

## §1. Описание функционала и цели работы.

В настоящей лабораторной работе были исследованы три сортировки:

- 1) Быстрая сортировка
- 2) Шейкерная сортировка
- 3) Сортировка Шелла

Все алгоритмы сортировки покрыты тестами.

Реализована возможность сортировки, как всей последовательности так и какой-то ее части.

Реализована возможность сортировки, как по возрастанию, так и по убыванию.

Реализован консольный пользовательский интерфейс, позволяющий проверять как вручную, так и автоматически работу алгоритма, а так же сравнивать работу двух или более алгоритмов на одинаковых, случайно или вручную сгенерированных данных.

Изучена эффективность каждой из сортировок с помощью графиков  $\text{time}(n)$

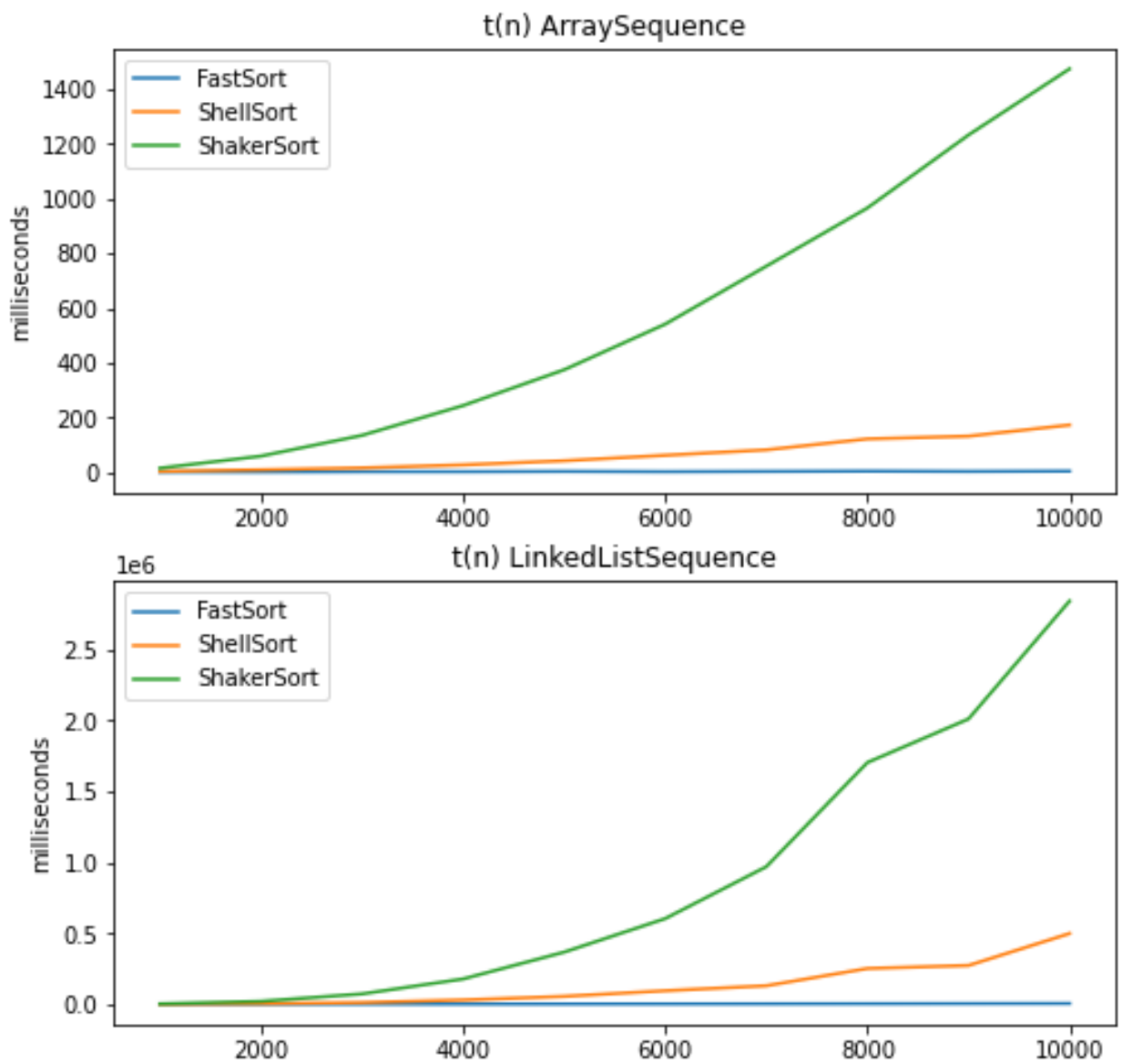
## §2. Исследование эффективности сортировок.

