

罗朝辉（飘飘白云）

关注C++，iOS，Android，3D 图形图像以及嵌入式系统开发。个人博客：<http://kesalin.github.io>(翻墙可见)

目录视图

摘要视图

RSS 订阅

个人资料



飘飘白云

访问：990143次

积分：11653

等级：BLOG 7

排名：第455名

原创：179篇 转载：45篇

译文：10篇 评论：476条

我的微博

微博



kesalin

加关注

博客专栏



OpenGL ES 2.0
iOS教程

文章：9篇

阅读：83157



深入浅出Cocoa

文章：26篇

阅读：290130

文章搜索

文章分类

iOS 开发 (62)

C/C++ (21)

Markdown博文大赛清新开启

天天爱答题 一大波C币袭来

寻找Java大牛!

大数据完美组合Spark+Scala, 这样学才够值!

[Cocoa]XCode的一些调试技巧

分类：iOS 开发

2012-01-31 16:02

16538人阅读

评论(6)

收藏

举报

xcode

cocoa

exception

printing

immutable

crash

XCode的一些调试技巧

罗朝辉 (<http://blog.csdn.net/kesalin/>)

CC 许可，转载请注明出处

XCode 内置GDB，我们可以在命令行中使用 GDB 命令来调试我们的程序。下面将介绍一些常用的命令以及调试技巧。

po 命令：为 print object 的缩写，显示对象的文本描述（显示从对象的 description 消息获得的字符串信息）。

比如：



上图中，我使用 po 命令显示一个 NSDictionary 的内容。注意在左侧我们可以看到 dict 的一些信息：3 key/value pairs，显示该 dict 包含的数据量，而展开的信息显示 isa 层次体系（即class 和 metaclass结构关系）。我们可以右击左侧的 dict，选中“Print Description of “dict””，则可以在控制台输出 dict 的详细信息：

```
[cpp] view plain copy print ?
01. Printing description of dict:
02. <CFBasicHash 0x1001149e0 [0x7fff7e27ff40]>{type = immutable dict, count = 3,
03. entries =>
04.   0 : <CFString 0x100002458 [0x7fff7e27ff40]>
    {contents = "first"} = <CFString 0x100002438 [0x7fff7e27ff40]>{contents = "one"}
05.   1 : <CFString 0x100002498 [0x7fff7e27ff40]>
```

<http://blog.csdn.net/kesalin/article/details/7222153>

1/4

- Android (8)
- 3D技术 (21)
- 移动开发 (42)
- OpenGL ES (9)
- 算法 (9)
- Windows (33)
- 游戏开发 (6)
- Java (7)
- .NET (2)
- 游戏人生 (4)
- 读书新知 (5)
- 开发环境 (16)
- Mac 技巧 (5)
- 医疗开发 (3)
- 网络技术 (8)
- 脚本语言 (4)
- 软件工程 (6)

文章存档

- 2014年10月 (2)
- 2014年07月 (5)
- 2014年06月 (2)
- 2014年04月 (4)
- 2014年03月 (1)

展开

最新评论

- [Cocoa]深入浅出 Cocoa 多线程之软件攻城师: 以下代码运行结果为: trueint main(int argc, const char * argv...
[Cocoa]深入浅出 Cocoa 多线程之软件攻城师: 这个例子有问题dispatch_block_t block; if (x) { bl...
[深入浅出Cocoa]iOS网络编程之!阿高: 为啥我这样连接, 但是代码直接跳过了connect这个函数, 我在connect之前和之后都打印日志, 结...
Dicom Image Viewer for iPad baidu_15521413: 大神。这个真机测试会内存警告, 导致杀死程序, 我测试了下 是这个方法的问题BOOL result = ...
[OpenGL ES 01]OpenGL ES之小小搬码工: 非常感谢博主的分享, 写得很好.我转载到自己的博客里了..再次感谢博主。
【转】阴影锥(shadow volume)原職熱天竺: 0.0 图挂了! 原链接也是!
[iOS]在 iOS 平台上编译 DCMTK oEveryone: @huamei_520:换 CMake版本, 根据xcode提示该错误。(前面加上this->)
[Swift]数组恒等 === 的 bug?! zhao_92221: === 号是判断地址的! == 号是判断内容的! 而 Swift中数组类是 值类型 和OC不同, 它不是引用...
[OpenGL ES 04]3D变换实践篇: 去寻找1991: 模型变换和视图变换应该是一个意思吧,效果是相反的是不是?模型变换是物体自己的旋转缩放移动,视角变换是...
[深入浅出Cocoa]详解键值观察 (micro_cai: @wzzvictory_tjsd:自动和手动的概念在 objc.io 有提到, http://objc...

