

Алгоритмы и коллекции стандартных библиотек

C++, C#, Java, Objective-C

Об одной жемчужине *Microsoft*,
портированной на *Obj-C...*

АНТОН БУКОВ
@k06a

A problem has been detected and windows has been shut down to prevent damage to your computer.

PROCESS1_INITIALIZATION_FAILED

If this is the first time you've seen this Stop error screen, restart your computer. If this screen appears again, follow these steps:

Check to make sure any new hardware or software is properly installed. If this is a new installation, ask your hardware or software manufacturer for any windows updates you might need.

If problems continue, disable or remove any newly installed hardware or software. Disable BIOS memory options such as caching or shadowing. If you need to use Safe Mode to remove or disable components, restart your computer, press F8 to select Advanced Startup Options, and then select Safe Mode.

Technical information:

*** STOP: 0x0000006B (0xC0000022,0x00000002,0x00000000,0x00000000)

Основные сущности стандартной библиотеки

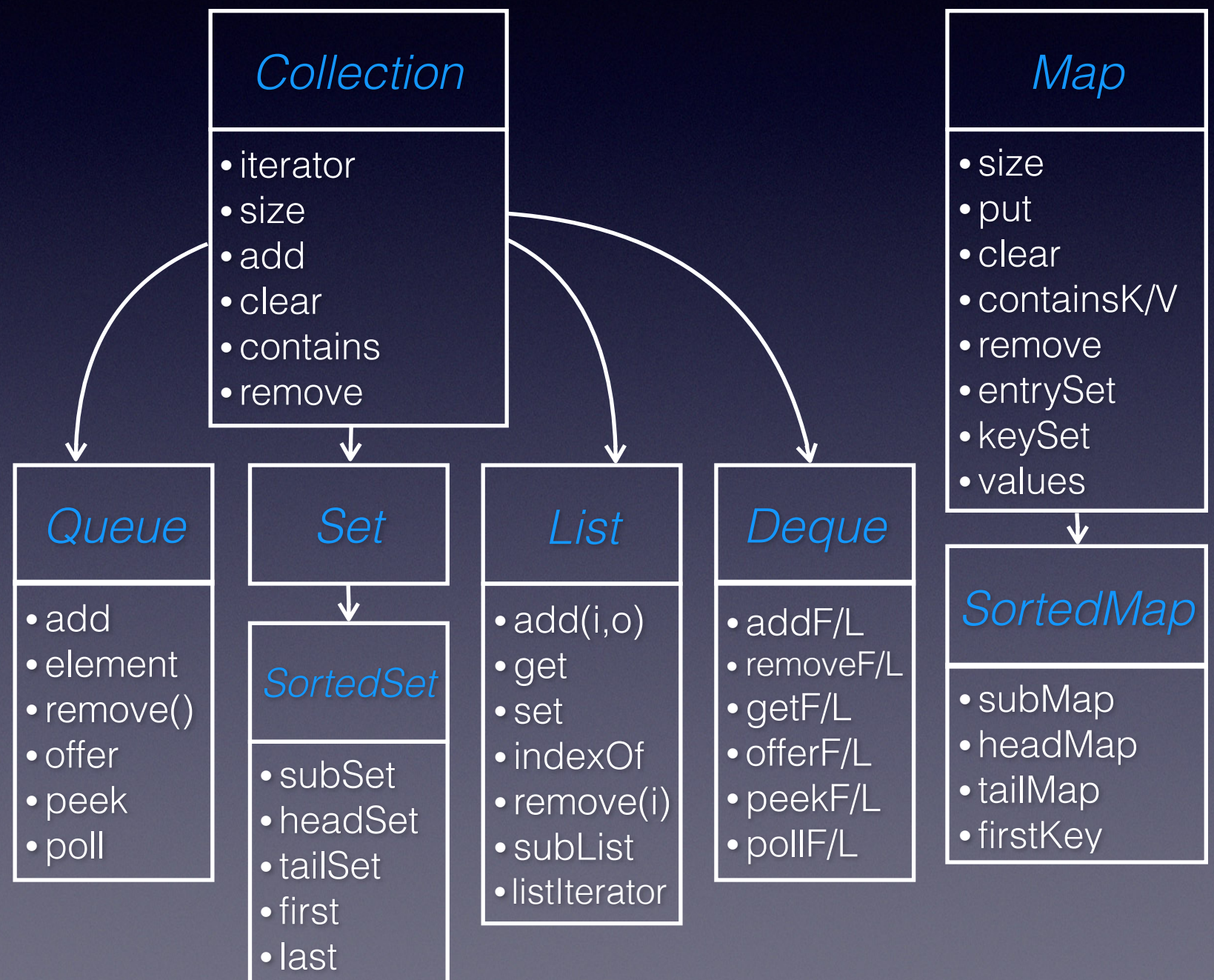
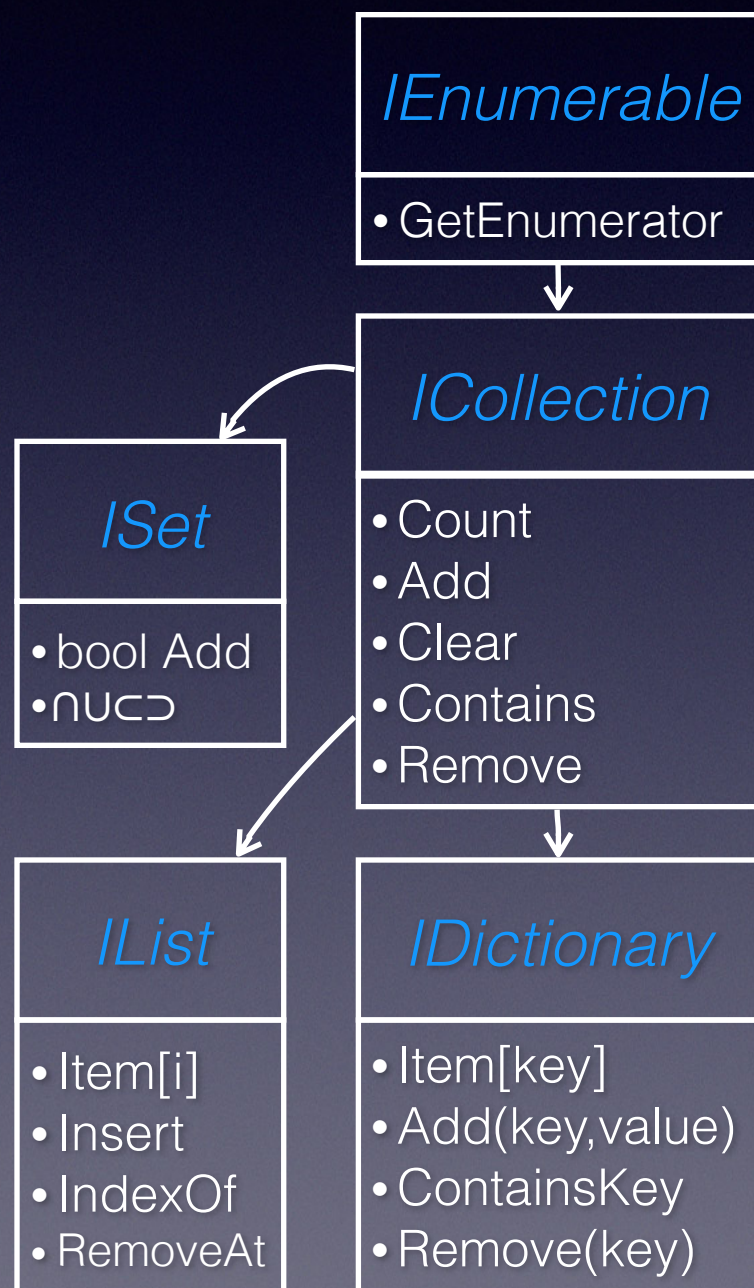
- Коллекции (структуры данных)
- Алгоритмы (универсальные и специальные)
- Перечислители (iterator, enumerator)

