

Сергей Савчук
software engineer

MAP & SET

Сложные структуры данных



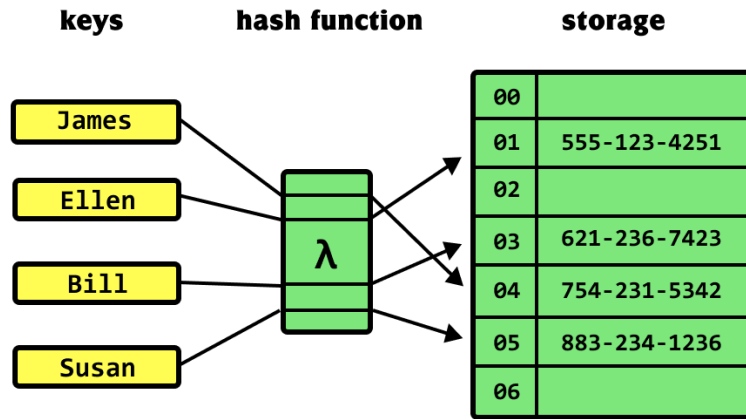


MAP

*Объекты для хранения
именованных коллекций*

MAP

Коллекция ключ/значение, как и Object. Но основное отличие в том, что Map позволяет использовать ключи любого типа.



Основные операции

- `new Map(iterable)` – создает коллекцию.
- `map.set(key, value)` – записывает по key значение value.
- `map.get(key)` – возвращает значение по key или undefined.
- `map.has(key)` – возвращает true если key присутствует или false.
- `map.delete(key)` – удаляет элемент по key.
- `map.clear()` – очищает коллекцию.
- `map.size` – возвращает количество элементов в коллекции.

WEACK MAP

Если мы используем объект как ключ в Map, то до тех пор, пока существует Map, также будет существовать и этот объект. Он занимает место в памяти и не может быть удалён сборщиком мусора.

В Weak Map ключи должны быть объектами.

Основные операции

- `new WeakMap(iterable)` – создает коллекцию.
- `map.set(key, value)` – записывает по `key` значение `value`.
- `map.get(key)` – возвращает значение по `key` или `undefined`.
- `map.has(key)` – возвращает `true` если `key` присутствует или `false`.
- `map.delete(key)` – удаляет элемент по `key`.

СЛОЖНОСТЬ ОПЕРАЦИЙ

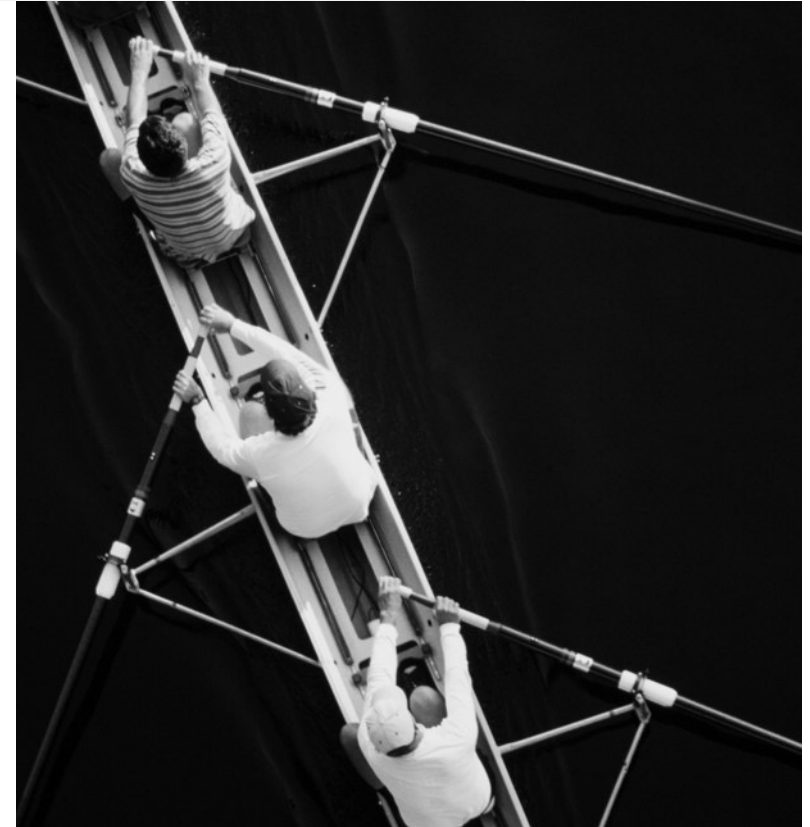
Скорость операций - [статья на habr](#)

