

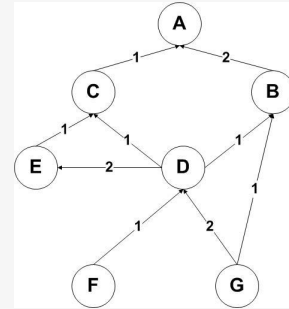
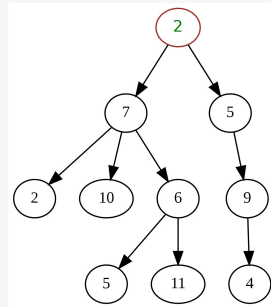
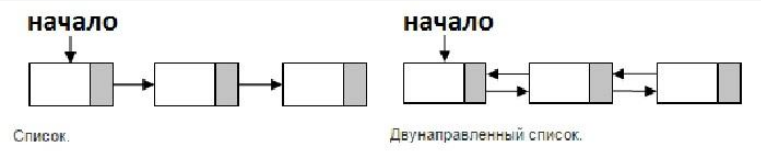
# Структуры данных. Алгоритмы. Коллекции.

## План

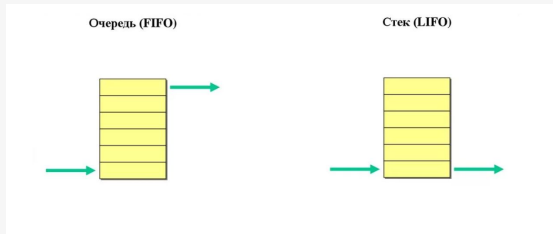
1. Структуры данных
2. Сложность алгоритма
3. Основные алгоритмы (поиск, сортировка)
4. Иерархия коллекций
5. Скорость операций
6. Immutable collections

# Структуры данных

Структура данных — это контейнер, который хранит данные в определенном макете. Этот «макет» позволяет структуре данных быть эффективной в некоторых операциях и неэффективной в других.



3	<key>	<data>
⋮		
16	<key>	<data>
17	<key>	<data>



<https://habr.com/ru/post/422259/>

## Сложность алгоритма

Сложность алгоритма - зависимость количества необходимых действий, от количества входных элементов.

1.  $O(1)$
2.  $O(\log n)$
3.  $O(n)$
4.  $O(n * \log n)$
5.  $O(n*n)$
6.  $O(2 \text{ в степени } n)$

