

如何学好 iphone 游戏开发

整理：qq963922432 和 apiter（泰然论坛管理组）

著作权声明：本文由 子龙山人 翻译，欢迎转载分享。请尊重作者劳动，转载时保留该声明和作者博客链接，谢谢！首发于[泰然论坛](#)

自从发布[《如何学习 iphone 游戏开发》](#)到现在，大概有半年时间了。其实在写《如何学习 iphone 游戏开发》的时候，我就写下豪言，说哪天学好了，再来写一篇《如何学好 iphone 游戏开发》。但是现在，当我坐下来写这篇文章的时候，心里还是有点忐忑不安的，怕写不好，或者水平不够，写出来的东西达不到预期的效果。但是，管他呢，我手写我口，我把自己这半年来的学习经历和心得体会，以后自己写博客的经历做了一些小结，希望能给后来人带来帮助。同时，我还会在文章的最后，为初学 iphone 游戏开发的人指明一些方向，为已经入室的朋友提供一个更好的切磋与交流平台。

首先，基础很重要。像 c/c++，特别是指针和内存管理部分，如果理解得不够深刻，那么学习起来会很吃力。不过，你可能会问，“搞 iphone 游戏开发不是用的 objective-c 么？”。没错，目前用的最多的应该是 objective-c，当然用 Unity 的除外（里面用的 c# 和 js）。但是，其实 objc 就是在 c 的基础之上封装了一层面向对象的特性，同时把我们平常熟知的后缀表达式改成中缀表达式了（也就是语法差异）。而且，但凡学习过 c/c++ 的人都知道，内存管理是非常难的，而且 bug 基本上都是出在内存上面。所以，导致很多人不喜欢 c 和 c++，转而投向 c# 和 java 的怀抱了。我在这里，不想再讨论哪种语言好了，这种话题讨论太多了，再讨论下去也没意思。每种语言都有它的优势与特长，有其应用的领域，有支撑它的公司或组织，有维护它的 fans。我们程序员要做的就是，针对问题领域的特点，选择合适的编程环境（包括语言、工具、SDK 等）。

那么 objc 要不要做内存管理呢？目前来说，是要的。但是，在 mac 上面，objc 2.0 引入了垃圾回收机制，而且新版本的 sdk ios5，引入了 ARC（自动引用计数功能），或许在将来，利用 objc 开发 iphone 上面的应用和游戏也不需要手动再管理内存了。不过这种自动管理机制肯定会带来运行效率的问题，这和一切的 gc 机制一样。所以，我们暂且观望一下，到底 ios5 的 ARC 机制表现如何吧。

c/c++ 的基础好的话，学习 objc 会非常快，至少我是花了 2 周时间就把 objc 学习完了。但是，如果要把 objc 的内存管理，也就是引用计数掌握的话，那还是需要时日的。这里推荐大家看一些书和 blogs。假如你的 c/c++ 基础不是很好的话，建立你在学习 objc 之前先补一点 c 和 c++ 相关的知识。《c 语言和指针》、《the c programming language》、《c++ primer》、《the c++ programming language》、《Thinking in c++》、《c++ 程序原理与实践》、《effective c++》、《more effective c++》、《inside c++ object model》等等。你上网搜一下，推荐 c++ 的书太多了，我这里只是列举了一部分比较经典的，当然，如果你把这些书都看完了。根据这些书后面的 Reference 还可以把其它的优秀 c 和 c++ 书籍找出来。我看书都是这样的，看完一本好书，然后按照这本书的参考文献去一本一本地看其实的书，然后看完另一本，又去看其参考文献。这样不仅扩大了自己的知识面，而且能够把握所学知识的来龙去脉。理解地越深刻，你写成程序来就会越得心顺手。当然，这里有这么多书，对了那些天天要加班的程序猿来说，真是伤不起啊。但是，这些东西迟早还是要补的，如果你还在读大学，那么更加要注重这方面了。

