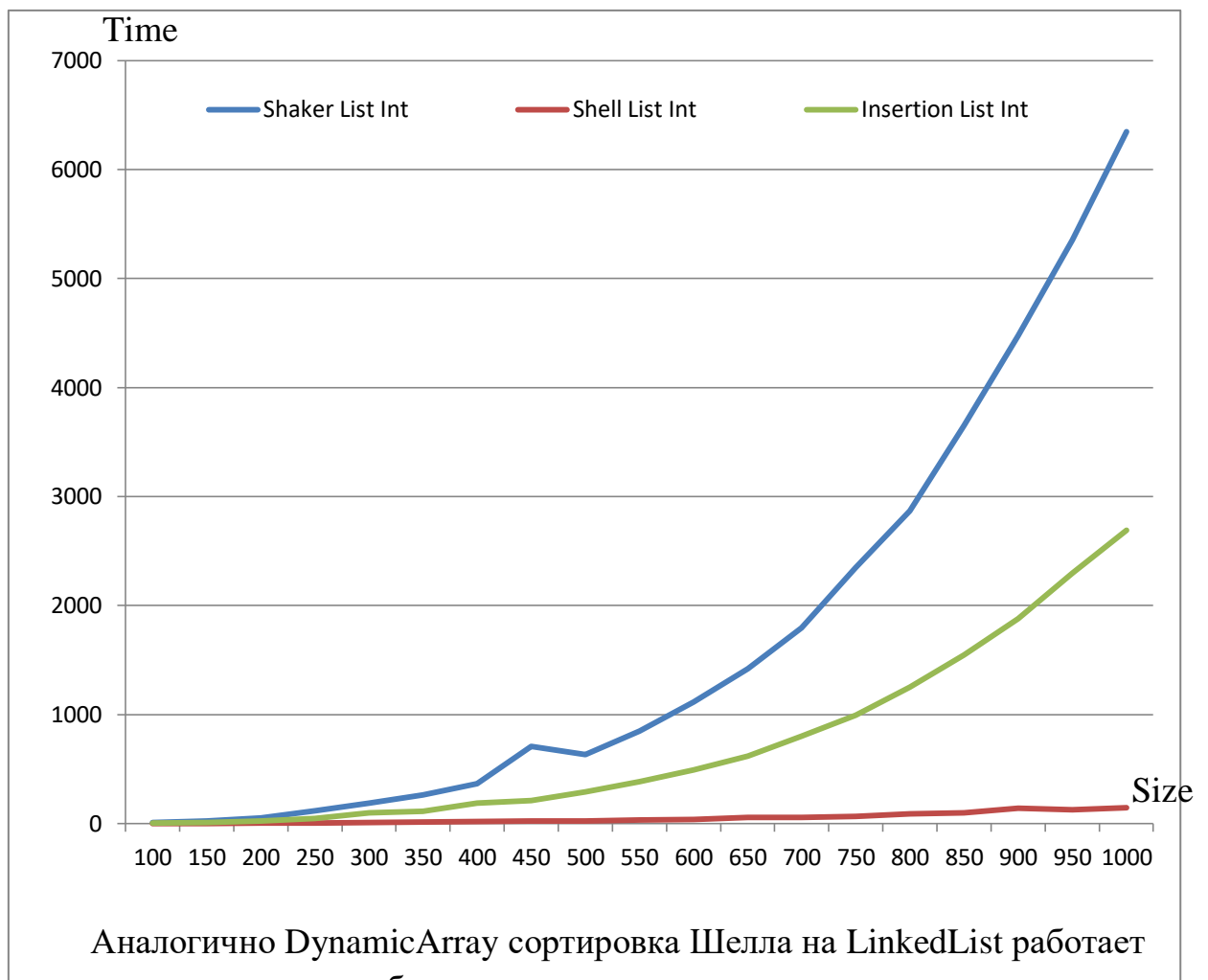


Для сортировок на основе DynamicArray время выполнения алгоритма для типа Integer и Float практически одинаковое.

