



作者 南望青天 (/users/ba7700edc70b) 2015.07.24 23:28*

写了12322字，被127人关注，获得了70个喜欢

(/users

+ 添加关注 (/sign_in)

关于EnumerateObjectsUsingBlock和for-in之间的较量

字数769 阅读1129 评论0 喜欢3

如果我们要遍历一个数组，上过编程课程的童鞋都会想到For语句去循环。
Objective C 提供一个Block的遍历方法，那么用它还是用For好呢？

下面我们去实践一下：

遍历一个数组看谁快

参赛选手 ForLoop，For - in，enumerateObjectsUsingBlock 这三个方法：

NSMutableArray *test = [NSMutableArray array];

[登录\(/sign_in\)](#) [注册\(/sign_up\)](#)

(/)

⋮

(/collections)

📱

(/apps)

```
for (int i = 0; i < 1000000; i++) {
    [test addObject:@(i)];
}

__block int sum = 0;
double date_s = CFAbsoluteTimeGetCurrent();
for (int i = 0; i < test.count; i++) {
    sum += [test[i] integerValue];
}
double date_current = CFAbsoluteTimeGetCurrent() - date_s;
NSLog(@"Sum : %d ForLoop Time: %f ms",sum,date_current * 1000);

sum = 0;
date_s = CFAbsoluteTimeGetCurrent();
for (NSNumber *num in test) {
    sum += [num integerValue];
}
date_current = CFAbsoluteTimeGetCurrent() - date_s;
NSLog(@"Sum : %d For-in Time: %f ms",sum,date_current * 1000);

sum = 0;
date_s = CFAbsoluteTimeGetCurrent();
[test enumerateObjectsUsingBlock:^(id obj, NSUInteger idx, BOOL *stop) {
    sum += [obj integerValue];
}];
date_current = CFAbsoluteTimeGetCurrent() - date_s;
NSLog(@"Sum : %d enumrateBlock Time: %f ms",sum,date_current * 1000);
```

最后输出如下:

```
2015-07-24 16:47:58.899 EnmuTest[74954:280032] Sum : 1783293664 ForLoop Time: 31.885982 ms
2015-07-24 16:47:58.920 EnmuTest[74954:280032] Sum : 1783293664 For-in Time: 20.569980 ms
2015-07-24 16:47:58.973 EnmuTest[74954:280032] Sum : 1783293664 enumrateBlock Time: 52.585959 ms
```

打印输出

• For-in方法最快速

结论:

当只是遍历一个数组的时候使用For-in会比较快速, 推荐使用For-in遍历数组.

通过Value查找Index看谁快

假如现在我们要查找一个Value, 这个Value 值是100001, 找出它的index (数组的序号).

简

(/)

::

(/collections)

☑

(/apps)

参赛选手 For - in, enumerateObjectsUsingBlock, enumerateObjectsWithOptions 这个三

个方法: (ForLoop已经不再继续讨论了)

```

NSMutableDictionary *test = [NSMutableDictionary array];
for (int i = 0; i < 10000000; i++) {
    [test addObject:@(i + 10)];
}

//For-in
__block NSInteger index = 0;
double date_s = CFAbsoluteTimeGetCurrent();
for (NSNumber *num in test) {
    if ([num integerValue] == 9999999) {
        index = [test indexOfObject:num];
        break;
    }
}
double date_current = CFAbsoluteTimeGetCurrent() - date_s;
NSLog(@"index : %ld For-in Time: %f ms", (long)index, date_current * 1000);

//enumerateObjectsUsingBlock
index = 0;
date_s = CFAbsoluteTimeGetCurrent();
[test enumerateObjectsUsingBlock:^(id num, NSUInteger idx, BOOL *stop) {
    if ([num integerValue] == 9999999) {
        index = idx;
        *stop = YES;
    }
}];
date_current = CFAbsoluteTimeGetCurrent() - date_s;
NSLog(@"index : %ld enumerateBlock Time: %f ms", (long)index, date_current * 1000);

//enumerateObjectsWithOptions
index = 0;
date_s = CFAbsoluteTimeGetCurrent();
[test enumerateObjectsWithOptions:NSEnumerationConcurrent usingBlock:^(id num, N
    if ([num integerValue] == 9999999) {
        index = idx;
        *stop = YES;
    }
}];
date_current = CFAbsoluteTimeGetCurrent() - date_s;
NSLog(@"index : %ld enumerateObjectsWithOptions Time: %f ms", (long)index, date_cu

```

最后输出如下图:

A



简
(/)

• enumerateObjectsWithOptions方法最快速

(/collections) 结论:



(/apps)

通过Value查询index的时候, 面对大量的数组推荐使用 enumerateObjectsWithOptions 的并行方法.