阅读排行

- [Cocoa]深入浅出 Cocoa

```
06. {contents = "second"} = <CFString 0x100002478 [0x7fff7e27ff40]>{contents = "two"}
    2 : <CFString 0x1000024d8 [0x7fff7e27ff40]>
07. {contents = "third"} = <CFString 0x1000024b8 [0x7fff7e27ff40]>{contents = "three"}
08. }
```

print 命令: 有点类似于格式化输出, 可以输出对象的不同信息:

如:

```
[cpp] view plain copy print ?
01. (gdb) print (char *)[[dict description] cStringUsingEncoding:4]
02. $1 = 0x1001159c0 "{\n first = one;\n second = two;\n third = three;\n}"
03. (gdb) print (int)[dict retainCount]
04. $2 = 1
05. (gdb)
```

注: 4是 NSStringEncoding 的值。

info 命令: 我们可以查看内存地址所在信息

比如 "info symbol 内存地址" 可以获取内存地址所在的 symbol 相关信息:

```
[cpp] view plain copy print ?
01. (gdb) info symbol 0x00000001000017f7
02. main + 343 in section LC_SEGMENT.__TEXT.__text of /Users/LuoZhaohui/Library/Developer/Xcode/Derive
anzd1hiw1bizpfureuvenvmatnp/Build/Products/Debug/RunTimeSystem
```

比如 "info line *内存地址" 可以获取内存地址所在的代码行相关信息:

```
[cpp] view plain copy print ?
01. (gdb) info line *0x00000001000017f7
02. Line 62 of "/Users/LuoZhaohui/Documents/Study/RunTimeSystem/RunTimeSystem/main.m" starts at address
```

show 命令: 显示 GDB 相关的信息。如: show version 显示GDB版本信息

```
[cpp] view plain copy print ?
01. (gdb) show version
02. GNU gdb 6.3.50-20050815 (Apple version gdb-1708) (Mon Aug 8 20:32:45 UTC 2011)
03. Copyright 2004 Free Software Foundation, Inc.
04. GDB is free software, covered by the GNU General Public License, and you are
05. welcome to change it and/or distribute copies of it under certain conditions.
06. Type "show copying" to see the conditions.
07. There is absolutely no warranty for GDB. Type "show warranty" for details.
08. This GDB was configured as "x86_64-apple-darwin".
```

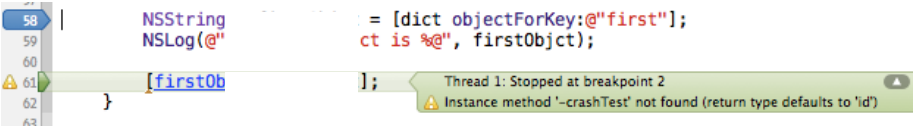
help 命令: 如果忘记某条命令的语法了, 可以使用 help 命令名 来获取帮助信息。如: help info 显示 info 命令的用法。

```
[cpp] view plain copy print ?
01. (gdb) help info
02. Generic command for showing things about the program being debugged.
03.
04. List of info subcommands:
05.
06. info address -- Describe where symbol SYM is stored
07. info all-registers -- List of all registers and their contents
08. info args -- Argument variables of current stack frame
09. info auxv -- Display the inferior's auxiliary vector
10. info breakpoints -- Status of user-settable breakpoints
11. info catch -- Exceptions that can be caught in the current stack frame
12. info checkpoints -- Help
13. info classes -- All Objective-C classes
14. ....
15.
16. Type "help info" followed by info subcommand name for full documentation.
17. Command name abbreviations are allowed if unambiguous.
18. (gdb)
```

- 使用System.arraycopy(): (57377)
- [OpenGL ES 01]OpenGL (30642)
- [Cocoa]深入浅出 Cocoa (24079)
- 说说ShellExecuteEx (19397)
- 在 Asp.NET MVC 中使用 (18084)
- [深入浅出Cocoa]iOS网络 (18028)
- [深入浅出Cocoa]iOS网络 (17212)
- [Cocoa]XCode的一些调 (16848)
- 在Mac下安装XAMPP (16530)
- (16485)