Коллекции. (структуры данных)

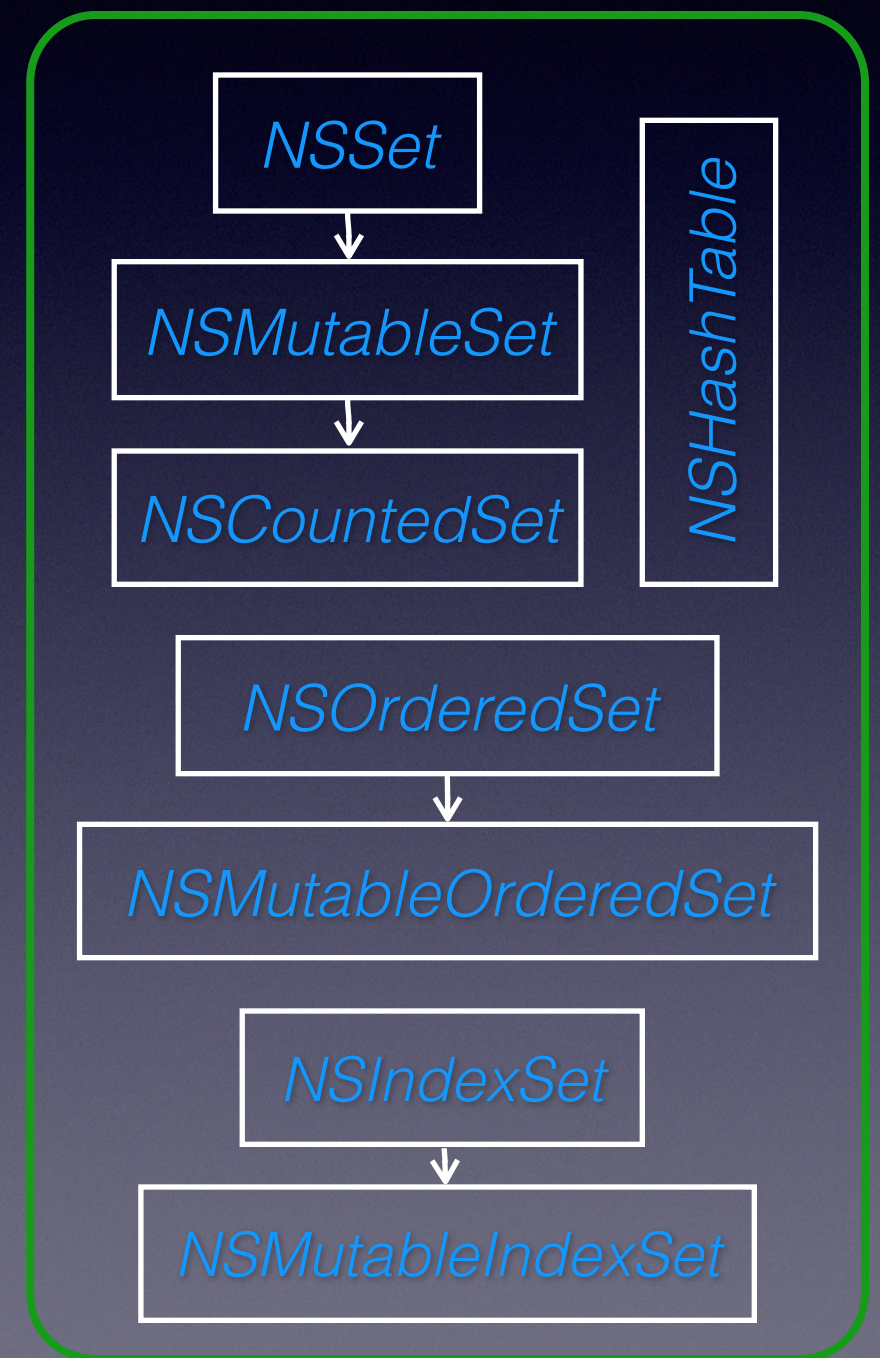
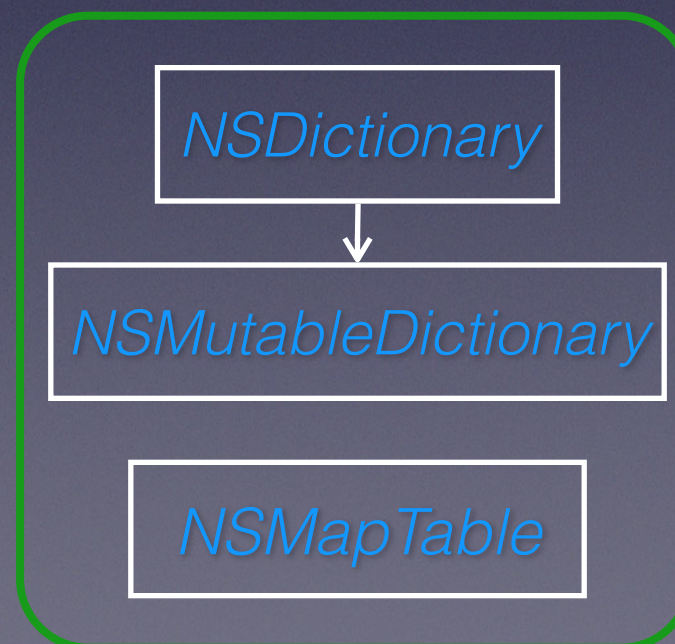
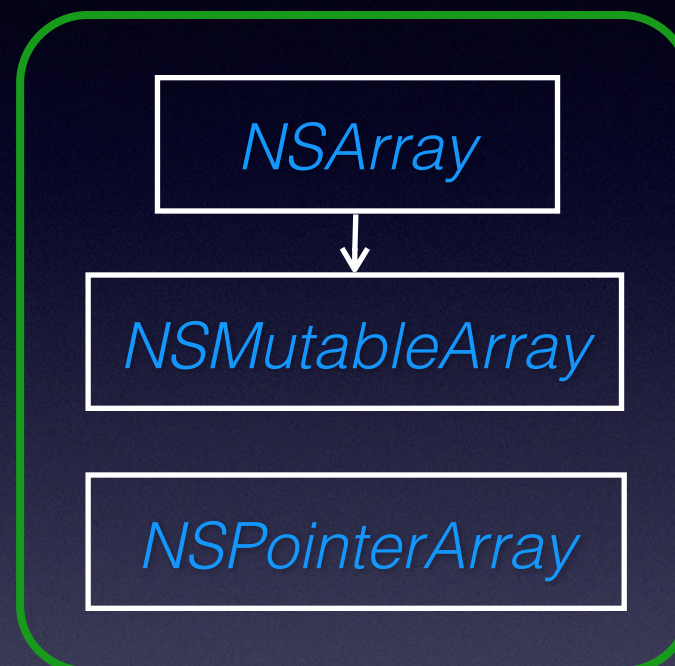
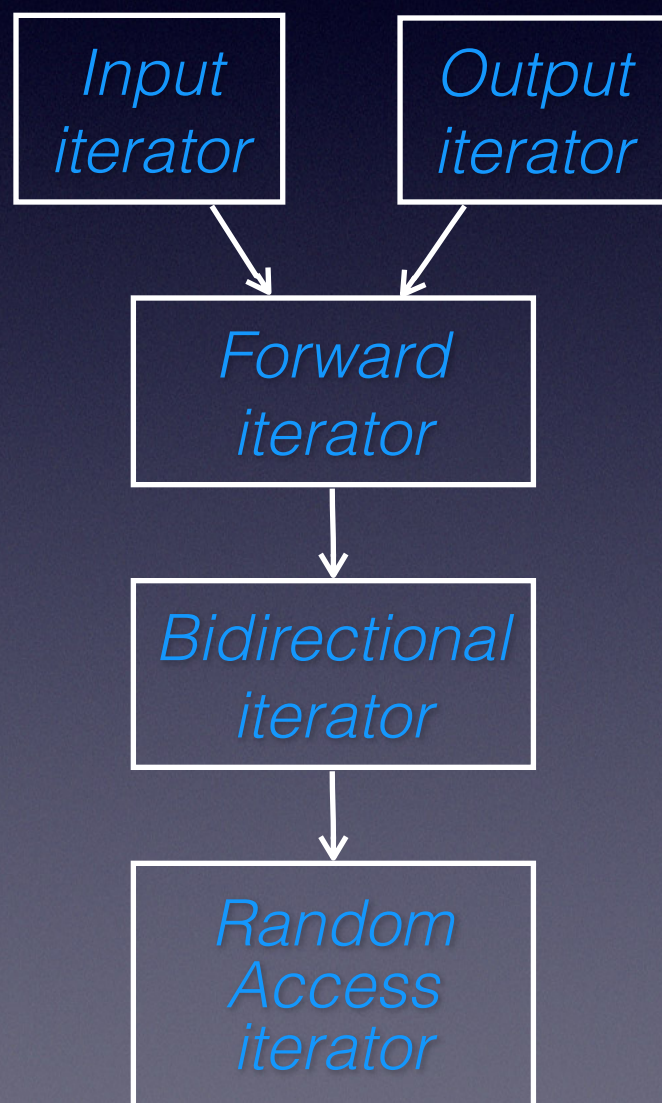
	C++ (STL)	C# (.NET Framework)	Java (Collections Framework)	Obj-C (Foundation Framework)
Постоянного размера	[] std::array	Array	Array	[] NSArray
Не зависят от значений	vector list stack queue deque	List LinkedList Stack Queue	ArrayList LinkedList ArrayDeque	NSMutableArray NSPointerArray
Сравнивают значения	set map	SortedSet SortedDictionary	TreeSet TreeMap	NSMutableOrderedSet
Считают хеши и сравнивают значения	unordered_set unordered_map	Set Dictionary	HashSet/ LinkedHashSet HashMap/ LinkedHashMap	NSMutableSet NSCountedSet NSHashTable NSMutableDictionary NSMapTable

