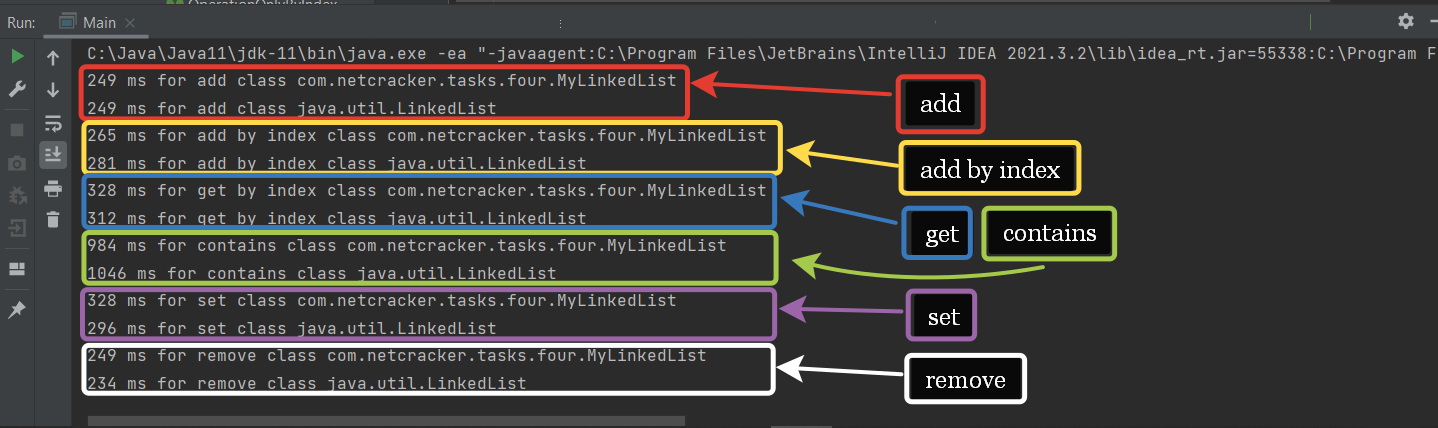
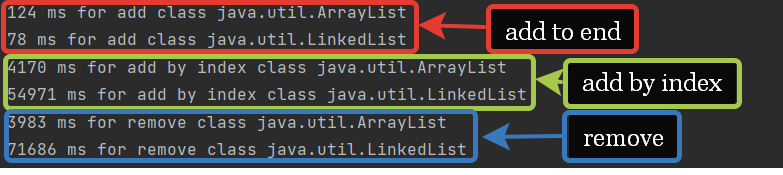
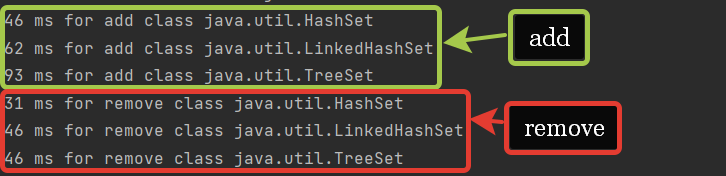
1. Итог: Односвязный созданный MyLinkedList немного проигрывает двусвязному LinkedList из стандартной библиотеки, все операции, где необходимо найти Node в двусвязном списке будут работать быстрее, тестовые данные от 10000 - 100000 элементов для разных операций. В качестве добавляемого типа был взят класс Person  
   
2. Сравнение коллекций из стандартной библиотеки
3. ArrayList vs LinkedList: добавление элементов в начало или конец коллекции выполняется быстрее у LinkedList, но операции добавления по индексу и удаление производятся быстрее. LinkedList рекомендуется использовать при создании коллекции для хранения и добавления/удаления элементов в начале или конце, для более быстрого поиска элементов, добавления, удаления рекомендуется использовать ArrayList. Выборка 100000-200000 элементов.  
   
4. HashSet vs LinkedHashSet vs TreeSet: HashSet показал лучшие результаты для добавления и удаления элементов, рекомендуется его использовать для работы с неупорядоченными элементами. LinkedHashSet элементы упорядочены в порядке добавления в небольшой ущерб быстродействию. TreeSet упорядоченная коллекция, в ущерб производительности из-за сортировки и перестройки дерева. Какую из коллекций использовать зависит от необходимости упорядочивания элементов. Выборка 100000-200000 элементов.  
   
5. HashMap vs LinkedHashMap vs TreeMap: HashMap показал лучшие результаты для добавления и удаления элементов, рекомендуется его использовать для работы с неупорядоченными элементами. LinkedHashMap элементы упорядочены в порядке добавления с возможным небольшим ущербом быстродействию. LinkedHashMap существует только для итерации в порядке вставки (или доступа). TreeMap упорядоченная коллекция (красно-черное дерево), в ущерб производительности из-за сортировки и перестройки дерева. Какую из коллекций использовать зависит от необходимости упорядочивания элементов. Выборка 100000-200000 элементов.   
     
   