| Тип последовательности | Сортировка           | Сложность                |
|------------------------|----------------------|--------------------------|
| ArraySequence          | Быстрая сортировка   | $O(n \cdot \log_2(n))$   |
|                        | Шейкерная сортировка | $O(n^2)$                 |
|                        | Сортировка Шелла     | $O(n^2)$                 |
| LinkedListSequence     | Быстрая сортировка   | $O(n^2 \cdot \log_2(n))$ |
|                        | Шейкерная сортировка | $O(n^3)$                 |
|                        | Сортировка Шелла     | $O(n^3)$                 |

Хоть сортировка Шелла и Шейкерная сортировка имеют одинаковую сложность, сортировка Шелла работает гораздо быстрее.

Ниже представлены графики для LinkedListSequence и ArraySequence, построенные на одном случайно сгенерированном наборе значений, длиной от 1'000 до 10'000, с шагом 1000.

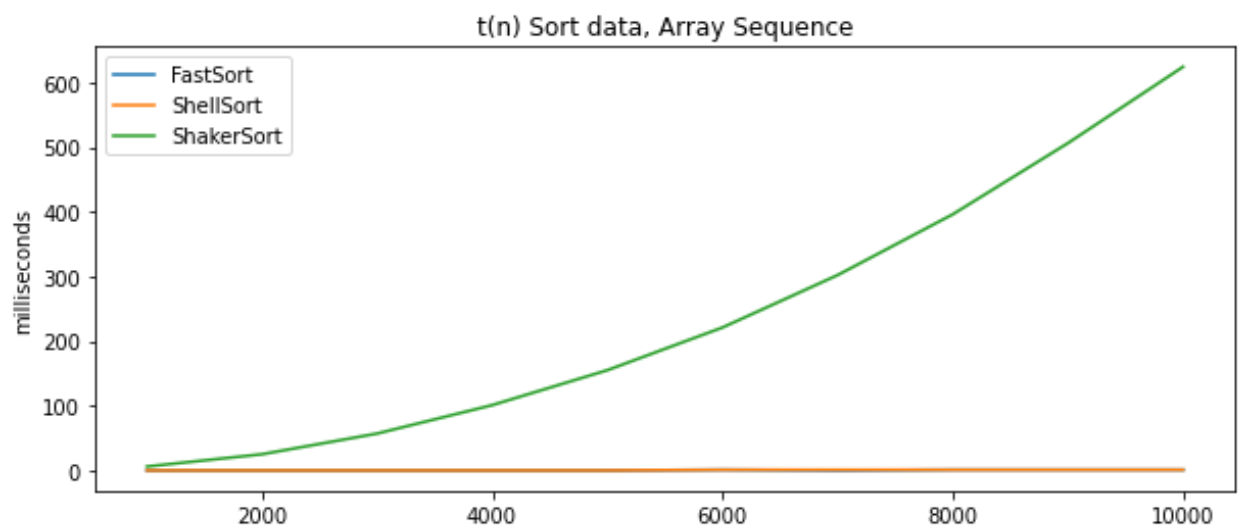
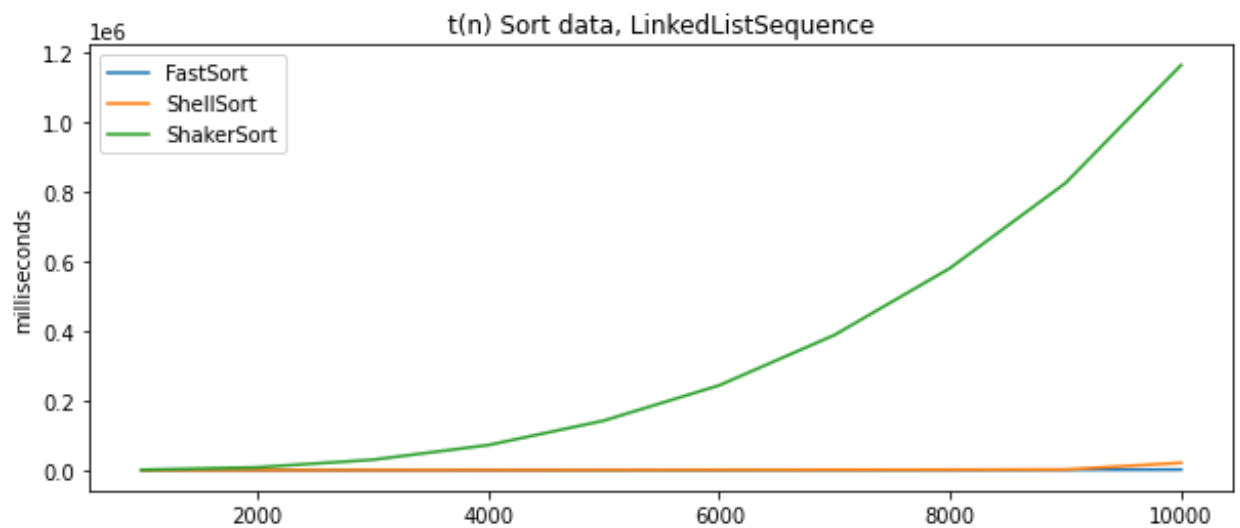
| Тип последовательности | Сортировка           | Время при data = $10^4$ |
|------------------------|----------------------|-------------------------|
| ArraySequence          | Быстрая сортировка   | 3 мс                    |
|                        | Шейкерная сортировка | 1,4 с                   |
|                        | Сортировка Шелла     | 172 мс                  |
| LinkedListSequence     | Быстрая сортировка   | 4,6 с                   |
|                        | Шейкерная сортировка | 47,4 мин                |
|                        | Сортировка Шелла     | 8,3 мин                 |