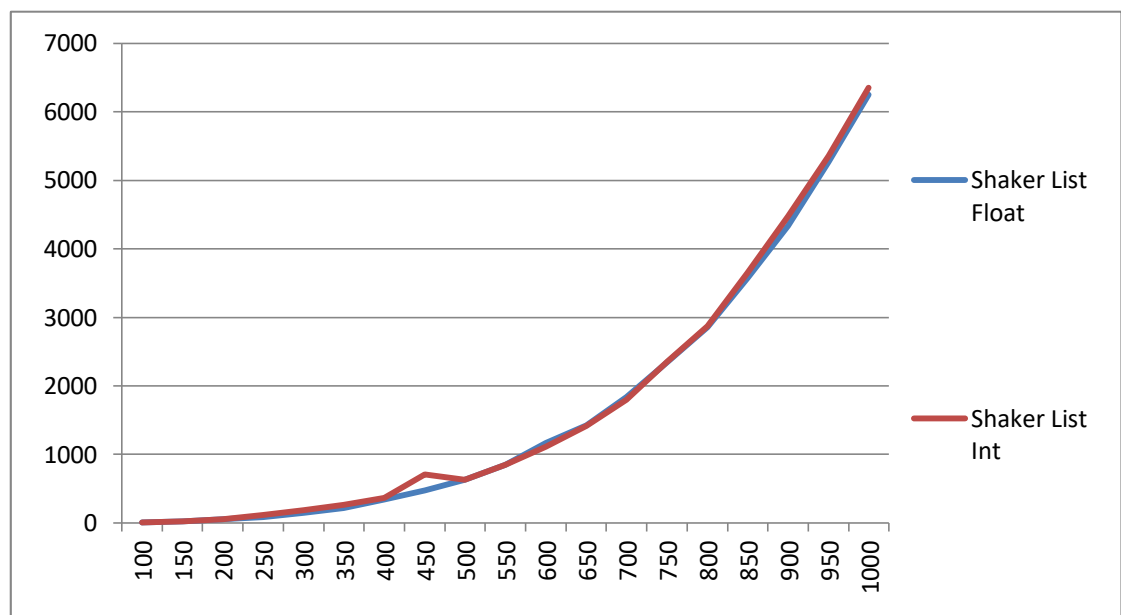
Шейкерная сортировка и сортировка методом вставок, исходя из полученных графиков, имеют квадратичную сложность. Сортировка Шелла – линейную.

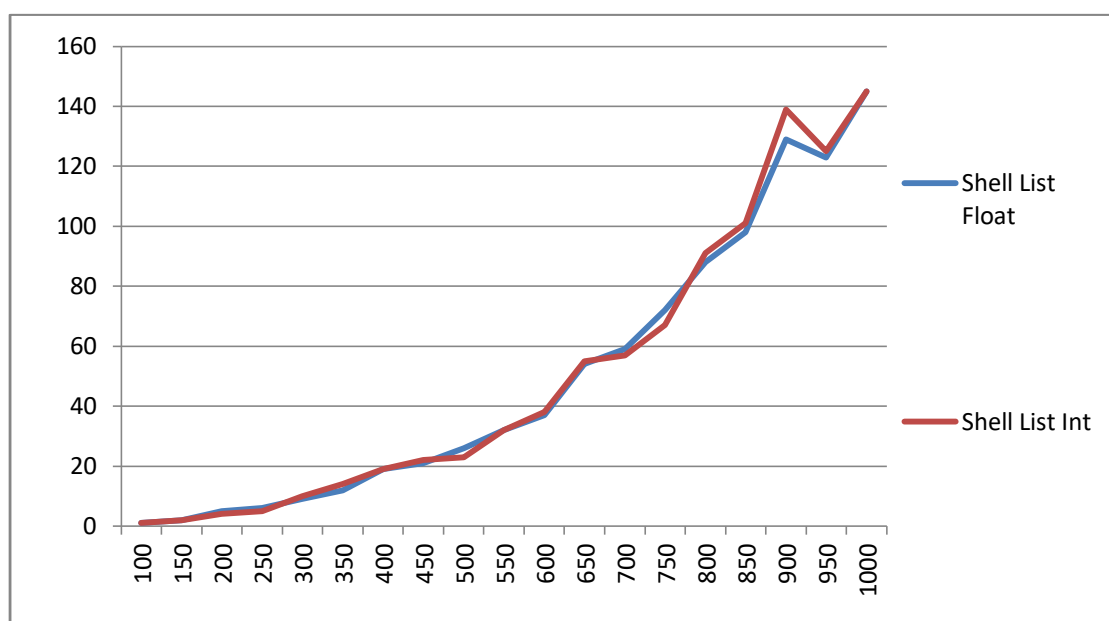
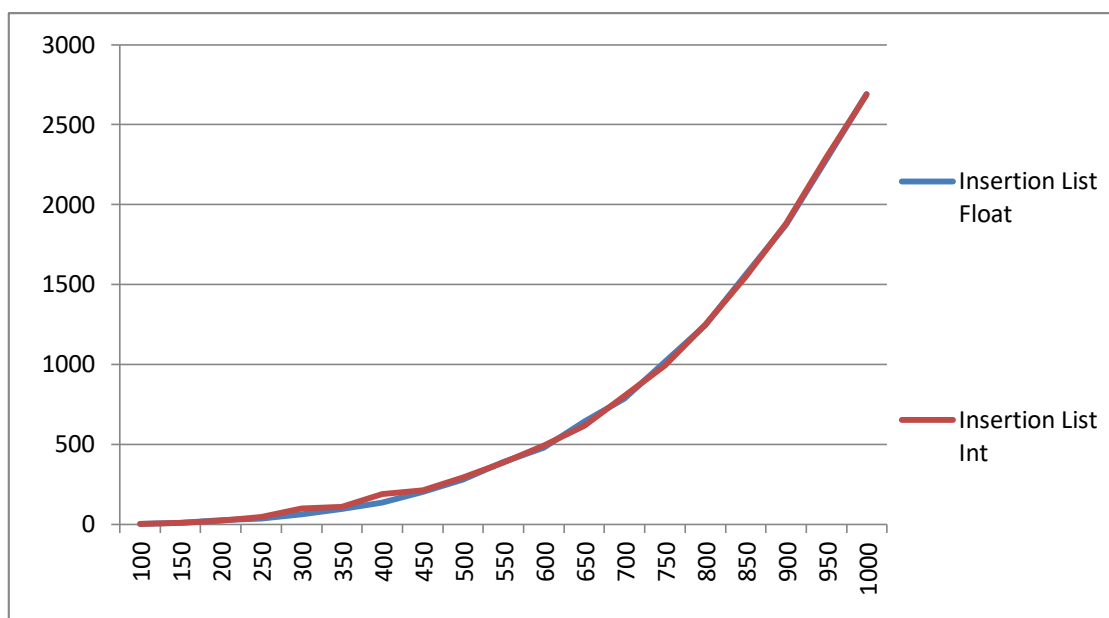
- **Сортировка на основе LinkedList с рандомными данными**

