**CSCI3310 Mobile Computing & Application Development**

Assignment 2 (iOS)

Programmable Calculator

Due : Nov 4, 2015 11:55pm

This assignment consists of two parts. In the first part, you should complete the calculator example of iOS walkthrough in the Stanford CS193 of Developing iOS 8 Apps with Swift. Then you will need to code some extensions to the calculator so as to make it possible to run a program.

**Part 1 (40%)**

Go to iTune U and find the course “Developing iOS 8 Apps with Swift”. It is a wonderful iOS programming course offered by Paul Hegarty from Stanford. You may choose to subscribe it so as to get more information on iOS programming. You may also go there through this link:  
<http://web.stanford.edu/class/cs193p/cgi-bin/drupal/>

The video to watch:

1. Logistic, iOS overview

To save your time, you may directly start from Lecture 1 video at 12:00. That is the starting point for introduction on using Xcode, with very detail explanations. After watching to finish, this should set up the basic display on the storyboard in Xcode.

2. More Xcode and Swift, MVC

From start to 57:00. This will set up primitive calculator functions without model.

3. Applying MVC

The whole video.

Through these , you should be able to produce a stack calculator on iOS device that can perform addition, subtraction, multiplication and division.

Note : Our department (CSE) Mac computers are installed with mac OS 10.9.5 (Maverick) with Xcode 6.2, which is exactly the same configuration with that in the video. So basically you should not have too much problem with following the video. However if your machine is more recent i.e. Yosemite with Xcode 6.5 or above, then the steps may be a little bit different as Swift also changed to 2.0.

**Submission**

Please zip/rar the calculator folder after you stepped through all the steps in the iOS walkthrough video and submit it in a single archive to the part 1 submission slot.

**Part 2 (60%)**

At this point, you should have completed the calculator in the video series Stanford CS193p above of developing the calculator as your first step in working with Xcode. Now we will extend the calculator code to do the following:

1. extend the CalculatorBrain to allow inputting 3 variables into the calculator’s program and to show the user the steps they have entered to get the result showing in the display (i.e. showing the CalculatorBrain’s “program”)
2. Add a “About” button and upon pressing, will show an about screen which show information about the developer i.e. yourself. You may arbitrarily add any images/icons you like in the About screen.

The above features will require you to use id type and introspection, as well as building another UIViewController, creating a UINavigationController, and finally creating a segue in a storyboard that we have covered so far.

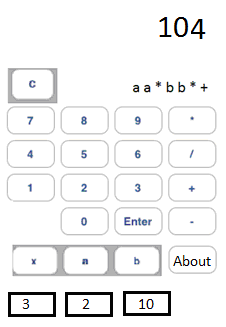
**Requirements**

Before you proceed to do this part, you should first make a copy of your results in Part 1 first. You should be able to run the part 1 calculator after this step without any problem. Then proceed to the steps below.

1. You need to extend the functionality of the calculator by implementing the following change:

Add a new text label (UILabel) called *inputFormula* to your user-interface which shows everything that has been sent to the brain (separated by spaces). For example, if the user has entered “6 Enter 5 + 2 \*”, this new text label would show 6 5 + 2 \*. Don’t forget to have the C button (in pt. 2) clear this too. All of the code for this task should be in your Controller (no changes to your Model are required for this one). You do not have to display an unlimited number of operations and operands, just a reasonable amount.

1. Add a “C” button that clears everything (your display, the new UILabel you added above, etc.). The Calculator should be in the same state as it is at application startup after you touch this new button.
2. You also need to modify your calculator so that it now can accept variables as operands. A variable will be specified as an *NSString* object. You will also need to add the capability of your CalculatorBrain to allow the pushing of variables onto its internal stack. Also the evaluate() function should also now be able to use a variable’s value. The values of the variables will only be supplied when the “program” is “run.” The values of the variables will only be supplied when the “program” is “run.” You must add a *runProgram* method using the stored variable values. Add three text entry boxes and three buttons “x”, ‘a”, and ‘b” respectively. The user will click on the text entry box to input values for the variables. When a new value is being input to the calculator, the runProgram should be executed to show the new result based on current input formula. The result show be blank in case of any invalid or empty input. The layout may like this:



After you have finished your tasks above, do the following:

1. When your application is run on the iOS device, it must present the user-interface of your calculator from above inside a UINavigationController.

2. Design an “about screen” which shows whatever information about the calculator e.g. your favorite photo, label telling the development date etc.

Use Segue to connect the two screens together so that user can freely navigate between these two screens.

**Submission**

You should packed all your program and related files e.g. icon file, settings etc. into a folder named 3310\_asg2b, and zip the folder into the same named zip or rar file, and submitted it into our assignment collection slot in Blackboard system before the deadline, Nov 4, 2015 11:55pm.

Late submissions will risk a mark deduction from 5% to 30% if they are being done within 24 hours after the deadline. Submission later than Nov 5 11:55pm won’t be considered.