Mobile Application Programming: iOS

Introduction
My Interests

- Computer Graphics
- CTO Pixio Software
- Wrote MobileFinder for the iOS 1.4 jailbroken SDK
- Lived in Brazil and speak Brazilian Portuguese
- Send balloons to 100,000 feet and take pictures
- Read US revolutionary biographies, fantasy, and sci-fi
- Current obsessions: NASA, Space, Warhammer 40k
Course Website

http://www.eng.utah.edu/~cs4962
Cheating in the context of this course is generally, but not limited to, sharing and copying of code from other students or the Internet. Any code making up your solution should be written and understood by you. Small quantities of template code will at times be provided by the instructor. You can use this code in submissions but should still be able to fully explain the function of all template code you use. Refer to but do not copy code from the examples given in class.
Why Not Android?

- Small percentage of Android device owners actually purchase software from the various markets.
- **Fragmentation** in the hardware and software make development for many devices more difficult than iOS.
- Companies overwhelmingly prefer to target iOS platforms before they target Android.
- Hardware running Android **arguably inferior** to that of other companies, mostly due to price of units.
Xcode

- Integrated Development Environment
- Editor with advanced features
- Documentation (Option-click)
- Debugger
- Compiler (llvm)
- Profiler (Instruments)
- Graphical Interface Builder (Interface Builder)
iPhone Project Anatomy

```swift
import UIKit

@UIApplicationMain
class AppDelegate: UIResponder, UIApplicationDelegate {
    var window: UIWindow?

    func application(_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplicationLaunchOptionsKey: Any]?) -> Bool {
        window = UIWindow(frame: UIScreen.mainScreen().bounds)
        window?.rootViewController = UINavigationController()
        window?.makeKeyAndVisible()
        return true
    }

    func applicationWillResignActive(_ application: UIApplication) {
        // Sent when the application is about to move from active to inactive state. This
        // can occur for certain types of temporary interruptions (such as an incoming phone
        // call or SMS message) or when the user quits the application and it begins the
        // transition to the background state.
        // Use this method to perform actions that the application will later resume from.
    }

    func applicationDidEnterBackground(_ application: UIApplication) {
        // Use this method to save user data, invalidate timers, and store enough application state information to restore your application to its
        // current state in case it is terminated later.
        // If your application supports background execution, this method is called instead
        // of applicationWillTerminate when the user quits.
    }

    func applicationWillEnterForeground(_ application: UIApplication) {
        // Called as part of the transition from the background to the active state; here
        // you can undo many of the changes made on entering the background.
    }

    func applicationDidBecomeActive(_ application: UIApplication) {
        // Restart any tasks that were paused (or not yet started) while the application was
        // inactive. If the application was previously in the background, optionally refresh
        // the user interface.
    }
}```