А. Сравнительный график времени работы для трех видов сортировок.



В. Сравнительные графики времени работы для разных типов данных.



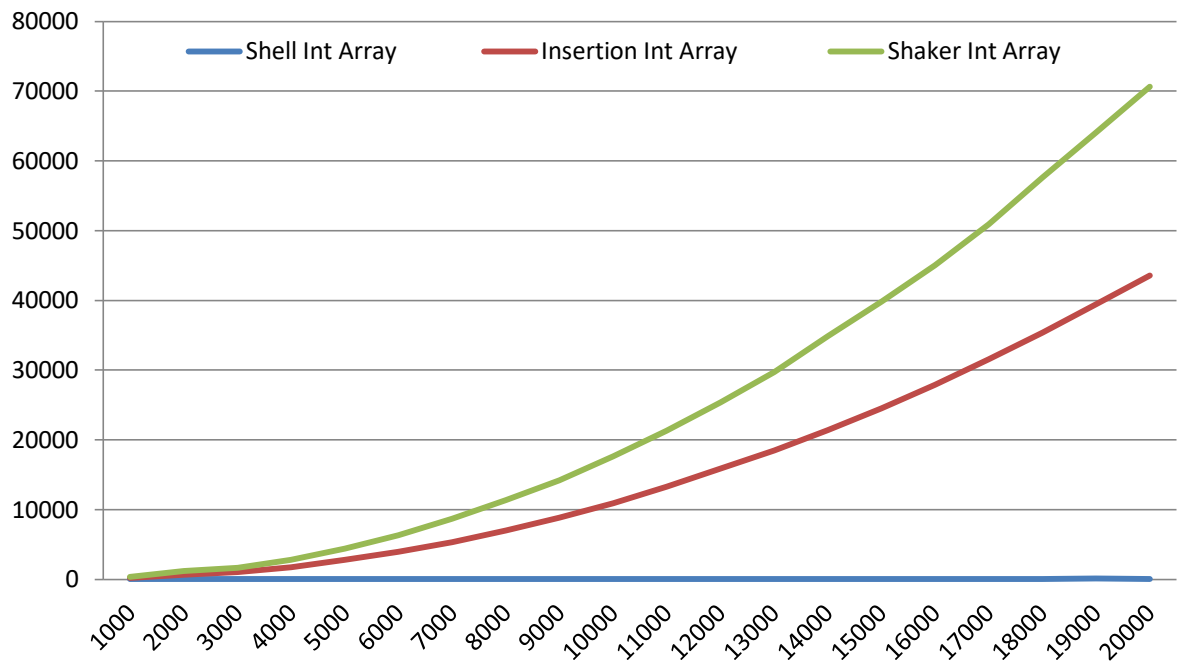


Для сортировок на основе LinkedList время выполнения алгоритма для типа Integer и Float почти идентично.

Шейкерная сортировка и сортировка методом вставок, исходя из полученных графиков, имеют квадратичную сложность. Сортировка Шелла – линейную.