Иерархия коллекций.

Интерфейсы C# и Java



Иерархия коллекций. C++ и Obj-C



Иерархия коллекций.

Итераторы в C++ (STL)

Категории				Характеристика	Выражение
все				Может быть скопирован и создан по образу и подобию	X b(a); и b = a;
				Может быть увеличен на единицу	++a и a++
RA	BD	FW	I	Поддерживает сравнение на равенство/неравенство	a == b и a != b
			I	Может быть разыменован как rvalue для получения значения	*a
		O	O	Может быть разыменован как lvalue для использования слева от знака присваивания	*a = t *a++ = t
				Может быть создан конструктором по-умолчанию	X a;
				Может быть скопирован и использован для повторного обхода	X b(a); ++a == ++b
				Может быть уменьшен на единицу	--a и a--
				Поддерживает арифметические операции + и -	a+n и a-n
				Поддерживает сравнения (<, >, <= и >=) между итераторами	a < b и a <= b
				Поддерживает операции увеличения += и уменьшения -=	a += n и a -= n
				Поддерживает разыменование по индексу	a[n]

RA — *Random Access iterator*

I — *Input iterator*

FW — *Forward iterator*

BD — *Bidirectional iterator*

O — *Output iterator*

Перечислители (iterator, enumerator)

C++	C#
<pre>for (auto it = arr.begin(); it != arr.end(); ++it) { std::cout << *it; sum += *it; } // --it, it+n, it-n</pre>	<pre>var en = arr.GetEnumerator(); while (en.MoveNext()) { Console.Write(en.Current); sum += en.Current; } en.Dispose(); // using</pre>
Java	Objective-C
<pre>Iterator it = arr.iterator(); while (it.hasNext()) { int value = (Integer) it.next(); System.out.print(value); sum += value; }</pre>	<pre>id en = [arr objectAtIndex:]; id object; while (object = [en nextObject]) { NSLog(@"%@", object); sum += [object intValue]; }</pre>

Алгоритмы

(1 из 2)

