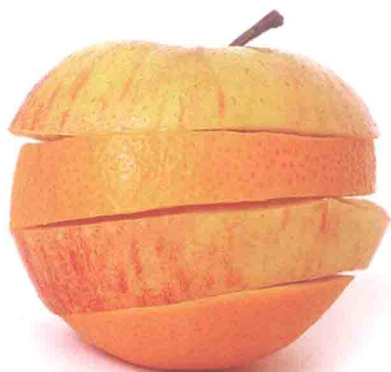


日本Objective-C圣经级教材



Objective-C

编程全解

(第3版)

[日] 荻原刚志 著 唐璐 翟俊杰 译

你能找到的最细致、最全面的Objective-C教程



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书



Objective-C

编程全解

(第3版)

[日] 荻原刚志 著 唐璐 翟俊杰 译

人民邮电出版社

北京

图书在版编目(CIP)数据

Objective-C编程全解:第3版/(日)荻原刚志著;
唐璐,翟俊杰译.--北京:人民邮电出版社,2015.1

(图灵程序设计丛书)

ISBN 978-7-115-37719-7

I. ①O… II. ①荻… ②唐… ③翟… III. ①C语言—
程序设计 IV. ①TP312

中国版本图书馆CIP数据核字(2014)第277728号

内 容 提 要

本书结合理论知识和实例程序,全面而系统地介绍了Objective-C编程的相关内容,包括类和继承、对象的类型和动态绑定、基于引用计数的内存管理、垃圾回收、属性声明、类NSObject和运行时系统、Foundation框架中常用的类、范畴、抽象类与类簇、对象的复制及存储、块对象、消息发送模式、图像视图、异常和错误、并行编程、键值编码等。

本书适合iOS应用和Mac OS X开发初学者系统入门、有经验的开发者深入理解语言本质,也适合开发团队负责人、项目负责人作为综合性的Objective-C参考书阅读。

-
- ◆ 著 [日]荻原刚志
译 唐 璐 翟俊杰
责任编辑 乐 馨
执行编辑 杜晓静
责任印制 杨林杰
- ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京天宇星印刷厂印刷
- ◆ 开本:800×1000 1/16
印张:28.75
字数:731千字 2015年1月第1版
印数:1-3 000册 2015年1月北京第1次印刷
著作权合同登记号 图字:01-2013-2656号
-

定价:79.00元

读者服务热线:(010)51095186转600 印装质量热线:(010)81055316

反盗版热线:(010)81055315

广告经营许可证:京崇工商广字第0021号

译者序

Objective-C 近年来一直备受关注，它诞生于 1983 年，设计之初和 C++ 一样都是为给 C 语言加入面向对象的功能。在 iPhone 发布之前，Objective-C 一直都是一门默默无闻的小众语言，而 C++ 则长期占据编程语言排行榜的前几名。随着 iPhone 销量的爆发性增长，Objective-C 语言的份额也迅速增长。在 TIOBE 编程语言排行榜上，Objective-C 只用了短短的 3 年时间就从 2009 年 8 月的第二十名增长到了 2012 年 7 月的第三名（仅次于 C 语言和 Java），截至今天仍然排名第三。

因为 Objective-C 流行的时间还比较短，所以就算是开发过多个上架 App 的 iOS 程序员也仍可能不完全了解 Objective-C 语言。例如，不知道消息转发背后的工作机制，不了解 Objective-C 几种内存管理之间的区别，不清楚该如何选择 Delegate、Notification 和 KVO 等。

本书是一本专门讲解 Objective-C 的图书，虽然作者并不会教你如何使用 Xcode 开发一个运行在 iPhone 上的 App，但用了 20 章的内容深入讲解了 Objective-C 的各个特性。

2014 年苹果公司在 WWDC 2014 开发者大会上发布了用于 Mac OS X 和 iOS 编程的新一代编程语言 Swift，相信很多读者一定在为应该学习 Objective-C 还是应该直接学习 Swift 而感到困惑。Swift 于 2014 年才发布，而目前市面上至少有上百万个 App 和类库都是用 Objective-C 编写的，如果你未来的工作是为现有 App 添加新的功能（这种工作的可能性很大），那你一定要懂得 Objective-C。

除了讲解 Objective-C 语言本身外，本书还介绍了与其密不可分的 Foundation 框架，介绍了字符串、数组、字典等常用类。

本书的作者获原刚志教授从 iPhone 还没诞生的 2001 年就开始写 Mac OS X 上的 Objective-C 编程入门的书籍，是 Objective-C 方面真正的专家。作者写这本书时把日本人认真严谨的天性发挥到了极致，用了 400 多页来讲解 Objective-C。本书已经再版了三次，本次翻译的就是最新的第三版。作者在讲述每个知识点的时候都精心配置了示例代码，所有的示例代码均可运行。相信通过对本书的学习，你一定能够打下坚实的 Objective-C 基础，开发出更美妙的 App。