以后工作了，想静下心来学习真不是那么容易。虽然我也还在学校，但是，看看加班的筒子们天天在咆哮“天天加班有木有！”，想想搞 IT 的还真是伤不起啊！

但是，编程这玩意儿是慢慢积累起来的。就算你看完了这些书，不去实践，也是没什么很大用的。编程能力其实跟你重复的次数也有一定的关系，如果只会纸上谈兵，也不见得能编写出好的程序来。好了，说了这么多，转而来谈 objc 吧。如果你觉得 c/c++ 基础还不错，推荐你几本不错的 objc 书籍。《Programming in Objective-C 2.0》、《Learning Objective-C 2.0》、《objective-c 基础教程》，还有就是 apple 的一些技术手册啦（对了，顺便提一下，大家千万不要留言问我要电子书。你只需要打开 google，然后输入英文书籍名+pdf，就 ok 了！）。学习 objc 一定不能偷懒，一定要把引用计数来管理内存的方法掌握，因为，后面不管是做应用还是做游戏，80% 的问题，你都会卡在内存管理上面。各种 crash 有木有！各种 EXC_BAD_ACCESS 有木有！各种莫名其妙的程序挂掉有木有！（大家一定要搞清楚 Autorelease Pool，Autorelease Pool Stack，Main Loop，Ownership，这些概念非常重要，如果你现在还不是很明白的话，多看 apple 的 objective-c memory programming guide。）

然后，等你 objc 基础打好了（我其实是先做一些 iphone 的应用来练手的，因为后面做游戏，很多东西还是要用到 UIKit 和 Cocoa Foundation 框架），xcode 也摸熟悉了（能够掌握用 xcode 设置断点调试，能够使用 Zombie 解决内存问题，能够使用 instruments 来检测内存泄露和内存分配情况，以及 cpu 使用率情况，能够新建 target，搞得清 4.0 的 schema 等），这时候，你再去学习 iphone 游戏开发会容易一些。至少你对于开发环境和开发工具是熟悉了，遇到问题，自己也能够解决，而不是一天到晚去网上提问或者到群里咆哮了。其实很多时候，自己解决一个问题，比问别人轻而易举地得到答案收获要多得多。如果实在是自己非常不熟悉，搞了好久都没解决的问题，这时候你再拿出来。自己也会把问题描述得更清楚一些，而且和别人的讨论中，有时候也能启迪思路，豁然开朗。有些人喜欢说提问没人回答，其实有很多原因，一是你描述问题不清楚，二是别人也不知道答案，三可能是你根本就没有找到问题。所以，希望大家遇到问题，尽量自己想办法解决，这样绝对是有百益而无一害的。（关于工具的使用，我的博客上有几篇 Ray 写的 objective-c 的教程，大家可以看看，在 objective-c 分类里面）（做应用的话，还可以听一听斯坦福大学的 iphone 开发视频，现在网易公开课里有，大家可以关注下）

接下来，你是不是可以马上学习 iphone 游戏开发了？先别急，还有一些知识你需要补充一下——那就是编码规范和设计模式。编码规范非常重要，每个平台都有自己的编码风格和规范，如果你遵守这些规范的话，你写的代码能够容易被他人阅读和修改，而你阅读别人的代码也会觉得倍感亲切，因为它们长相很接近。这里我不详细讨论编码规范了，因为 google 已经写好了，[objective-c 编码规范](#)、[Apple's Cocoa Coding Guidelines](#)、[c++ 编码规范](#)。为什么又是 c++？因为后面学习游戏物理引擎，会用到 box2d，到时候是 objc 和 c++ 混合编程，有些东西需要注意一下。然后，我们谈一下设计模式。对于刚开始接触 iphone 开发的人来说，最好是先学习半个月左右的应用开发，这对你以后的游戏开发有好处。相信做过 iphone 应用的人都知道 MVC，在 apple 的世界里面，这个模式被广泛地应用，甚至每一个角落都是 MVC。其实，MVC 不仅仅是一种模式，一种组织代码的方式，更是一种软件设计的思想。后面我会向大家推荐 cocos2d，它是一个开源的 2d 游戏引擎。因为本身引擎级别的程序都包含了很多软件设计和代码组织的技巧在里面，所以，如果我们对设计模式有一些了解的话，接下来阅读 cocos2d 的源代码将会有巨大的帮助。对于设计模式，我推荐两本书，一本是 GOF 的《设计模式，可复用面向对象软件的基础》，还有一本是《cocoa 设计模式》。当然，如果是初识设计模式的话，最好是先看《Head first 设计模式》和《大话设计模式》。（读到这里，有童鞋可能会纳闷了，你一个学生怎么知道这么多？别忘了，我们有 google。

如果现在还不知道怎么充分利用网络资源的童鞋，请看看我写的[《如何查找资料--我的一些心得和体会》](#)吧！)