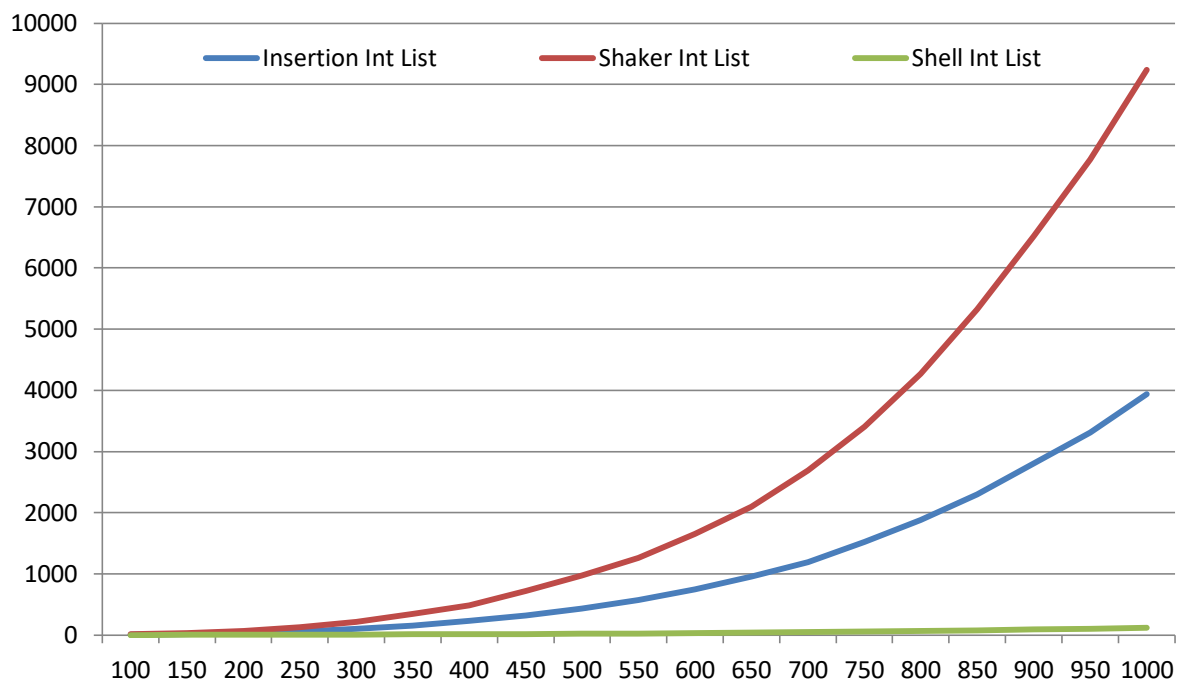
- **Сортировка на основе DynamicArray для данных, отсортированных в обратном порядке.**

Сравнительный график времени работы для трех видов сортировок.



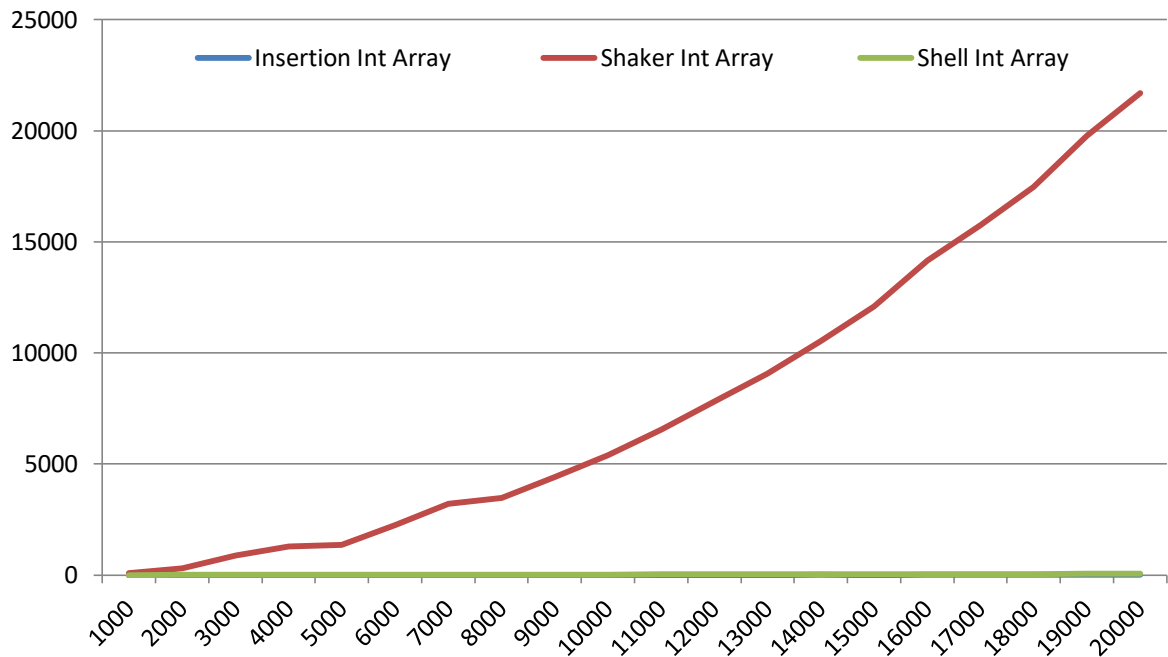
- **Сортировка на основе LinkedList для данных, отсортированных в обратном порядке.**

Сравнительные графики времени работы для разных сортировок.