本书由唐璐、翟俊杰翻译。在翻译的过程中得到了图灵编辑的大力帮助，在此深表谢意。由于时间仓促，译者水平有限，错误与疏漏之处在所难免，敬请读者谅解并批评指正。

序言

Objective-C 是一门为 C 语言增加了面向对象功能的语言，是开发 Mac OS X、iPhone、iPod touch 和 iPad 应用的主要语言。

在为 C 语言加入了面向对象功能的语言中，C++ 是最有名的一种。Objective-C 和 C++ 完全不同。Objective-C 和大家所熟知的 Java、C# 和 Ruby 也有所不同，是一门比较独特的语言。

Objective-C 最大的特点是支持面向对象编程，具备很多动态语言才有的动态特征，同时在效率上还可以媲美 C 语言。学习过其他面向对象语言的人可能会对 Objective-C 为 C 语言添加的功能之少感到惊讶。

随着 Mac OS X 和 iOS 的逐步更新，Objective-C 运行的系统环境也做了同步升级。同时 Objective-C 语言本身也引入了不少新的特性，包括一种新的内存管理方式——ARC。另外属性声明和代码块（block）的使用范围也得到了扩大。Objective-C 的编程风格这些年一直在不停地更新。

本书以开发 Mac OS X 或 iOS 应用为目的来介绍 Objective-C，默认读者具备 C 语言基础，但并不要求精通 C 语言。

Objective-C 和苹果公司的产品和运行环境紧密相关，介绍 Objective-C 的时候无法脱离具体的操作系统或框架。本书在介绍语言本身的同时也介绍了 Foundation 框架中的主要类，同时也尽可能指出了 Mac OS X 和 iOS 的不同之处。

另外，本书并不是一本讲解 Mac OS X 和 iPhone 图形界面编程的书，不会涉及 GUI 控件的使用，所以并不是说读了本书就能立刻做出一个具有优美界面的应用程序。已经有太多的优秀书籍介绍 GUI 编程的方方面面，请参考这些图书的内容。

Xcode 是苹果公司向开发人员提供的集成开发环境，用于开发 Mac OS X 和 iOS 的应用，其中自带了 Objective-C 的编译器。到笔者写作本书为止，Xcode 可以免费下载安装（只能安装在苹果系统中，没有 Windows 版）。而且只要加入苹果公司的开发者计划，注册成为 Apple Developer，就可免费获取创建 iOS 应用和 Mac 应用的资源，包括开发工具、示例代码、技术文档等。如果想在 iPhone 或 iPad 的真机上测试自己开发的应用并发布到 Apple Store，则需要付费加入苹果公司的 iOS 开发者计划。

本书中的代码都是终端类型的程序，只要安装了 Xcode，不需要对 Xcode 作任何设置，就可以编译、运行示例程序。对学习的内容有任何疑问时都可以通过运行程序来找到答案。

本书是《Objective-C 编程全解》的全新修订版本，新增了以下内容。

此为试读，需要完整PDF请访问：www.ertongbook.com

- 详细介绍了 Objective-C 新引入的内存管理方式 ARC 的方方面面，从工作原理到编程时的各种注意事项。以使用 ARC 为前提对示例程序做了大幅修改，同时简化了手动内存管理和垃圾回收方面的说明。
- 介绍了新的属性声明方法，并使用了新的方法重写了示例程序。
- 追加了使用 ARC 时 Core Foundation 对象和 Objective-C 对象之间如何进行类型转换的内容。

本书中讲解的所有内容都经过了实际编码测试。本书的代码在 Mac OS X 10.7 和 iOS 5 以上的环境下能正常执行。书中源代码可以从以下链接下载。

<http://download.sbcr.jp/getDLService.php?id=c2e364e9c1c1181de9c303520a68493a>

另外，本书还提供了以下三部分内容作为附录，帮助读者理解。

- 附录 A Foundation 框架的概要
- 附录 B Core Foundation 框架概要
- 附录 C 编码原则

请读者在开始阅读本书前自行下载：

<http://www.ituring.com.cn/book/download/9f2a581b-81f3-4873-8b4c-82addf7211ab>

Mac OS X 和 iOS 的开发环境的可扩展性非常好，也很人性化，只要具备基础知识就能够开发出简单的应用。但如果要开发一个真正的商用应用，还需要具备系统化的 Objective-C 知识，因为运行环境是为了最大限度地发挥 Objective-C 的特征而设计的。通过本书学习了 Objective-C 的基础知识之后，你也就可以更加得心应手地使用其他框架了。