现在，(叹了一口气吧)让我们来谈谈怎么学习 cocos2d 吧！cocos2d 不仅仅是一个游戏图形引擎，它里面还集成了声音引擎和物理引擎，而且有活跃的社区和丰富的资料。最重要的是，它是开源的，免费的。cocos2d 怎么学习呢？之前，大家一般是推荐看知易写的系列教程，我也是这么干的。所以，我在《如何学习 iphone 游戏开发》里面也是这么建议大家的。知易的教程是永恒的经典，虽然所使用的 api 比较旧，所演示的例子也比较陈旧，造成很多初学者在实践的过程中，各种函数找不到，各种困惑。所以，我觉得，还是从 Ray 的教程入手会相对来说容易一些。Ray 写的都是英文的，不过，我已经把他的大部分教程翻译成中文了。

现在，我想重新整理一下学习 cocos2d iphone 游戏开发的过程。

从[\(译\)如何使用 cocos2d 来做一个简单的 iphone 游戏教程\(第一部分\)](#)这篇文章开始，顺藤摸瓜，把这个系列的教程3个部分都看完。然后找到 cocos2d-iphone 下载的程序里面的一个测试例子 TomTheTurret，它位于 CocoDension/TestAndDemos 目录下面。这个例子是 Ray 对这个系列教程的完善。

接下来，我觉得可以看看知易写的例子了，把坦克大战的游戏学会。知易的博客地址是：

- 1 [知易 Cocos2D-iPhone 游戏开发教程001 \(341\)](#)
- 2 [知易 Cocos2D-iPhone 游戏开发教程002 \(94\)](#)
- 3 [知易 Cocos2D-iPhone 游戏开发教程003 \(71\)](#)
- 4 [知易 Cocos2D-iPhone 游戏开发教程004 \(73\)](#)
- 5 [知易 Cocos2D-iPhone 游戏开发教程005 \(65\)](#)
- 6 [知易 Cocos2D-iPhone 游戏开发教程006 \(66\)](#)
- 7 [知易 Cocos2D-iPhone 游戏开发教程007 \(102\)](#)

大家记住，如果你使用的是最新版本的 cocos2d，那么知易的教程里面可能很多类名和函数名都改了，大家不要怕，其实只是一些很小的改动。比如 CCColourLayer 改成 CCLayerColor，把 CCXXXXTransition 改成 CCTransitionXXXX，把 CCXXXXAction 改成 CCActionXXX，同时把 CCSpriteSheet 改成了 CCSpriteBatchNode 等，像这样一些改动。其实只是一些微小的重构，方便大家更好地使用 cocos2d 引擎。这里，我顺便提一下，有很多人写代码的时候，对于一些变量名和类名的命名很不放在心上。想怎么命名就怎么弄，特别是发现在命名不好之后，还舍不得改。因为，代码引用原来不好的类名太多了，一一改过来很麻烦。但是，其实 Xcode 是提供 refactor 功能的。

接下来，大家可以看看

[\(译\)如何在 cocos2d 里面使用动画和 spritesheet](#)

[\(译\)cocos2d 精灵教程：第一部分](#)

[\(译\)cocos2d 菜单教程：第一部分](#)

[\(译\)如何使用 cocos2d 制作基于 tile 地图的游戏教程：第一部分](#)

这里写第一部分，说明是个系列教程，大家可以顺着教程最后的导航一路看过去。相信学习完这些东西以后，你对于 sprite, menu, layer, scene, spritesheet, spriteBatchNode, tiledMap 应该有一些认识和理解了。

然后，大家可以继承在我的博客上面学习，还有许多其它的教程。建立大家从初级游戏教程看起，然后再是中高级的，然后再看游戏物理分类的教程。

同时，这时候，大家应该去看一些书了。这里推荐2本书，之前也推荐过的《Learn iPhone and iPad Cocos2D Game Development》和《Leaning Cocos2d》。

如果大家把我的博客上面的翻译教程都看完了，同时，上面提到的两本书也看完了的话。当然，重要的是理解了，还要动手实践!!! 不要为了数量而乱来，欲速则不达！接下来，最好是拿一些项目来练练手。如果是为公司工作的人，那最好了，直接用公司的项目就可以练手。如果是个人学习，建立从已有的系列教程入手，因为很多图片和声音素材已经有了，你可以在此基础上，再扩展，实现更多更 cool 的功能。如果你愿意跟我分享你的成果的话，我可以把你的成果写成教程，这样就会有更多的人能从中获益啦。

