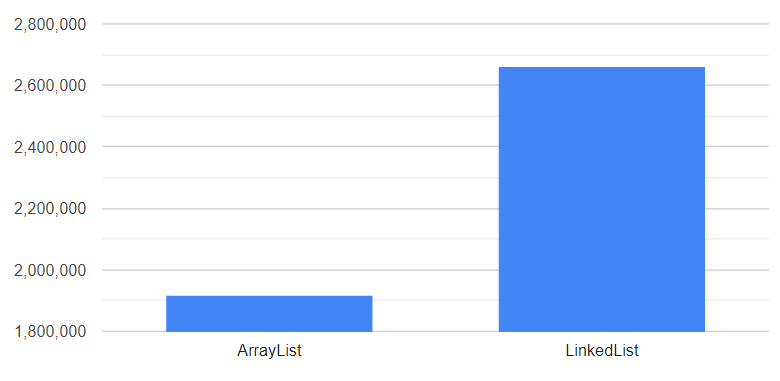
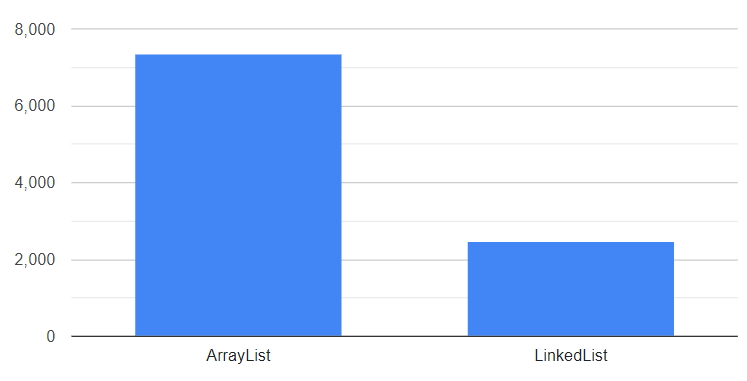
**Сравнение ArrayList, LinkedList**