开始享受你的 Objective-C 编程之旅吧！

2011 年 10 月 5 日，Apple 公司（NeXT 公司）的缔造者乔布斯逝世。乔布斯总能像魔法师一样给我们展示一些令人难以置信的新产品，真心感谢他带给我们的一切。一路走好！

版权声明

SHOUKAI OBJECTIVE-C 2.0, THE THIRD EDITION

Copyright © 2011 Takeshi Ogihara

Originally published in Japan by SB Creative Corp.

Chinese (in simplified character only) translation rights arranged with

SB Creative Corp., Japan through CREEK & RIVER Co., Ltd

All rights reserved.

本书中文简体字版由 SB Creative Corp. 授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

目录

第 1 章 面向对象的编程

1

1.1	对象的概念	2
1.1.1	面向功能的软件设计的局限性	2
1.1.2	面向对象的模块化	2
1.1.3	消息传递	3
1.1.4	模块的抽象化	4
1.1.5	对象的属性	5
专栏	面向对象的方方面面	5
1.1.6	类	6
专栏	Cocoa 和 Objective-C 的历史	8
1.2	模块和对象	8
1.2.1	软件模块	8
1.2.2	高独立性的模块	9
1.2.3	模块的信息隐蔽	9
1.2.4	类的定义和接口	10
1.2.5	消息发送的实现	10
专栏	C 语言的新标准	11

第 2 章 Objective-C 程序设计

13

2.1	对象和消息	14
2.1.1	消息表达式	14
2.1.2	消息选择器	15
2.1.3	实例变量的生成和初始化	16
2.2	类的定义	17
2.2.1	类的接口声明	17
专栏	各种各样的布尔类型	18
专栏	不指定方法的返回值	18

2.2.2	类的实现	19
2.2.3	一个遥控器的例子	19
2.3	编译	21
2.3.1	简单的编译方法	21
2.3.2	多文件编译	23
2.4	程序的书写风格	24
2.4.1	混合编程	24
2.4.2	C 语言函数的使用方法	25
2.4.3	静态变量的定义	26
2.4.4	头文件的引入	26
专栏	参考文档和 SDK	27
专栏	Xcode 的安装	28

第 3 章 类和继承

29

3.1	继承的概念	30
3.1.1	父类和子类	30
3.1.2	类的层次结构	31
3.2	利用继承定义新类	32
3.2.1	继承的定义	32
3.2.2	类定义和头文件	33
3.2.3	继承和方法调用	34
3.2.4	调用父类的方法	34
3.2.5	初始化方法的定义	35
3.3	使用继承的程序示例	36
3.3.1	追加新方法的例子	36
3.3.2	方法重写的例子	37
3.4	继承和方法调用	38
3.4.1	使用 self 调用方法	38
3.4.2	使用 super 调用方法	39
3.4.3	测试程序	40
3.5	方法定义时的注意事项	41
3.5.1	局部方法	41
3.5.2	指定初始化方法	42

专栏 Objective-C 与开源软件	44
----------------------------	----

第 4 章 对象的类型和动态绑定 45

4.1 动态绑定	46
4.1.1 什么是动态绑定	46
4.1.2 多态	47
4.2 作为类型的类	48
4.2.1 把类作为一种类型	48
4.2.2 空指针 nil	49
专栏 关系表达式	50
4.2.3 静态类型检查	51
4.2.4 静态类型检查的总结	53
4.3 编程中的类型定义	54
4.3.1 签名不一致时的情况	54
专栏 重载	55
4.3.2 类的前置声明	56
4.3.3 强制类型转换的使用示例	57
4.4 实例变量的数据封装	58
4.4.1 实例变量的访问权限	58
4.4.2 访问器	60
4.4.3 实例变量的可见性	61
4.4.4 在实现部分中定义实例变量	62
4.5 类对象	63
4.5.1 什么是类对象	63
4.5.2 类对象的类型	64
4.5.3 类方法的定义	65
4.5.4 类变量	65
4.5.5 类对象的初始化	66
4.5.6 初始化方法的返回值	68