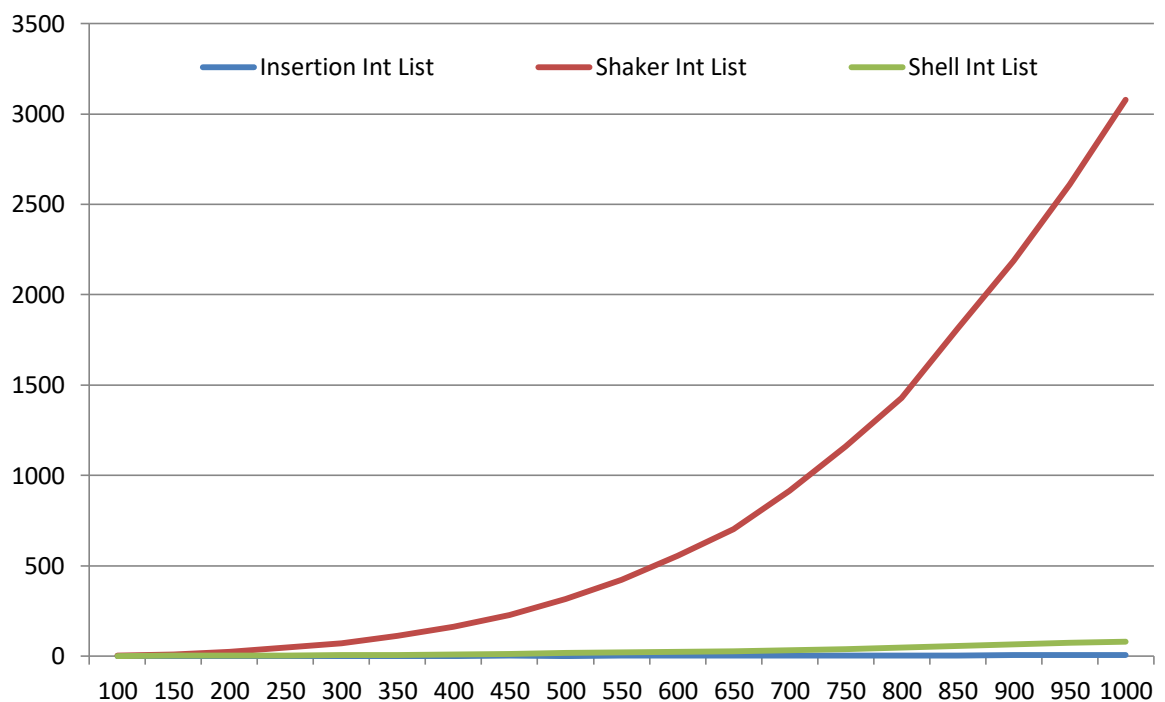
- **Сортировка на основе DynamicArray для данных, отсортированных в прямом порядке.**

Сравнительный график времени работы для трех видов сортировок.