	C++ (STL)	C# (.NET Framework)	Java (Collections Framework)	Obj-C (Foundation Framework)
Работа с множествами	merge set_union set_intersect set_difference	Union Intersect Except	disjoint	unionSet minusSet intersectSet removeObjects- InArray
Сортировки	sort stable_sort partial_sort	OrderBy ThenBy OrderByDescending ThenByDescending	sort sorted	sortedArray* keysSortedBy*
Поисковые	find search binary_search	List.BinarySearch List.Find*** IList.IndexOf Where TakeWhile SkipWhile	binarySearch indexOfSubList filter	indexOfObject* filterUsing filtered*Using objectsPassingTest NSPredicate NSExpression
Агрегирующие	all_of any_of none_of count min max	All Any Count Sum Average Min Max	frequency min max allMatch anyMatch reduce noneMatch	valueForKeyPath @avg @count @max @min @sum

Алгоритмы

(2 из 2)

	C++ (STL)	C# (.NET Framework)	Java (Collections Framework)	Obj-C (Foundation Framework)
Прочие	copy swap transform replace fill generate remove unique reverse rotate shuffle	List.Reverse Select Take Skip Enumerable.Repeat Enumerable.Range Distinct Reverse Join GroupBy SelectMany	copy swap replaceAll fill reverse rotate shuffle	@distinctUnion-OfObjects @unionOfObjects @distinctUnion-OfArrays @unionOfArrays @unionOfSets

Пример

(1 из 6)

- Дано:

```
struct Tweet { id, time, text, url, user_id };
```

```
struct User { id, name, nick };
```

```
array<Tweet> tweets;
```

```
map<int, User> userDict;
```

- Найти:

Имя пользователя с максимальным числом твитов за фиксированную дату (1 сент 2013), содержащих ссылку в поле `url` и само количество таких твитов.

Пример

(2 из 6)

C++:

```
map<int,int> countById;
for (auto it = tweets.begin(); it != tweets.end(); ++it)
    if ("2013-09-01" <= it->time && it->time < "2013-09-02")
        if (it->url != 0)
            countById[it->user_id]++;

auto para = max_element(countById.begin(),
                        countById.end(),
                        [](pair<int,int> a, pair<int,int> b)
                        { return a.second < b.second; });

cout << "User = " << userDict[para.first].name << endl
      << "Value = " << para.second << endl;
```


Пример

(3 из 6)

Obj-C:

```
NSMutableDictionary * countByUserId = [NSMutableDictionary dictionary];
for (Tweet * tweet in tweets)
    if ([@"2013-09-01" compareTo:tweet.time] <= 0 && [tweet.time compareTo:@"2013-09-02"] < 0)
        if (tweet.url != 0) {
            int count = [countByUserId[tweet->user_id] intValue];
            countByUserId[tweet->user_id] = @(count + 1);
        }

int maxCount = 0;
id maxKey = nil;
[countByUserId enumerateKeysAndObjectsUsingBlock:^(id key, id obj, BOOL *stop) {
    if ([obj intValue] > maxCount) {
        maxCount = [obj intValue];
        maxKey = key;
    }
}];

NSLog(@"User = %@\nValue = %d",
      [userDict[maxKey] name],
      maxCount);
```


Пример

(4 из 6)

Java:

```
HashMap<int,int> countByUserId = new HashMap<int,int>();
for (Tweet tweet : tweets)
    if ("2013-09-01" <= tweet.time && tweet.time < "2013-09-02")
        if (tweet.url != 0) {
            int count = countByUserId.get(tweet.user_id);
            countByUserId.set(tweet->user_id, (count==null?0:count) + 1);
        }

int maxCount = 0;
id maxKey = nil;
for(Entry<int,int> entry : countByUserId.entrySet()) {
    if (entry.getValue() > maxCount) {
        maxCount = entry.getValue();
        maxKey = entry.getKey();
    }
}

System.out.format("User = %s\nValue = %d",
                  userDict.get(maxKey),
                  maxCount);
```


Пример

(5 из 6)