第 5 章 基于引用计数的内存管理 69

5.1 动态内存管理	70
------------------	----

5.1.1	内存管理的必要性	70
5.1.2	引用计数、自动引用计数和自动垃圾回收	70
5.2	手动引用计数内存管理	71
5.2.1	引用计数	71
5.2.2	测试引用计数的例子	73
5.2.3	释放对象的方法	74
5.2.4	访问方法和对象所有权	75
专栏	静态对象	75
5.2.5	自动释放	76
5.2.6	使用自动释放池时需要注意的地方	77
5.2.7	临时对象的生成	77
5.2.8	运行回路和自动释放池	78
5.2.9	常量对象	78
专栏	常量修饰符 const	79
5.3	分数计算器的例子	80
5.3.1	分数类 Fraction	80
5.3.2	保存计算结果的 FracRegister 类	83
5.3.3	主函数和执行示例	85
5.4	ARC 概要	88
5.4.1	什么是 ARC	88
5.4.2	禁止调用引用计数的相关函数	89
5.4.3	管理自动释放池的新语法	90
5.4.4	变量的初始值	90
5.4.5	方法族	90
5.4.6	方法 dealloc 的定义	92
5.4.7	使用 ARC 的程序的编译	93
5.4.8	ARC 的基本注意事项	94
5.4.9	使用 ARC 重构分数计算器	94
5.5	循环引用和弱引用	95
5.5.1	循环引用	95
5.5.2	所有权和对象间的关系	96
5.5.3	弱引用	97
5.5.4	自动 nil 化的弱引用	98
5.5.5	对象之间引用关系的基本原则	99
5.6	ARC 编程时其他一些注意事项	100

5.6.1	可以像通常的指针一样使用的对象	100
5.6.2	setter 方法的注意事项	101
5.6.3	通过函数的参数返回结果对象	102
5.6.4	C 语言数组保存 Objective-C 对象	103
5.6.5	ARC 对结构体的一些限制	105
5.6.6	提示编译器进行特别处理	106

第 6 章 垃圾回收

107

6.1	垃圾回收的概要	108
6.1.1	查找不再使用的对象	108
6.1.2	编程时的注意事项	109
6.1.3	垃圾收集器	110
6.1.4	finalize 方法的定义	111
6.1.5	编译时的设定	112
6.1.6	引用计数管理方式中方法的处理	113
6.1.7	使用垃圾回收编程小结	114
6.2	垃圾回收的详细功能	114
6.2.1	分代垃圾回收	114
6.2.2	弱引用	115
6.2.3	自动 nil 化	115
6.2.4	通过垃圾回收回收动态分配的内存	116
6.2.5	__strong 修饰符的使用方法	117
6.2.6	NSGarbageCollector 类	117
6.2.7	实时 API	118
6.3	内存管理方式的比较	119
6.3.1	引用计数和垃圾回收	119
6.3.2	更改内存管理方式	120
6.3.3	各种内存管理方式的比较	120

第 7 章 属性声明

123

7.1	属性是什么	124
7.1.1	使用属性编程	124
7.1.2	属性的概念	125

专栏 自省	125
7.2 属性的声明和功能	126
7.2.1 显式声明属性	126
7.2.2 属性的实现	127
7.2.3 @synthesize 和实例变量	129
7.2.4 通过 @synthesize 生成实例变量	130
7.2.5 给属性指定选项	131
7.2.6 赋值时的选项	132
7.2.7 原子性	134
7.2.8 属性声明和继承	135
7.2.9 方法族和属性的关系	135
7.3 通过点操作符访问属性	136
7.3.1 点操作符的使用方法	136
7.3.2 复杂的点操作符的使用方法	137
7.3.3 何时使用点操作符	139

第 8 章 类 NSObject 和运行时系统

141

8.1 类 NSObject	142
8.1.1 根类的作用	142
8.1.2 类和实例	142
8.1.3 实例对象的生成和释放	143
8.1.4 初始化	144
8.1.5 对象的比较	144
8.1.6 对象的内容描述	145
8.2 消息发送机制	145
8.2.1 选择器和 SEL 类型	145
8.2.2 消息搜索	146
8.2.3 以函数的形式来调用方法	147
专栏 函数指针	148
8.2.4 对 self 进行赋值	149
8.2.5 发送消息的速度	149
8.2.6 类对象和根类	152
8.2.7 Target-action paradigm	153
8.2.8 Xcode 中的动作方法和 Outlet 的写法	155

8.3	Objective-C 和 Cocoa 环境	156
8.3.1	cocoa 环境和 Mac OS X	156
8.3.2	Cocoa Touch 和 iOS	156
8.3.3	框架	157
8.3.4	框架的构成和头文件	157
8.4	全新的运行时系统	159
8.4.1	对 64 位的对应和现代运行时系统	159
8.4.2	数据模型	159
8.4.3	64 位模型和整数类型	159
8.4.4	Core Graphics 的浮点数类型	160
8.4.5	健壮实例变量	161
专栏	条件编译	162