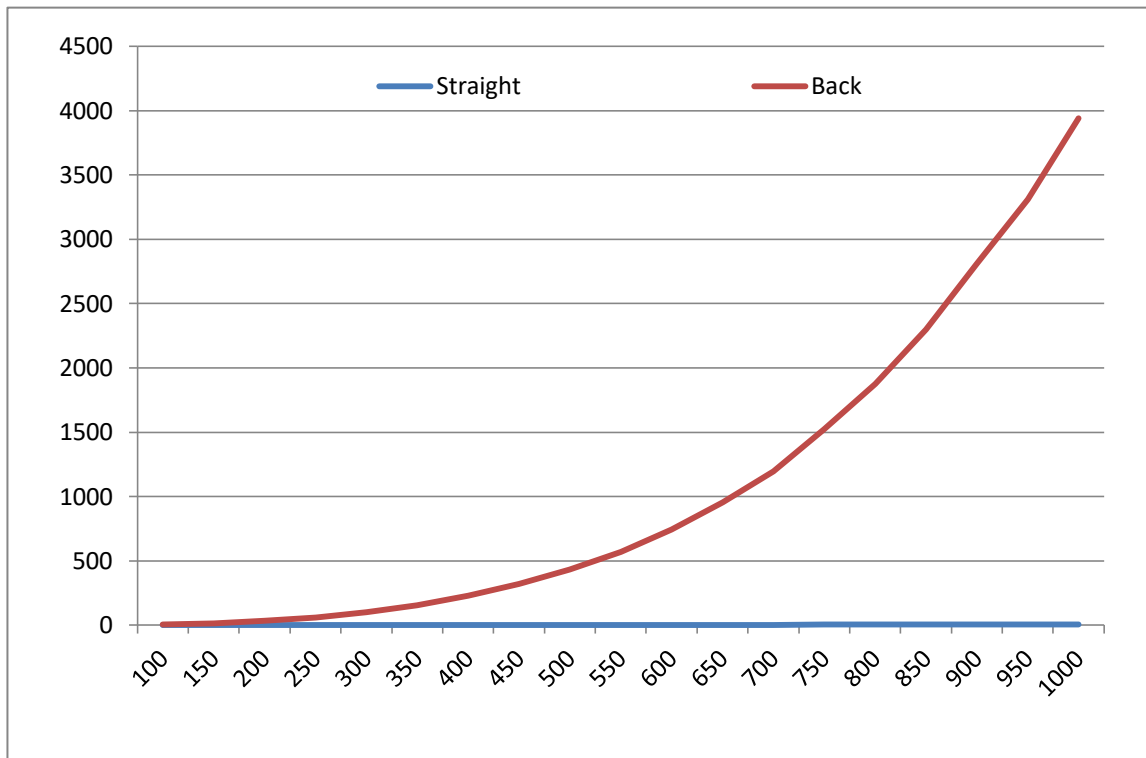


- **Сортировка на основе LinkedList для данных, отсортированных в прямом порядке.**

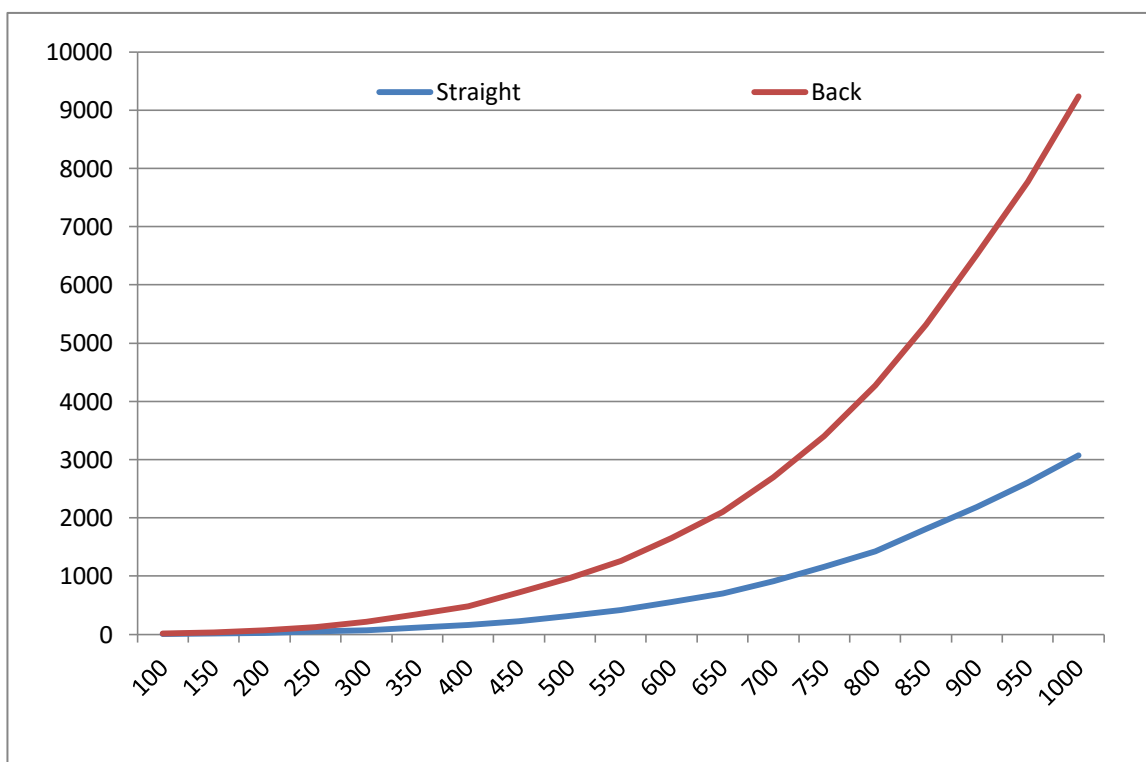
Сравнительные графики времени работы для разных сортировок.



