# Выводы по Java – коллекциям

Общее количество элементов равно  10 000.

Время выполнения основных операций

|  |  |  |  |
| --- | --- | --- | --- |
|  | Добавление, нс | Вставка, нс | Удаление, нс |
| ArrayList | 2 760 | 12 942 | 235 380 |
| LinkedList | 2 203 | 433 463 | 179 215 |
| HashSet | 9 187 | - | 6 065 |
| LinkedHashSet | 19 307 | - | 3 261 |
| TreeSet | 18 773 | - | 78 404 |
| HashMap | 5 667 | - | 7 793 |
| LinkedHashMap | 2 050 | - | 3 470 |
| TreeMap | 5 158 | - | 9 232 |

1. Вставка элементов происходит быстрее в ArrayList примерно в 30 раз, чем у LinkedList. Для хранения данных в списке, где требуется частая вставка или удаление, лучше использовать ArrayList, так как производительность данной реализации выше, чем у LinkedList.
2. В Set коллекциях нет повторяющихся элементов. Лучший по скорости добавления элементов среди Set интерфейсов – это HashSet. Но если важно, чтобы элементы были расположены в порядке их добавления, то стоит выбрать LinkedHashSet. Наименьшее время удаления у LinkedList возможно связано с тем, что проверялось время удаления последнего вставленного элемента. TreeSet примерно одинаков по времени добавления элементов, но в данной реализации множество отсортировано.
3. Время добавления у HashMap, LinkedHashMap и TreeMap примерно одно порядка. В Map коллекции хранится два объекта: ключ и значение. Ключи – уникальные объекты. TreeMap хранит значения ключей в отсортированном порядке, а LinkedHashMap хранит ключи в порядке вставки. Если необходимо постоянное добавление элементов, то лучше для этого подойдет LinkedHashMap, так как он имеет наименьшее время добавления.