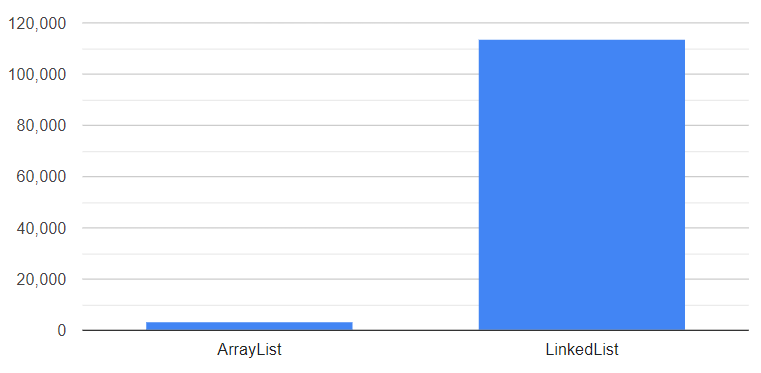
**Добавление элементов**



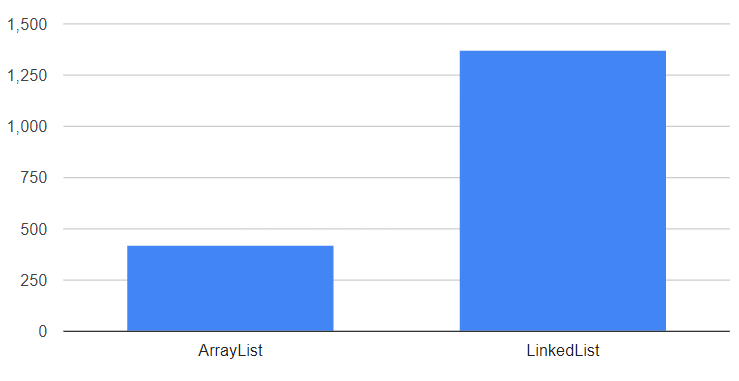
**Вставка элементов в начало**



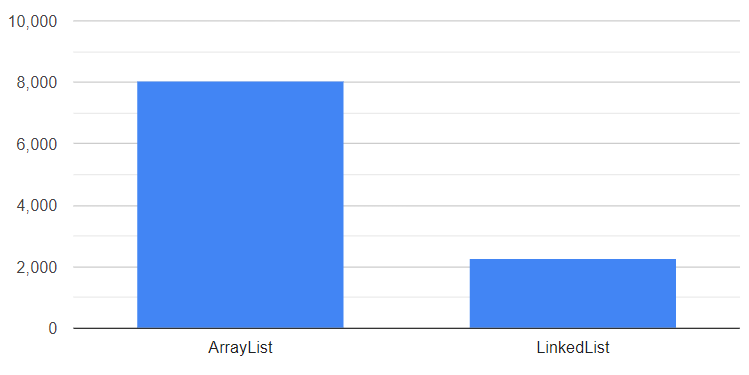
**Вставка элементов в середину**



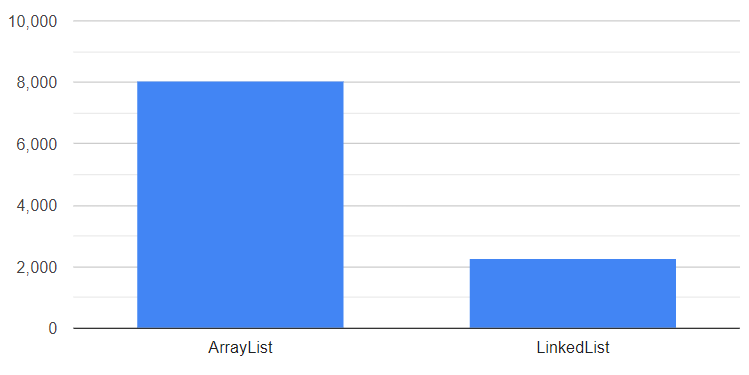
**Вставка элементов в конец**



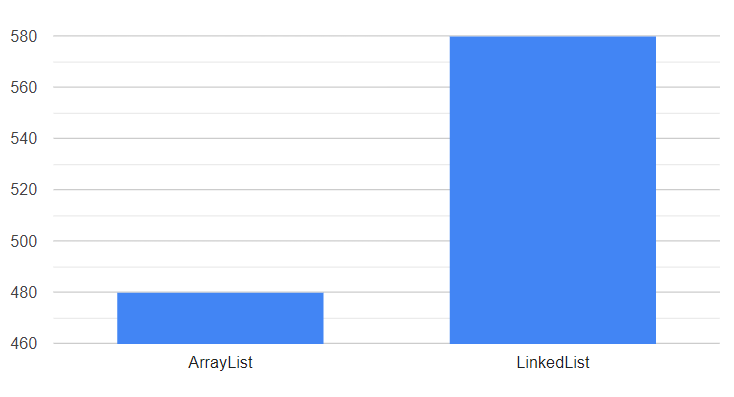
**Удаление элементов в начало**



**Удаление элементов в середину**



**Удаление элементов в конец**



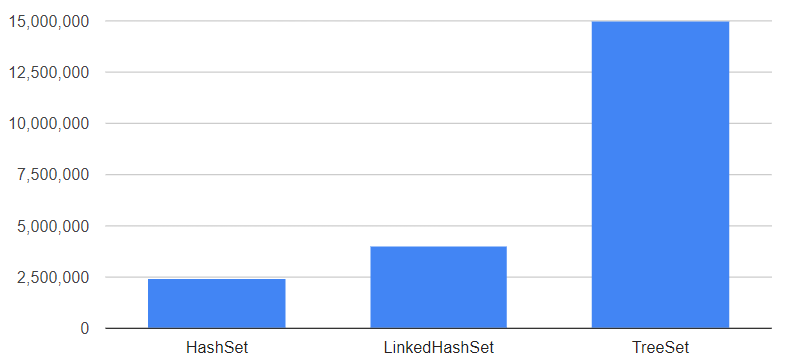
**Итоги сравнения:**

ArrayList работает быстрее при добавлении. При вставке в середину и в конец, а при удалении в конце тоже работает быстрее. Во всех других случаях LinkedList работает быстрее.

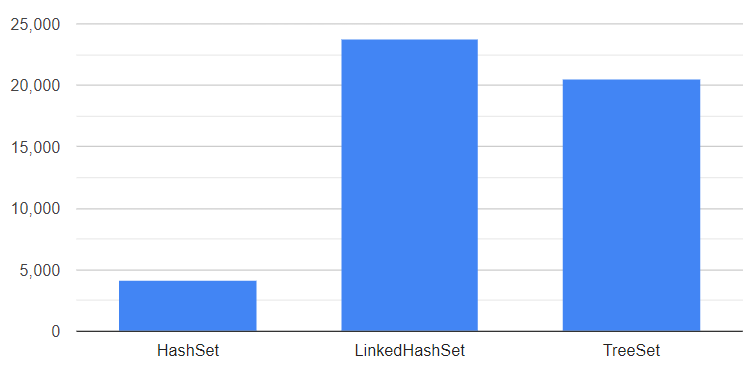
Следовательно при частой вставке и удаления лучше использовать LinkedList, в противном случае ArrayList.