<https://habr.com/ru/post/104219/>

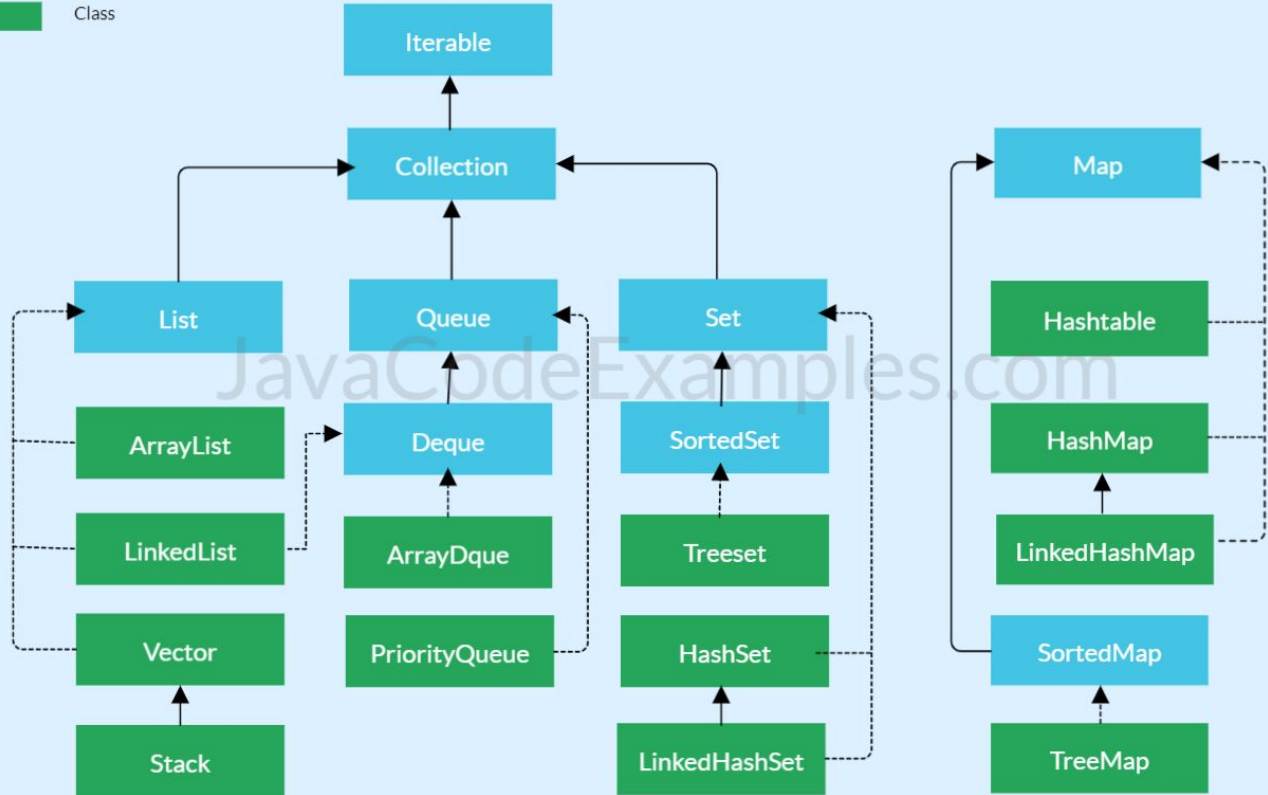
## Основные алгоритмы

1. Поиск
  - a. Линейный поиск
  - b. Бинарный поиск
2. Сортировка
  - a. пузырьковая сортировка
  - b. сортировка вставками
  - c. сортировка слиянием
  - d. быстрая сортировка
  - e. пирамидальная сортировка

<https://academy.yandex.ru/posts/osnovnye-vidy-sortirovok-i-primery-ikh-realizatsii>

# Java Collections

<https://www.JavaCodeExamples.com>



## Скорость операций

	Временная сложность							
	Среднее				Худшее			
	Индекс	Поиск	Вставка	Удаление	Индекс	Поиск	Вставка	Удаление
ArrayList	$O(1)$	$O(n)$	$O(n)$	$O(n)$	$O(1)$	$O(n)$	$O(n)$	$O(n)$
Vector	$O(1)$	$O(n)$	$O(n)$	$O(n)$	$O(1)$	$O(n)$	$O(n)$	$O(n)$
LinkedList	$O(n)$	$O(n)$	$O(1)$	$O(1)$	$O(n)$	$O(n)$	$O(1)$	$O(1)$
Hashtable	n/a	$O(1)$	$O(1)$	$O(1)$	n/a	$O(n)$	$O(n)$	$O(n)$
HashMap	n/a	$O(1)$	$O(1)$	$O(1)$	n/a	$O(n)$	$O(n)$	$O(n)$
LinkedHashMap	n/a	$O(1)$	$O(1)$	$O(1)$	n/a	$O(n)$	$O(n)$	$O(n)$
TreeMap	n/a	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$	n/a	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$
HashSet	n/a	$O(1)$	$O(1)$	$O(1)$	n/a	$O(n)$	$O(n)$	$O(n)$
LinkedHashSet	n/a	$O(1)$	$O(1)$	$O(1)$	n/a	$O(n)$	$O(n)$	$O(n)$
TreeSet	n/a	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$	n/a	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$

## Нормализация и денормализация

Неизменяемым (англ. *immutable*) называется объект, состояние которого не может быть изменено после создания. Результатом любой модификации такого объекта всегда будет новый объект, при этом старый объект не изменится.

<code>static &lt;T&gt; Collection&lt;T&gt;</code>	<code>unmodifiableCollection(Collection&lt;? extends T&gt; c)</code> Returns an unmodifiable view of the specified collection.
<code>static &lt;T&gt; List&lt;T&gt;</code>	<code>unmodifiableList(List&lt;? extends T&gt; list)</code> Returns an unmodifiable view of the specified list.
<code>static &lt;K,V&gt; Map&lt;K,V&gt;</code>	<code>unmodifiableMap(Map&lt;? extends K,? extends V&gt; m)</code> Returns an unmodifiable view of the specified map.
<code>static &lt;T&gt; Set&lt;T&gt;</code>	<code>unmodifiableSet(Set&lt;? extends T&gt; s)</code> Returns an unmodifiable view of the specified set.
<code>static &lt;K,V&gt; SortedMap&lt;K,V&gt;</code>	<code>unmodifiableSortedMap(SortedMap&lt;K,? extends V&gt; m)</code> Returns an unmodifiable view of the specified sorted map.
<code>static &lt;T&gt; SortedSet&lt;T&gt;</code>	<code>unmodifiableSortedSet(SortedSet&lt;T&gt; s)</code> Returns an unmodifiable view of the specified sorted set.

<code>static &lt;E&gt; List&lt;E&gt; copyOf(Collection&lt;? extends E&gt; coll)</code>	Returns an unmodifiable List containing the elements of the given Collection, in its iteration order.
--	---

<code>static &lt;E&gt; List&lt;E&gt; copyOf(Collection&lt;? extends E&gt; coll)</code>	Returns an unmodifiable List containing the elements of the given Collection, in its iteration order.
--	---





## Другие коллекции

- NavigableMap/Set/List
- Properties
- WeakHashMap
- Concurrent Collections

<https://habr.com/ru/company/citymobil/blog/545004/>