For-in 和 enumerateObjectsWithOptions 方法这里我比较喜欢第二种写法简洁直观.

现在咱们要遍历字典

这里我们比较一下使用 For-in 和 enumerateKeysAndObjectsUsingBlock 这两个方法:

```
NSDictionary *testDictionary = @{
    @"Auther" : @"南望青天",
    @"Game" : @"Dota",
    @"App" : @"麦刀塔",
    @"Market" : @"AppStore"
};

//For - in
NSMutableArray *forInArray = [NSMutableArray array];
double date_s = CFAbsoluteTimeGetCurrent();
NSArray *keys = [testDictionary allKeys];
for (NSString *key in keys) {
    NSString *Value = testDictionary[key];
    [forInArray addObject:Value];
}
double date_current = CFAbsoluteTimeGetCurrent() - date_s;
NSLog(@"index : %ld For-in Time: %f ms", (long)index, date_current * 1000);

//enumerateKeysAndObjectsUsingBlock
date_s = CFAbsoluteTimeGetCurrent();
NSMutableArray *enumArray = [NSMutableArray array];
[testDictionary enumerateKeysAndObjectsUsingBlock:^(id key, id obj, BOOL *stop) {
    [enumArray addObject:obj];
}];
date_current = CFAbsoluteTimeGetCurrent() - date_s;
NSLog(@"index : %ld For-in Time: %f ms", (long)index, date_current * 1000);

NSLog(@"ForInArr: %@", forInArray);
NSLog(@"enumArray: %@", enumArray);
```



打印输出:



```
2015-07-24 23:19:15.530 EnmuTest[3506:479580] index : 4389219473 For-in Time: 0.056982 ms
2015-07-24 23:19:15.531 EnmuTest[3506:479580] index : 4389219473 For-in Time: 0.012961 ms
2015-07-24 23:19:15.531 EnmuTest[3506:479580] ForInArr: (
    Data,
    "\U5357\U671b\U9752\U5929",
    "\U9ea6\U5200\U5854",
    AppStore
)
2015-07-24 23:19:15.531 EnmuTest[3506:479580] enumArry: (
    Data,
    "\U5357\U671b\U9752\U5929",
    "\U9ea6\U5200\U5854",
    AppStore
)
```

[打印输出](#)

• enumerateKeysAndObjectsUsingBlock胜出

结论:

当我们想遍历字典类型的时候, 推荐使用 enumerateKeysAndObjectsUsingBlock

不仅仅是因为速度快, 更是因为代码更优雅和直观.

➕ 推荐拓展阅读 (/sign_in)

如果觉得我的文章对您有用, 请随意打赏。您的支持将鼓励我继续创作!

¥ 打赏支持

♡ 喜欢 | 3

👤 分享到微博 🗨 分享到微信 更多分享 ▾

0条评论 (按时间正序 · 按时间倒序 · 按喜欢排序)

✎ 添加新评论 (/sign_in)

登录后发表评论 (/sign_in)

A

被以下专题收入, 发现更多相似内容:

(/)

(/collection/NEt52a) · 42909人关注



(/collections



iOS开发 (/collection/1cb96ea7b540)

271篇文章 (/collection/1cb96ea7b540) · 3707人关注

(/collection



(/apps)



iOS移动开发 (/collection/db63a204c1a3)

iOS移动开发相关

(/collection/493篇文章 (/collection/db63a204c1a3) · 2278人关注 /db63a204c1a3)

+ 添加关注 (/sign_in)

+ 添加关注 (/sign_in)