**Сравнение HashSet, LinkedHashSet, TreeSet**

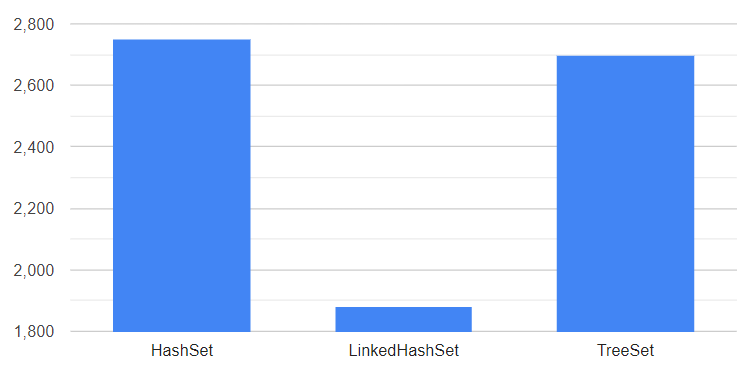
**Добавление элементов**



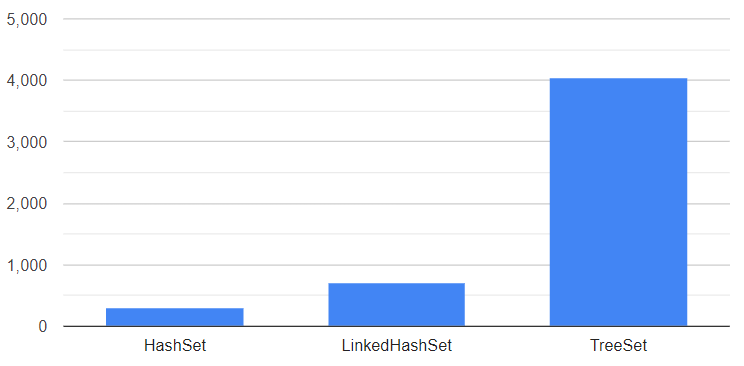
**Вставка элементов в начало**



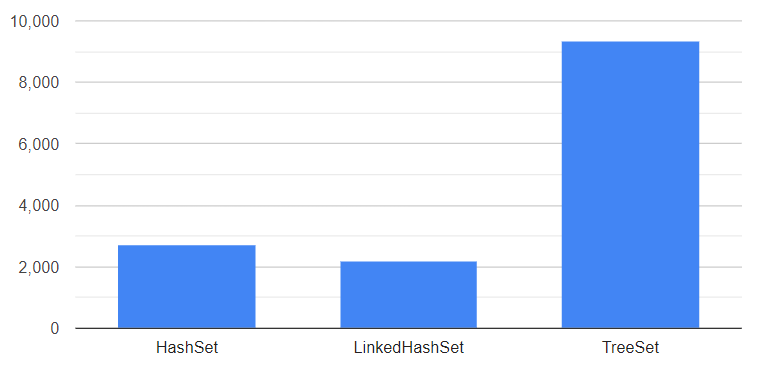
**Вставка элементов в середину**



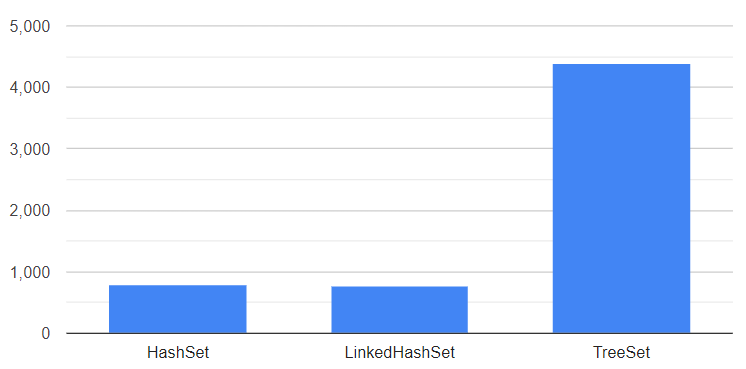
**Вставка элементов в конец**



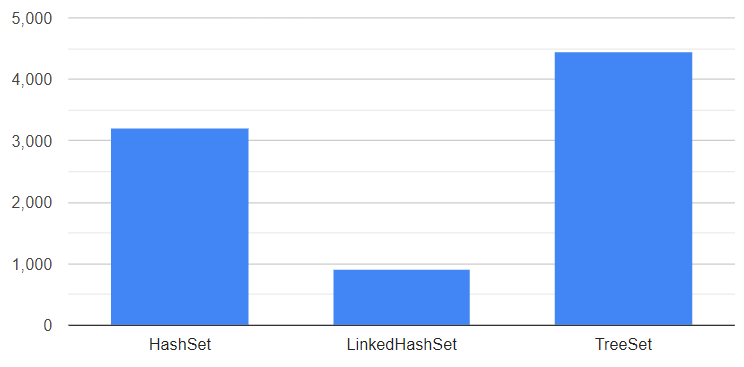
**Удаление элементов в начале**



**Удаление элементов в середине**



**Удаление элементов в конце**



