移动平台应用开发——计算器

计算机与信息工程学院 2013级 软件工程 于丹丹 20131105783

指导教师 朝力萌 讲师

摘 要 iOS 是世界上最先进的移动平台，它重新定义了只有口袋大小的移动设备可以实现的功能。与 Xcode 集成开发环境工具集成的 iOS SDK 使得创建优秀的应用程序变得十分轻松。由于苹果 App Store 存在于每一个 iOS 操作系统设备上，并在世界各地都有本地化，因此对移动开发者们来说根本没有什么其他的平台比它更吸引人。Swift 是一种新的编程语言，用于编写 iOS 和 OS X 应用。Swift 采用安全的编程模式并添加了很多新特性，这将使编程更简单，更灵活，也更有趣。Swift 是基于成熟而且倍受喜爱得 Cocoa 和 Cocoa Touch 框架，他的降临将重新定义软件开发。

关键词 ios；Xcode集成开发环境；swift；

# 1引言

当今时代，是一个网络时代。传统的计算器已经渐渐的远离人们的实现，大多数应用计算器充斥在人们的生活中。计算器应用出现在手机、ipad、电脑，计算器应用在各行各业。20世纪末，电子技术获得了飞速的发展，在其推动下，现代计算器应用几乎渗透了社会的各个领域，有力地推动了社会生产力的发展和信息化程序的提高，同时计算器功能性能进一步提高，产品更新换代的节奏越来越快。

# 2 Swift简介

## 2.1历史

Swift是Apple公司于2014年6月2日发布的全新的开发语言，Swift吸收了众多现代编译语言的优点，尽量提供简洁的编程语言和强大的功能。2014-6-12《Swift中文版》第一版发布。2015年12月4日，苹果公司宣布其Swift编程语言现在开放源代码。长600多页的The Swift Programming Language可以在线免费下载。 Swift 是一种新的编程语言，用于编写 iOS 和 OS X 应用。Swift 结合了 C 和 Objective-C 的优点并且不受C兼容性的限制。Swift 采用安全的编程模式并添加了很多新特性，这将使编程更简单，更灵活，也更有趣。Swift 是基于成熟而且倍受喜爱得 Cocoa 和 Cocoa Touch 框架，他的降临也重新定义软件开发的方式。 Swift既支持面向过程的编程机制，也支持面向对象的编程方式，而且Swift 的面向对象功能很强大。Swift采用了类似JavaScript的定义变量的方式，例如如下代码，var name=”fkit.org”;

2.2多态

Swift引用变量有两个类型：一个是编译时类型，一个是运行时类型。编译时 类型由声明该变量时使用的类型决定，编译器只认每个变量的编译时类型；运行时类型由实际赋给该变量的实例决定。如果编译时类型和运行时类型不一致，就可能出现所谓的多态。例如，当把一个子类实例直接赋给父类引用变量时，在运行时调用该引用变量的方法时，其方法行为总是表现出子类方法的行为特征，而不是父类的行为特征，这就可能出现：相同类型的变量、调用同一个方法时呈现出多种不同的行为特征，这就是多态。 可以使用is运算符判断前面的引用变量是否引用后面的类，或者其子类、实 现类的实例。如果是，则返回true，否则返回false。 编写Swift程序时，引用变量只能调用其编译时类型的方法，而不能调用其 运行时类型的方法，即使它实际所引用的实例确实包含该方法。如果需要让这个引用变量调用其运行时类型的方法，则必须把它强制转换其实际的类型，这种强制类型转换也称为向下转型。向下转型需要借助向下转型运算符，Swift提供了向下转型运算符as强制将运算符前面的引用变量转换为后面的类型。例如,letobjStr:NString=obj as NSString。

# 

# 3主窗体设计



# 4 功能实现

4.1清除功能实现

@IBAction func 重置当前(sender: AnyObject) {

显示缓存 = ""

self.屏幕.text = ""

}

@IBAction func 重置所有(sender: AnyObject) {

显示缓存 = ""

缓存 = 0.0

缓存2 = nil

Calculator.设置当前算法(.未选择)

self.屏幕.text = ""

}

4.2计算功能实现

@IBAction func 算法加(sender: AnyObject) {

Calculator.设置当前算法(.加法)

求值()

}

@IBAction func 算法减(sender: AnyObject) {

Calculator.设置当前算法(.减法)

求值()