在系统抛出异常处设置断点

有时候我们的程序不知道跑到哪个地方就 crash 了，而 crash 又很难重现。保守的做法是在系统抛出异常之前设置断点，具体来说是在 objc_exception_throw处设置断点。设置步骤为：首先在 XCode 按 CMD + 6，进入断点管理窗口；然后点击右下方的 +，增加新的 Symbolic Breakpoint，在 Symbol 一栏输入：objc_exception_throw，然后点击 done，完成。这样在 Debug 模式下，如果程序即将抛出异常，就能在抛出异常处中断了。比如在前面的代码中，我让 [firstObjctcrashTest]; 抛出异常。在 objc_exception_throw 处设置断点之后，程序就能在该代码处中断了，我们从而知道代码在什么地方出问题了。



上一篇 [Cocoa]深入浅出Cocoa 之动态创建类
下一篇 深入分析Sleep(0)与Sleep(1)的区别

顶 11
踩 0

主题推荐 cocoa xcode objective-c 控制台 内存

猜你在找

- Xcode Scheme Apple Document 什么是Xcode 中的《iOS应用逆向工程》学习笔记三iOS文件权限
- iOS安全攻防八键盘缓存与安全键盘
- iOS开发笔记7基本控件
- iPhone开发十七多线程开发之
- 【精品课程】Spark+Scala套餐值不值，看你敢不敢学！
- 【精品课程】libGDX项目实战——2048
- 【精品课程】系统集成项目管理工程师-基础精讲班-第
- 【精品课程】Python开发实战（第二季）
- 【精品课程】Java Web进阶开发

准备好了么？跳 吧！ 更多职位尽在 CSDN JOB

网页设计师	我要跳槽	牛B的PHP程序猿 你在哪儿？	我要跳槽
广州良业贸易有限公司	8-16K/月	广州良业贸易有限公司	8-16K/月
IOS开发工程师	我要跳槽	高级测试工程师	我要跳槽
深圳咔嚓移动科技有限公司	8-15K/月	北京橘子文化传媒有限公司	10-20K/月

BERKELEY DATA SCIENCE
online.


REQUEST
INFO

- 查看评论
- 3楼 Zealot88 2013-08-20 14:17发表
- mark 只用过po
- 2楼 Chen_Yilong 2013-04-17 10:24发表
- 请问Exception Breakpoint与 Symbolic Breakpoint的区别在哪儿?,我点击Add Exception Breakpoint与点击Add Symbolic Breakpoint 效果一样啊!
- Re: 飘飘白云 2013-04-17 16:03发表
- 回复Chen_Yilong: 符号断点，你可以指定第三方库内的符号，即使你没有第三方库的源码。而异常断点针对的是，异常。


你可以参考如下链接：
http://developer.apple.com/library/mac/#recipes/xcode_help-breakpoint_navigator/articles/adding_an_exception_breakpoint.html

http://developer.apple.com/library/ios/#recipes/xcode_help-breakpoint_navigator/articles/adding_a_symbolic_breakpoint.html


Re: [Chen_Yilong](#) 2013-04-17 18:15发表

 回复kesalin： 嗯,谢谢,我看看.

1楼 [linlinyao1](#) 2012-05-15 13:34发表

 新的Xcode在断点管理窗口是点左下角的+

Re: [Chen_Yilong](#) 2013-04-17 10:22发表

 回复linlinyao1： 请问Exception 与 Symbolic Breakpoint的区别在哪儿？

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题	Hadoop	AWS	移动游戏	Java	Android	iOS	Swift	智能硬件	Docker	OpenStack		
VPN	Spark	ERP	IE10	Eclipse	CRM	JavaScript	数据库	Ubuntu	NFC	WAP	jQuery	
BI	HTML5	Spring	Apache	.NET	API	HTML	SDK	IIS	Fedora	XML	LBS	Unity
Splashtop	UML	components	Windows Mobile	Rails	QEMU	KDE	Cassandra	CloudStack	FTC			
coremail	OPhone	CouchBase	云计算	iOS6	Rackspace	Web App	SpringSide	Maemo				
Compuware	大数据	aptech	Perl	Tornado	Ruby	Hibernate	ThinkPHP	HBase	Pure	Solr		
Angular	Cloud Foundry	Redis	Scala	Django	Bootstrap							

公司简介 | 招贤纳士 | 广告服务 | 银行汇款帐号 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-600-2320 | 北京创新乐知信息技术有限公司 版权所有 | 江苏乐知网络技术有限公司 提供商务支持
京 ICP 证 070598 号 | Copyright © 1999-2014, CSDN.NET, All Rights Reserved 