**Итоги сравнения:**

LinkedHashSet работает быстрее на удаление и на вставку в середине.

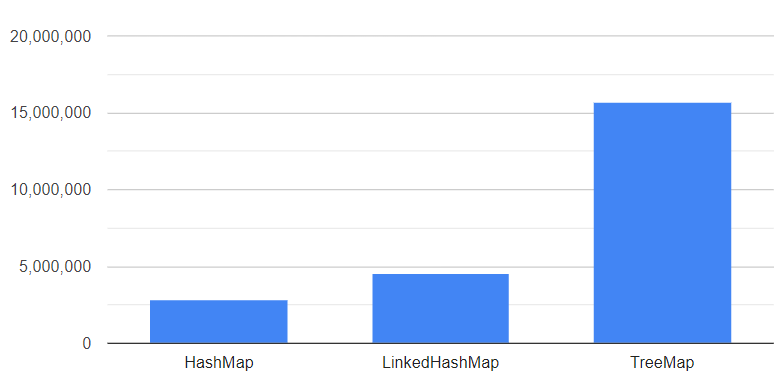
HashSet работает быстрее на добавление и на вставку в начало и в конец.

TreeSet работает медленнее во всех случаях.

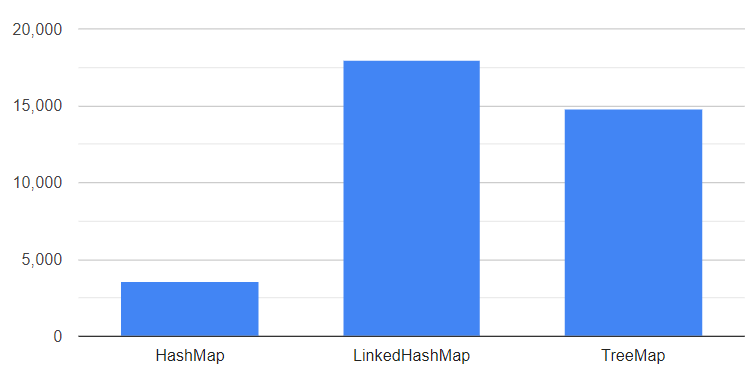
Следовательно HashSet - самый быстрый, LinkedHashSet - второй по производительности, но TreeSet работает медленнее из-за операции сортировки, которую он должен выполнять при каждой вставке.