Container	Insert Head	Insert Tail	Insert	Remove Head	Remove Tail	Remove	Index Search	Find
vector	n/a	O(1)	O(n)	O(1)	O(1)	O(n)	O(1)	O(log n)
list	O(1)	O(1)	O(1)	O(1)	O(1)	O(1)	n/a	O(n)
deque	O(1)	O(1)	n/a	O(1)	O(1)	O(n)	n/a	n/a
queue	n/a	O(1)	n/a	O(1)	n/a	n/a	O(1)	O(log n)
stack	O(1)	n/a	n/a	O(1)	n/a	n/a	n/a	n/a
map	n/a	n/a	O(log n)	n/a	n/a	O(log n)	O(1)	O(log n)
multimap	n/a	n/a	O(log n)	n/a	n/a	O(log n)	O(1)*	O(log n)
set	n/a	n/a	O(log n)	n/a	n/a	O(log n)	O(1)	O(log n)
multiset	n/a	n/a	O(log n)	n/a	n/a	O(log n)	O(1)*	O(log n)

ВЫВОД

МАР напоминает object но у него есть преимущества:

- Размер — карты имеют свойство size, в то время как объекты не имеют встроенного способа получения размера.
- Итерация — карты являются итерируемыми, а объекты нет.
- Гибкость — карты могут содержать любой тип данных (примитивы или объекты) в качестве ключа для значения, а объекты могут хранить только строки.
- Упорядоченность — карты сохраняют порядок добавления, а объекты не имеют гарантированного порядка.



SET

*Коллекция уникальных
значений*

SET

концептуально больше походит на массив, а не объект, поскольку он представляет собой список значений, а не пар ключ/значение.

Основные операции

- `new Set(iterable)` – создает коллекцию.
- `set.add(value)` – добавляет `value`.
- `set.has(value)` – возвращает `true` если `key` присутствует или `false`.
- `set.delete(value)` – удаляет элемент по `key`.
- `set.clear()` – очищает коллекцию.
- `set.size` – возвращает количество элементов в коллекции.

WEACK SET

Коллекция WeakSet ведёт себя похоже:

- Она аналогична Set, но мы можем добавлять в WeakSet только объекты (не примитивные значения).
- Объект присутствует в множестве только до тех пор, пока доступен где-то ещё.

Основные операции

- `new WeackSet(iterable)` – создает коллекцию.
- `set.add(value)` – добавляет value.
- `set.has(value)` – возвращает true если key присутствует или false.
- `set.delete(value)` – удаляет элемент по key.

СЛОЖНОСТЬ ОПЕРАЦИЙ

Container	Insert Head	Insert Tail	Insert	Remove Head	Remove Tail	Remove	Index Search	Find
vector	n/a	O(1)	O(n)	O(1)	O(1)	O(n)	O(1)	O(log n)
list	O(1)	O(1)	O(1)	O(1)	O(1)	O(1)	n/a	O(n)
deque	O(1)	O(1)	n/a	O(1)	O(1)	O(n)	n/a	n/a
queue	n/a	O(1)	n/a	O(1)	n/a	n/a	O(1)	O(log n)
stack	O(1)	n/a	n/a	O(1)	n/a	n/a	n/a	n/a
map	n/a	n/a	O(log n)	n/a	n/a	O(log n)	O(1)	O(log n)
multimap	n/a	n/a	O(log n)	n/a	n/a	O(log n)	O(1)*	O(log n)
set	n/a	n/a	O(log n)	n/a	n/a	O(log n)	O(1)	O(log n)
multiset	n/a	n/a	O(log n)	n/a	n/a	O(log n)	O(1)*	O(log n)

ВЫВОД

SET

можно использовать для поиска
идентичных, пересекающихся и отличных
элементов двух наборов данных.



THANK YOU

[Map & Set](#)

[Weak Map & Weak Set](#)

[Garbage collector](#)