Проведя исследования для отсортированных и отсортированных в обратном направлении получим:

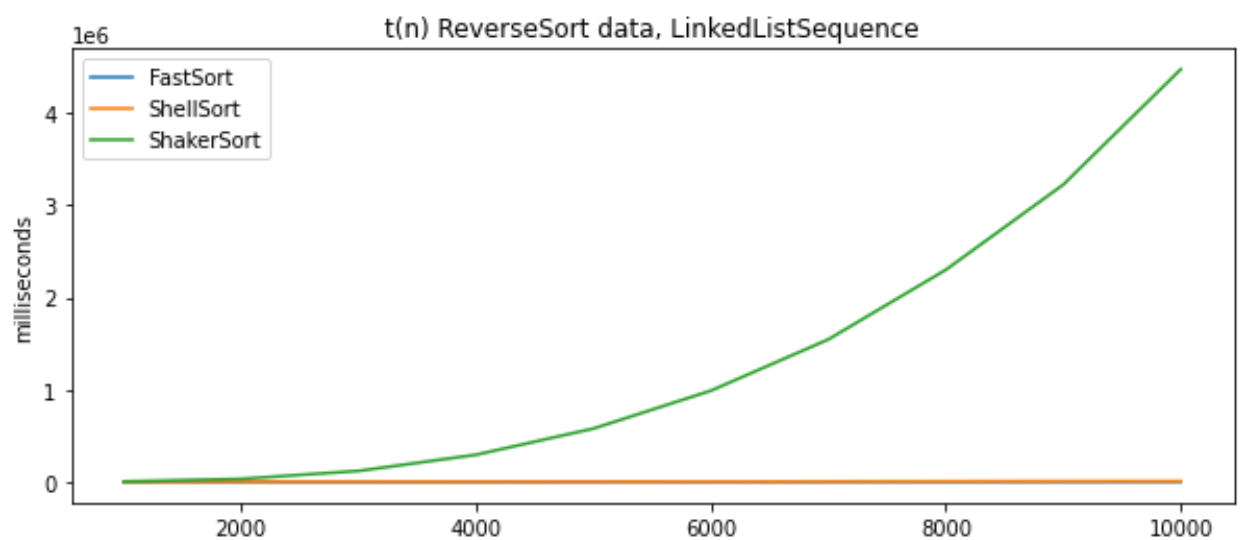
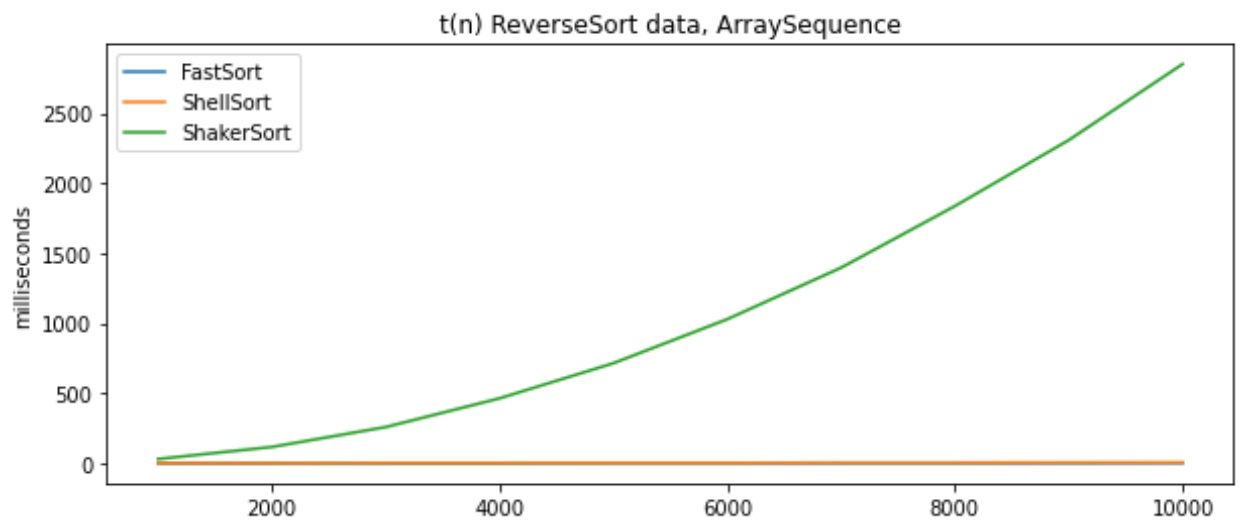
Для уже отсортированной последовательности:

| Тип последовательности | Сортировка           | Сложность                |
|------------------------|----------------------|--------------------------|
| ArraySequence          | Быстрая сортировка   | $O(n \cdot \log_2(n))$   |
|                        | Шейкерная сортировка | $O(n^2)$                 |
|                        | Сортировка Шелла     | $O(n)$                   |
| LinkedListSequence     | Быстрая сортировка   | $O(n^2 \cdot \log_2(n))$ |
|                        | Шейкерная сортировка | $O(n^3)$                 |
|                        | Сортировка Шелла     | $O(n^2)$                 |



Для последовательности отсортированной в обратном направлении:

| Тип последовательности | Сортировка           | Сложность                |
|------------------------|----------------------|--------------------------|
| ArraySequence          | Быстрая сортировка   | $O(n \cdot \log_2(n))$   |
|                        | Шейкерная сортировка | $O(n^2)$                 |
|                        | Сортировка Шелла     | $O(n \cdot \log_2(n))$   |
| LinkedListSequence     | Быстрая сортировка   | $O(n^2 \cdot \log_2(n))$ |
|                        | Шейкерная сортировка | $O(n^3)$                 |
|                        | Сортировка Шелла     | $O(n^2 \cdot \log_2(n))$ |



Выводы:

Для быстрой сортировки не важно как отсортирована последовательность, почти всегда сложность одна и та же, лишь в некоторых случаях(когда опорный элемент все время в начале или в конце) сложность будет  $O(n^2)$

Шейкерная сортировка одинаково плохо показывает себя на любых данных, но все же работает быстрее при отсортированной последовательности, и медленнее при неотсортированной

Данная реализация сортировки Шелла плохо работает на случайных данных, но в отличие от шейкерной сортировки быстрее, а так же имеет крайне хорошую сложность отсортированных данных и сложность равную сложности быстрой сортировки при обратно отсортированных данных, однако все равно работает в среднем медленнее чем быстрая сортировка.

Для теоретического анализа сортировки шелла использовалась следующая формула :

$2N^2/h + \sqrt{\pi N^3 h}$ , где  $h$  – шаг сортировки равный в данной реализации  $N/2$ .