C#:

```
var group = tweets.Where(t => "2013-09-01" <= t.time
                             && t.time < "2013-09-02")
                .Where(t => t.url != null)
                .GroupBy(t => t.user_id)
                .Max(gr => gr.Count());

Console.WriteLine("User = {0} \nValue = {1}",
                  userDict[group.Key].name,
                  group.Count());
```


Пример

(6 из 6)

Java 8 (java.util.stream):

```
Entry<int,List<int>>
entry = tweets.stream()
    .filter(t -> "2013-09-01" <= t.time
        && t.time < "2013-09-02")
    .filter(t -> t.url != null)
    .groupBy(t -> t.user_id) // Map<int,List<int>>
    .entrySet().stream()
    .max((a,b) -> Int.compare(a.getValue().size(),
        b.getValue().size()));

System.out.format("User = %s\nValue = %d",
    userDict.get(entry.getKey()),
    entry.getValue().size());
```


Пример

(7 из 6, ага)

Obj-C + NSEnumeratorLinq:

```
var para = [[[[tweets.objectEnumerator
    where:^(Tweet *t){return [@"2013-09-01" compareTo:t.time] <= 0
                                && [t.time compareTo:@"2013-09-02"] < 0;}]
    where:^(Tweet *t){return t.url != null;}]
    groupBy:^(Tweet *t){return t.user_id;}]
    max:^(NSKeyValuePair *p){return @(p.value count);}]];

NSLog(@"User = %@\nValue = %@",
    [userDict[para.key] name],
    [para.value count]);
```


Сравнение архитектур стандартных библиотек

C++:

- Иерархия итераторов
- Алгоритмы универсальные, зависят от категорий итераторов

C#:

- Иерархия интерфейсов
- Алгоритмы универсальные, зависят от интерфейсов, выстраиваются в цепь

Java:

- Иерархия интерфейсов
- Алгоритмы универсальные, зависят от интерфейсов ([java.util.Collections](#))
- Алгоритмы универсальные, выстраиваются в цепь ([java.util.stream](#))

Obj-C:

- Иерархия отсутствует, если не считать Immutable-Mutable
- Алгоритмы специфичные кроме одного :) [[NSEnumerator allObjects](#)]

NSEnumeratorLinq

- Основные методы:
 - where, where_i, ofType
 - select, select_i
 - distinct
 - skip, skipWhile
 - take, takeWhile
 - groupBy, selectMany
 - orderBy, orderByDescending
- Аккумуляция:
 - aggregate, elect, elementAt
 - all, any, none
 - min, max
 - sum, average
 - firstOrDefault, lastOrDefault
- Работа с множествами:
 - concat, concatOne
 - union, intersect, except
 - zip
 - join, groupJoin
- Экспорт:
 - toArray, toSet
 - toDictionary, toLookup
- Дополнительно:
 - readBytes, readLines
 - keyValueEnumerator
 - stringByJoin-withSeparator
 - enumerateCharacters

Полезные ссылки

- Collections Programming Guide - <https://developer.apple.com/library/mac/documentation/cocoa/conceptual/Collections/Collections.html>
- NSSortDescriptor - <http://nshipster.com/nssortdescriptor/>
- NSPredicate - <http://nshipster.com/npredicate/>
- NSExpression - <http://nshipster.com/nsexpression/>
- NSOrderedSet - <http://nshipster.com/nsorderedset/>
- Key-Value Collection Operators — <https://developer.apple.com/library/mac/documentation/Cocoa/Conceptual/KeyValueCoding/Articles/CollectionOperators.html>
- <https://github.com/robrix/RxCollections>
- <https://github.com/k06a/NSEnumeratorLinq>



One more thing...

- Оператор `if-null`

- Код на C#:

```
string priceStr = priceHolder.get(featureName);  
PriceLabel.Text = priceStr ?? "$0.99";
```

- Код на Obj-C:

```
NSDictionary *prices = [self.storeManager pricesDictionary];  
NSString *p1 = prices[@"com.identifier1"] ?: @"$0.99";
```