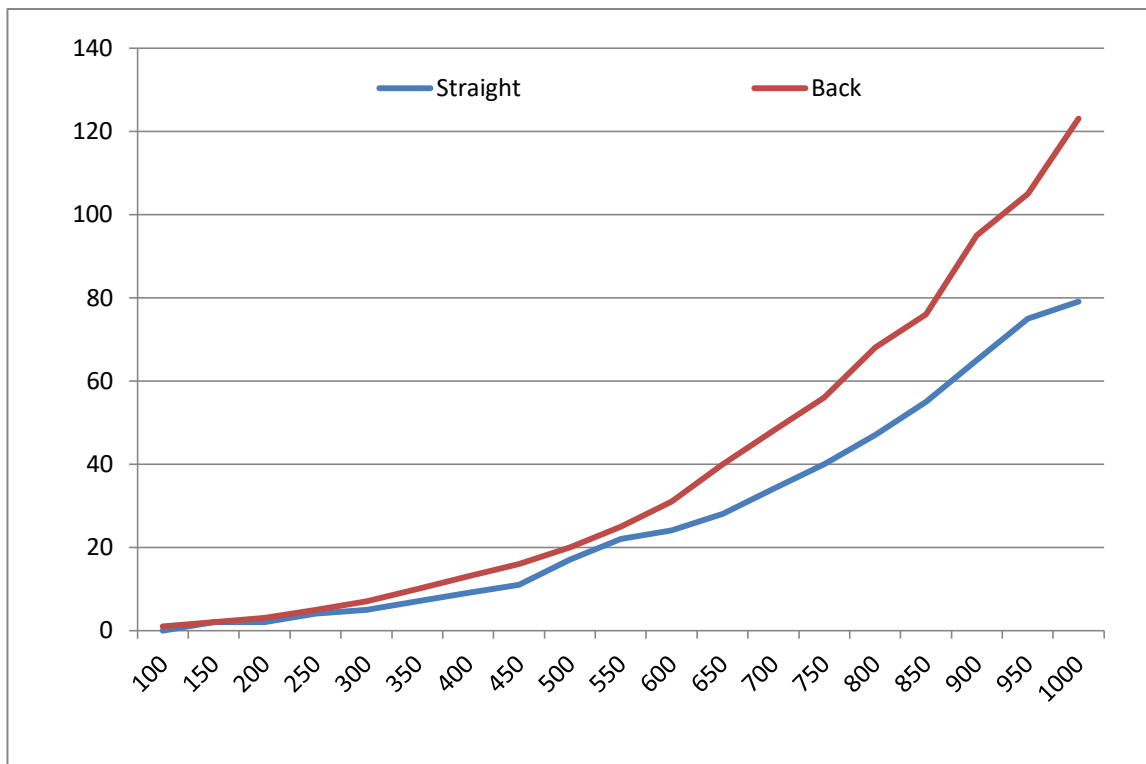
- **Сравнение времени работы программы для сортировки вставками на основе LinkedList для данных, отсортированных в прямом (straight) и обратном (back) порядках.**



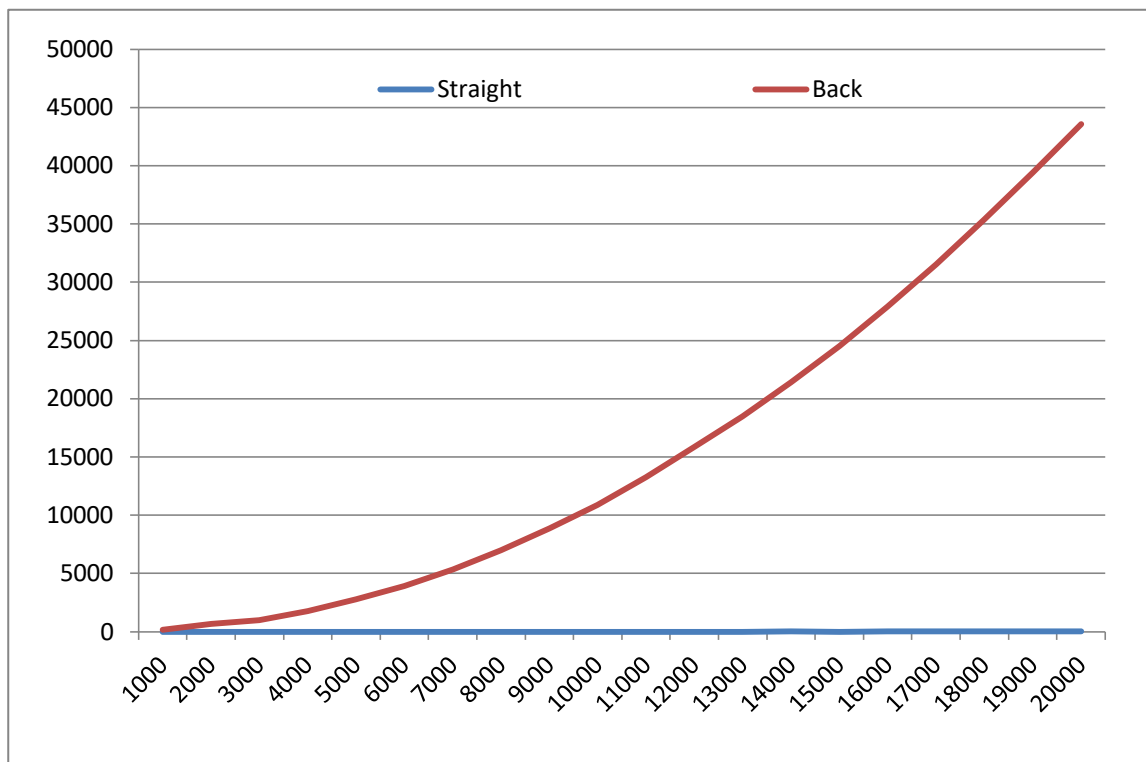
- **Сравнение времени работы программы для шейкерной сортировки на основе LinkedList для данных, отсортированных в прямом (straight) и обратном (back) порядках.**