第 9 章 Foundation 框架中常用的类

163

9.1	对象的可变性	164
9.1.1	可变对象和不可变对象	164
9.1.2	可变对象的生成	165
9.2	字符串类 NSString	166
9.2.1	常量字符串	166
9.2.2	NSString	167
9.2.3	NSMutableString	174
9.3	NSData	176
9.3.1	NSData	176
9.3.2	NSMutableData	178
9.4	数组类	179
9.4.1	NSArray	179
9.4.2	NSMutableArray	183
9.4.3	数组对象的所有权	184
9.4.4	快速枚举	185
9.4.5	枚举器 NSEnumerator	186
9.4.6	快速枚举和枚举器	187
9.4.7	集合类	188
9.5	词典类	189

9.5.1	NSDictionary	191
9.5.2	NSMutableDictionary.....	193
9.6	包裹类	194
9.6.1	NSNumber	194
9.6.2	NSNumber	196
9.6.3	类型编码和 @encode()	196
9.6.4	NSNumber	197
9.7	NSURL	198
9.7.1	关于 URL	198
9.7.2	NSURL 的概要	199
9.7.3	使用 NSURL 来访问资源	201

第 10 章 范畴

203

10.1	范畴	204
10.1.1	范畴	204
10.1.2	范畴和文件的组织	205
10.1.3	作为子模块的范畴	206
10.1.4	方法的前向声明	207
10.1.5	私有方法	208
10.1.6	类扩展	209
10.1.7	范畴和属性声明	210
10.2	给现有类追加范畴	211
10.2.1	追加新的方法	211
10.2.2	追加方法的例子	212
	专栏 可变参数的方法的定义	213
10.2.3	覆盖已有的方法	214
10.3	关联引用	215
10.3.1	关联引用的概念	215
10.3.2	添加和检索关联	215
10.3.3	对象的存储方法	216
10.3.4	断开关联	217
10.3.5	利用范畴的例子	217

第 11 章 抽象类和类簇**221**

11.1	抽象类	222
11.1.1	什么是抽象类	222
11.1.2	抽象类的例子	223
11.2	类簇	228
11.2.1	类簇的概念	228
11.2.2	测试程序	229
11.2.3	编程中的注意事项	230
11.3	生成类簇的子类	231
11.3.1	使用范畴	231
11.3.2	基本方法的重定义	231
11.3.3	生成字符串的子类	233

第 12 章 协议**235**

12.1	协议的概念	236
12.1.1	什么是协议	236
12.1.2	对象的协议	236
12.2	Objective-C 中协议的声明	238
12.2.1	协议的声明	238
12.2.2	协议的采用	239
12.2.3	协议的继承	240
12.2.4	指定协议的类型声明	240
12.2.5	协议的前置声明	241
12.2.6	协议适用性检查	241
12.2.7	必选功能和可选功能	242
12.2.8	使用协议的程序示例	242
专栏	类的多重继承	245
12.3	非正式协议	246
12.3.1	什么是非正式协议	246
12.3.2	非正式协议的用途	246
专栏	使用宏 (macro) 来区分系统版本的差异	247

第 13 章 对象的复制及存储

249

13.1	对象的复制	250
13.1.1	浅复制和深复制	250
13.1.2	区域	251
13.1.3	复制方法的定义	251
13.1.4	复制方法的例子	252
13.1.5	实现可变复制	254
13.2	归档	255
13.2.1	对象的归档	255
13.2.2	Foundation 框架的归档功能	255
13.2.3	归档方法的定义	256
13.2.4	归档的方法定义	257
13.2.5	归档和解档的初始化方法	258
13.3	属性表	259
13.3.1	属性表概况	259
13.3.2	ASCII 码格式属性表	260
13.3.3	XML 格式属性表	261
13.3.4	属性表的变换和检查	261

第 14 章 块对象

263

14.1	什么是块对象	264
14.1.1	C 编译器和 GCD	264
14.1.2	块对象的定义	264
14.1.3	块对象和类型声明	266
14.1.4	块对象中的变量行为	267
14.1.5	排序函数和块对象	269
14.2	块对象的构成	271
14.2.1	块对象的实例和生命周期	271
14.2.2	应该避免的编码模式	273
14.2.3	块对象的复制	273
14.2.4	指定特殊变量 __block	274
14.3	Objective-C 和块对象	276
14.3.1	方法定义和块对象	276