CS 646 iPad/iPhone Application Development
Fall Semester, 2015
Assignment 1

San Diego State University -- This page last updated 8/25/15

Assignment 1 (100 points)
Due Sept 5 11:59 pm
Objectives

1. Get started using Xcode.
2. Learn basic Swift

Programing Problems (10 points each)

1. Write a Swift function, call it quadTable, that has one argument, an integer. Lets call it N. The function prints out on the console the values k and $k^2 + 3*k - 1$ for the values k = 1, 2, ..., N. Print each value of k on a separate line. So the output of quadTable(3) is given below. (Where does your the output go?)

```
k=1 k*k + 3k - 1 = 3
k=2 k*k + 3k - 1 = 9
k=3 k*k + 3k - 1 = 17
```

2. Write a Swift function, call it polyTable, that has one argument, an Int (N) and returns a array. The function returns an array of size N. The k'th element of the array contains the value $k^3 + 2*k + 4$ for k = 0, 1, ..., N. So polyTable(3) would return [7, 16, 37].

3. Write a Swift function, call it busyStudents, that has one argument an array of sets of names and returns the intersection of all the sets in the array. In the code below busyStudents would return {"Peter"}.

```
let courseA: Set = ["Peter", "Paul", "Mary"]
let courseB: Set = ["Peter", "Paul", "Dylan"]
let courseC: Set = ["Tom", "Peter"]
busyStudents([courseA, courseB, courseC])
```

4. Write a Swift function, call it average, that has one argument an array of Ints and returns an optional double, which is the average of the inputs ints. If the input array is empty return the optional value nil.

5. Write a Swift function average2 which is the same as average in #4 except that the input is an array of optional ints.

6. Write a Swift function cost that has one argument a dictionary. The dictionary has three keys: "name", "price", and "quantity". The function cost returns the cost of the item, that is the price * quantity. The keys and values in the dictionary are all strings. The value at "price" is the string of a double like "3.45". The string at "quantity" is the string of an integer. Note that if either the key "price" or "quantity" is not in the map the function "cost" returns 0 (zero). Examples given below.

```
let iceCreamA = ["name":"Mochie Green Tea", "quantity": "2", "price": "2.3"]
let iceCreamB = ["name":"Mochie Green Tea", "price": "2.3"]

cost(iceCreamA)        // returns 4.6
cost(iceCreamB)        // returns 0
```

7. Write a Swift function wordCount that has two arguments, a string and an Int. The string contains words separated by a space. For example "cat bat cat rat mouse bat". wordCount returns a dictionary where the keys are the words in the string and the values are the number of times the word appears in the list. Only the words that occur at least as many times as the second argument are in the dictionary.

   wordCount(words: "cat bat cat rat mouse bat", count: 1) returns ["cat": 2, "bat": 2 "rat": 1, "mouse": 1]
   wordCount(words: "cat bat cat rat mouse bat", count: 2) returns ["cat": 2, "bat": 2]
   wordCount(words: "cat bat cat rat mouse bat", count: 3) returns [:]

8. Write a Swift function wordCount2 that has the same arguments as wordCount in problem 7 and returns the same result. However give the second argument a default value of 2 so we can call the function with one or two arguments as shown below.

   wordCount2(words: "cat bat cat rat mouse bat") returns ["cat": 2, "bat": 2]
   wordCount2(words: "cat bat cat rat mouse bat", count: 3) returns [:]

9. Write a Swift function wordCount3 that has one argument an Int, which has the same role as the second argument of wordCount. wordCount3 returns a function. The return function has one argument a String that contains words. When evaluated the returned function returns the dictionary of words in the string with the number of times the word appears in the list. But as in problem 7 it only contains the words that occur as many times as the argument to wordCount3. See examples below.

   let testA = wordCount3(2)
   testA(words: "cat bat cat rat mouse bat") returns ["cat": 2, "bat": 2]
   testA(words: "a a a b c c") returns ["a": 3, "c": 2]
   let testB = wordCount3(3)
   testB(words: "a a a b c c") returns ["a": 3]

10. The problem with polyTable in problem two is that if we what to change the equation (k^3 + 2*k + 4) we need to edit and recompile polyTable. Write a Swift function evaluate that has two arguments. The first argument is an Int as in problem two. The second argument of evaluate is itself a function that has an Int as an argument and returns a double. Your function evaluate then returns an array of doubles. The k'th element of the returned array is the result of evaluating the second argument with the value k for k = 0, 1, ..., N.

11. (Extra Credit 5 points) In problem 6 one might prefer to use a dictionary with values with different types. For example ["name":"Mochie Green Tea", "quantity": 2, "price": 2.3]. That is the value at "quantity" being an integer and the value at "price" being a double. Swift's type system makes this a bit tricky. Write a Swift function cost2 that performs the same as problem 6's cost but accepts the dictionary given above.

What to Turn in

Create a Xcode Playground for your assignment 1. Answer all questions in the single playground. Use a comment to separate each questions. Zip up the playground and turn in your zipped file using assignment 1 link on blackboard.

Late Penalty

An assignment turned in 1-7 days late, will lose 3% of the total value of the assignment per day late. The eight day late the penalty will be 40% of the assignment, the ninth day late the penalty will be 60%, after the ninth day late the penalty will be 90%. Once a solution to an assignment has been posted or discussed in class, the assignment will no longer be accepted. Late penalties are always rounded up to the next integer value.