那么接下来，何去何从呢？我觉得有以下2条路可供选择：

1.继续深入学习 cocos2d 框架，认真研读 cocos2d 源代码，体会作者的代码设计与实现思路。然后学习 opengles，尝试自己扩展 cocos2d 框架，在 cocos2d 的基础上定制出符合自己公司的框架出来。学习 opengles，我也推荐几本书《Learning iOS Game Programming》，《Oreilly.iPhone.3D.Programming.May.2010》，《OpenGL ES 2.0 Programming Guide》。

2.尝试一下 cocos2d-x。这是由国人主导的一个开源项目，采用 c++，能够跨 ios, android, 乐 phone, win32, linux 等多个平台。同时，目前 cocos2d-x 还提供了 lua 接口绑定。为什么要选择 cocos2d-x？现在 ios 上面竞争激烈，如果你的游戏能在多个平台上运行，相信吸金来源多个几也会有好处吧？

当然，如果大家想在游戏开发和学习这条路上走得更远的话，建议深入学习数学和物理、计算机图形学、人工智能、数据结构与算法等。我看过一个牛人的博客，说做游戏，最后就只剩下数学和物理了。而我自己在开发的过程中，也感觉数学非常之重要。

（好长啊，快点结束吧！）今后请大家一定要关注 [Ray 的博客](#)，关注 <http://www.learn-cocos2d.com/> 这个网站，此网站的作者正在制作一个新的游戏开发包 [Kobold2D](#)。他把目前与 cocos2d 相关的资源全部整合进 kobold2d 里面去了，现在还在 preview 阶段，相信发布以后会给开发人员带来不少便利。顺便提一下，此作者的网站里面有非常多的资源，上面导航链接里有，右边链接里也有，大家快去围观吧！当然，其实还有一个很重要的学习地方，那就是 cocos2d 官网：<http://www.cocos2d-iphone.org>，还有论坛，不要因为 E 文的就不去了，看 E 文才是王道！还有，我差点忘了，一定要学会用 git！

还有，为了避免版权问题，Ray 有说过只允许别人翻译他写的5篇教程，虽然我写了免责声明，但是，还是不爽。所以，以后我翻译 Ray 的教程会做一些“微创新”，用 cocos2d-x 重新实现一遍。因为现在 cocos2d-x 的资料实在是太少了。这样，如果大家想看 objc 版的，直接上 Ray 的网站上看就可以了，想看 c++版的，就看我翻译的吧。

最后，我想把自己的一些想法和打算跟大家说一下。我们几个朋友商量了一下，我们专门建立了一个网站（含论坛）：[泰然教程](#)。此网站是之前帮助我翻译过一些文章的 yy 同学提议建的。欢迎大家光临，多提宝贵建议！你们的支持将会是我们的最大动力！以后我翻译的教程都会在泰然论坛首发，然后1周之后再在博客园再发，望相互转告，谢谢！

泰然论坛团队由一群有共同目标的 ios 开发爱好者组成。我们发现国内的技术论坛都或多或少的缺少一些资源，尤其在做一些3D项目的和游戏的时候，尤其 OpenGL ES 的资源很是稀缺，所以萌生了一个念头：在泰然工作室手上的几个项目完成后，成立一个自己的论坛，为和我们有共同理想的开发者提供平台。今年 9 月，最后一个项目完成后，我们决定实现这个诺言。

论坛建立之初，我们定的方向主要是 OpenGL ES，因为我们的项目经验，可以写一些相关的教程，

让大家起步不像我们当年那样艰难。后来，我们把这个想法告诉了子龙山人，发现大家想法都很一致，经过一段时间的相互了解后，我们决定联手创造出一个全新的泰然论坛。

今后，利用之前项目中积累的经验以及测试设备，泰然论坛会给大家带来 OpenGL ES 教程，CocosD-iPnone 和 Cocos2D-x 教程（后期还会不断扩展）；不同于以前博客的是，我们会大力发展原创教程，也包括跨平台的 Cocos2D-x 教程。

首先，我们强调泰然论坛将是一个免费的开放性论坛，我们不会采取商业运作。

其次，论坛会有一些广告位，我们计划将广告位开放给大家，如果大家有新的 app，以后可以联系我们，我们会免费为你的 app 安排泰然论坛的广告，以帮助大家度过 app 最初的艰难过程。

最后，如果大家有好的建议或者意见，请告诉我们，这将让我们共同成长。联系邮箱：iven@ityran.com