}

@IBAction func 算法乘(sender: AnyObject) {

Calculator.设置当前算法(.乘法)

求值()

}

@IBAction func 算法除(sender: AnyObject) {

Calculator.设置当前算法(.除法)

求值()

}

@IBAction func 算法平方(sender: AnyObject) {

Calculator.设置当前算法(.平方)

求值()

}

@IBAction func 算法指数幂(sender: AnyObject) {

Calculator.设置当前算法(.指数幂)

求值()

}

4.3函数算法

func 求值() {

if !显示缓存.isEmpty {

let 临时 = 显示缓存 as NSString

缓存 = 临时.doubleValue

显示缓存 = ""

}

let 临时 = 缓存

var 结果 = ""

if let 前一个数字 = 缓存2 {

结果 = Calculator.求结果(前一个数字, 被操作数: 临时)

self.屏幕.text = 结果

let 临 = 结果 as NSString

缓存2 = 临.doubleValue

} else {

缓存2 = 缓存

缓存 = 0.0

}

}

# 5 关键技术

5.1枚举

Swift的枚举是一种完全重新定义的类型。Swift的枚举具有很多面向对象的特征，Swift的枚举支持计算属性，支持实例方法和类型方法，支持定义构造器完成初始化，Swift允许通过扩展来增强枚举的功能。

下面程序定义了一个算法枚举

enum 算法 {

case 加法

case 减法

case 乘法

case 除法

case 平方

case 指数幂

case 未选择

}

Swift的类和结构体高度相似，无论是从定义语法还是用法上，Swift的类和结构体都具有高度的相似性。主要区别在于：（1）结构体不支持继承（2）结构体不支持定义析构器（3）结构体是值类型，而类是引用类型 结构体和类的语法格式非常相似，定义类的格式如下， [修饰符] class 类名 { 零个到多个构造器 零个到多个属性 零个到多个方法 零个到多个下标 } 例如，定义一个类，class ViewController: UIViewController {

var Calculator:计算 = 计算()

var 显示缓存:String = ""

var 缓存 = 0.0

var 缓存2:Double?

}

下面定义一个计算结构体，

struct 计算 {

private var 当前算法:算法 = .未选择

mutating func 设置当前算法(临时:算法) {

当前算法 = 临时

}

func 读取当前算法() -> 算法! {

return 当前算法

}

构造器用于完成实例的构造过程，这个过程包括为实例中的每个存储属性设 置初始值和执行必要的准备和初始化任务。Swift的构造器无须显式声明返回值类型，也无须显式使用return返回实例，swift构造器的本质就是一个或多个名为init的函数。Swift为程序员提供了两个构造器：无参数的构造器和初始化所有实例存储属性的构造器。如果类中存在需要在构造器中指定初始值的实例存储属性，则该类必须由程序员定义构造器。 创建实例的根本途径是构造器，通过“类型名（参数）”的形式即可调用该 类或结构体构造器。

# 6 结束语

通过对移动平台开发的学习，基本了解了swift语言的编程思想，了解了swift语言强大的可移植性，随着时代的发展，我意识到知识的重要性。同时也让我深刻认识到了科学的严谨性，还有自己身上的很多不足之处，今后我会更加严格要求自己，以便提高自己的实操能力。本次的毕业设计将会成为我以后学习和工作的一笔很大的财富。整个学习并实践的过程都是宝贵的，在这里我很感谢学校和老师能够给我这次机会，让我将很多理论知识付诸实践，得到印证。

课题在开发的过程中得到了朝力萌老师的精心指导，以及同组学员的互相帮忙在此深表感谢。

Mobile platform application development - calculator

Computer and information engineering college Yu Dandan 20131105783

Directed by Chao Limeng Professor

**Abstract**  IOS is the most advanced mobile platform in the world, it re defines the function that can be realized only by pocket sized mobile devices. IOS integration with Xcode integration development environment tools to create excellent applications become very easy to create SDK. Because Apple's Store App exists on every iOS operating system device, and is localized around the world, so there is no other platform for mobile developers more attractive than it. Swift is a new programming language for the preparation of iOS and X OS applications. Swift uses a safe programming mode and adds a lot of new features, which makes programming easier, more flexible, and more fun. Swift is based on a mature and much loved Cocoa and Touch Cocoa framework, his advent will redefine the software development.

**Keywords**  Xcode; IOS integrated development environment; swift;