**Сравнение HashMap, LinkedHashMap, TreeMap**

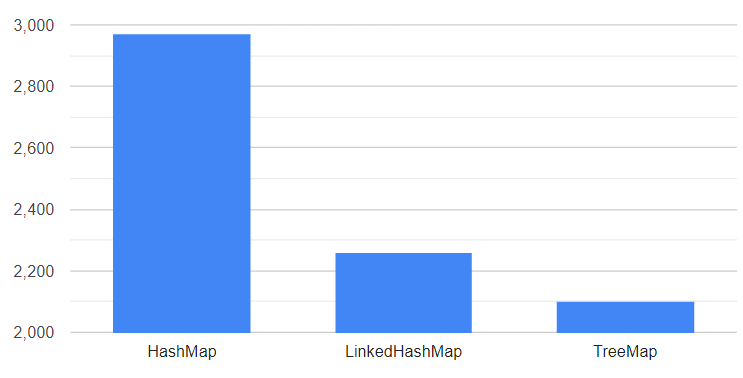
**Добавление элементов**



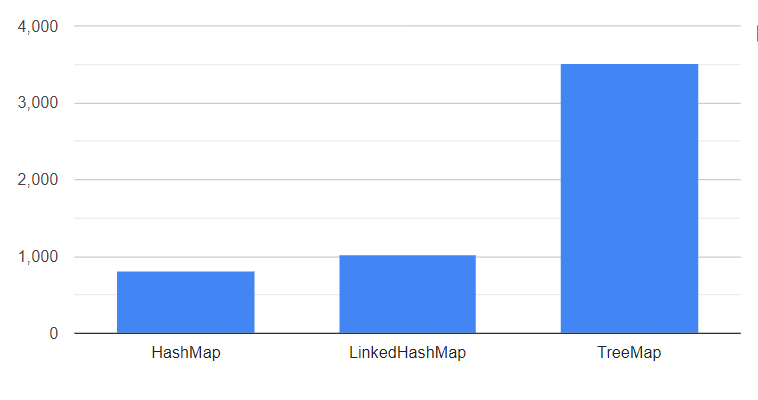
**Вставка элементов в начало**



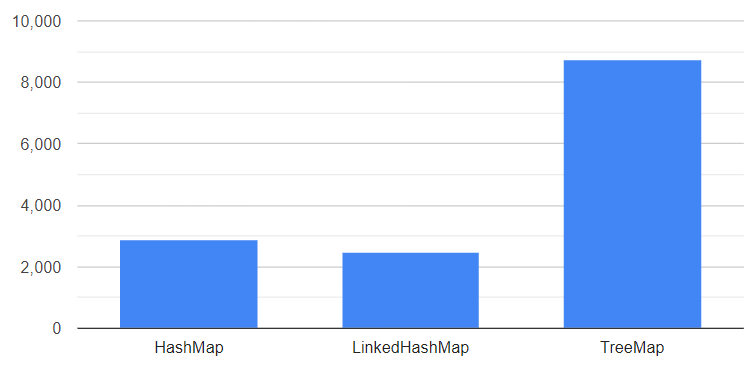
**Вставка элементов в середину**



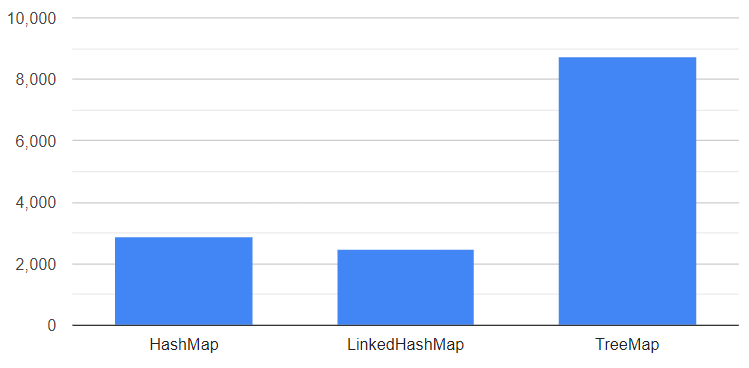
**Вставка элементов в конец**



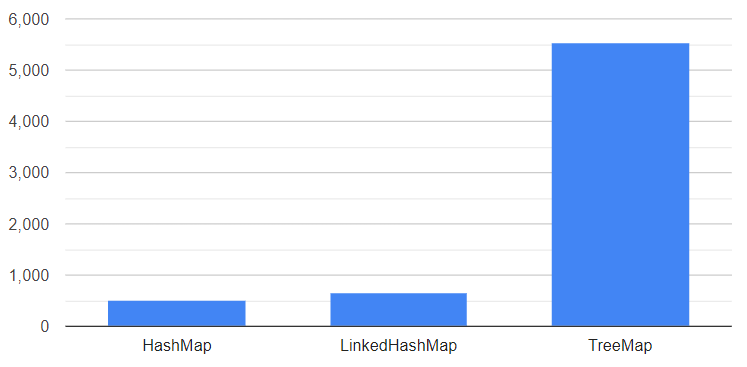
**Удаление элементов в начале**



**Удаление элементов в середине**



**Удаление элементов в конце**



**Итоги сравнения:**

HashMap работает быстрее на добавление и на вставку в начало и в конец.

LinkedHashMap работает быстрее на удаление в начале и в середине

TreeMap работает быстрее при вставке в середину.

Следовательно HashSet - самый быстрый, LinkedHashSet - второй по производительности и TreeSet работает медленнее.