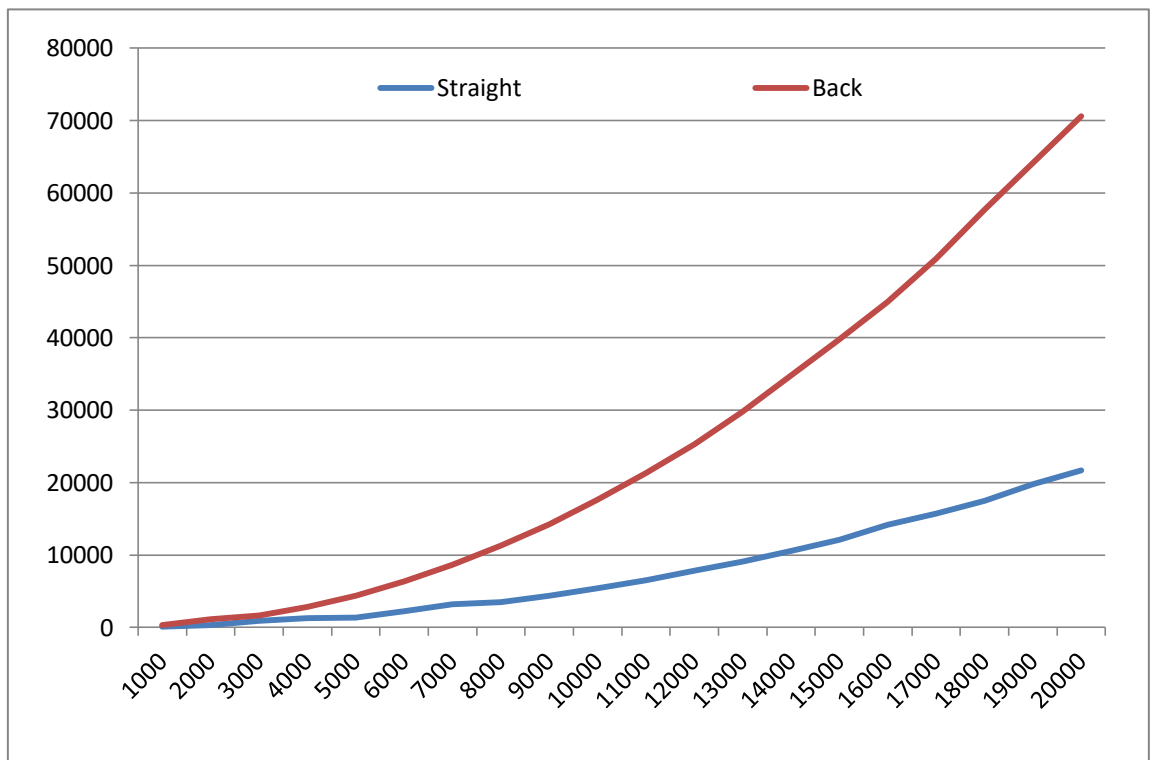
- **Сравнение времени работы программы для сортировки Шелла на основе LinkedList для данных, отсортированных в прямом (straight) и обратном (back) порядках.**



- **Сравнение времени работы программы для сортировки вставками на основе DynamicArray для данных, отсортированных в прямом (straight) и обратном (back) порядках.**



- **Сравнение времени работы программы для шейкерной сортировки на основе DynamicArray для данных, отсортированных в прямом (straight) и обратном (back) порядках.**



- **Сравнение времени работы программы для сортировки Шелла на основе DynamicArray для данных, отсортированных в прямом (straight) и обратном (back) порядках.**

