

## **Codestars - Problem Set 0**

Please submit your completed problem set as a Word document and copy a Dropbox link to this document for our review. The deadline to submit your answers to this problem set is **July 12th, 2015 at 11:59 PM**.

All questions presented below will be based on the readings and videos lectures assigned below. You are to watch the assigned videos and complete the assigned readings in order to answer the questions presented in the quiz. We recommend that you take detailed notes as you complete the video lectures and readings as this will help you reinforce your learning.

### **Pre-work Part I: C, Programming and Computer Science**

Watch the video lectures for weeks 0 - 5 of Harvard's CS50 course. While watching these lectures you are to also read chapters 1 - 12 of Objective-C Programming, The Big Nerd Ranch Guide, as it covers a lot of the same material covered in the CS50 lectures. During this phase of the pre-work you will learn about programming, the C programming language and Computer Science basics that will help you develop a solid foundation that will help you become a strong developer.

### **Pre-work Part II: Objective-C**

Read chapters 13 - 30 of Objective-C Programming, The Big Nerd Ranch Guide. During this portion of the pre-work you gain a solid understanding of the Objective-C programming language, which is used to build both iOS and OS X applications.

### **An important note on how to work through this material:**

You are advised to begin working on this curriculum as soon as possible and not to wait until the last minute to work through the work. During the actual course you will be expected to spend a minimum of two hours a day working on assignments outside of class so you should use this opportunity to get into this habit.

### **Links to Pre-work Material**

Harvard CS50: <https://cs50.harvard.edu/lectures>

Objective-C Programming, The Big Nerd Ranch Guide:  
<https://www.bignerdranch.com/documents/objective-c-prereading-assignment.pdf>

## **Basic Information**

Full Name:

Email address:

## **Programming Basics**

To the best of your ability, describe the difference between programming and computer science.

What is binary?

What are bits?

What are bytes?

What is ASCII?

What are algorithms? Using pseudo code, write an algorithm that will reverse a string.

What does it mean to compile your code? Describe how a compiler works.

What is the new line character? How does it work?

What is the terminal or command line? How can we use it as developers?

## **The C Programming Language**

What are header files and what are they used for? Write an example of how you would use a header file below.

List the different types used in C along with their size.

What's the difference between int, float and double?

What are variables? How might we use them in our program? How do we declare variables in C?

How do we create strings in C?

What is a char\*?

Write a simple if/else statement in C.

Explain what functions are and they help us write better code?

Describe how strings are represented in C.

What is the null terminator character?

What is an array? How are they represented in C? How are strings and arrays related in C?

What does void represent?

What is argv? How does it work?

What is a struct and why are they useful? Write a struct that represents an employee where an employ has a name, department, title and number of years on the job.

What is a memory address?

What are pointers? How do we use them when programming in C?

What is memory?

How do we ask the operating system for memory using C?

What is a segmentation fault?

Describe what a memory leak is. How can we avoid them?

What does it mean to pass-by-reference?

What is the difference between the stack and the heap? How does C use these data structures to perform memory management? You should explain, step-by-step, the process and key functions used in C to accomplish this.

## **Objective-C**

What are classes, what are objects? How are the two related?

What are methods?

What are the difference between class methods and instance methods?

What does it mean to send a message in Objective-C? Give an example of how this works.

What are alloc and init? What are they used for? What C functions do they remind you of?

How does memory management work in Objective-C? How does it differ and/or resemble memory management in C?

What is NSString? How is it different from a regular C string?

Name some ways that we can create arrays in Objective-C? How is this different from how we create arrays in C?

Explain what properties are.

What are instance variables?

Explain what self is.

How does inheritance work in Objective-C?

How do we override methods?

Explain the “super” keyword.

Explain the inheritance hierarchy.

What are strong references?

What are weak references?

What are some ways that we can prevent memory leaks?

What are immutable objects in Objective-C? Name some immutable objects.

What is the difference between an NSSet and an NSMutableSet?

What is the difference between NSDictionary and NSMutable Dictionary?

What are #import and #include used for?

What are global variables?

Explain what a callback is. How might callbacks be useful?

Explain what a block is, what would we use one for?

What are protocols? What do we use them for?

### **Problem Solving**

1) Write a function that cubes the sum of three numbers and returns the result. Please complete this problem using C.

2) If we list all the natural numbers below 10 that are multiples of 3 or 5, we get 3, 5, 6 and 9. The sum of these multiples is 23. Find the sum of all the multiples of 3 or 5 below 1000. Please complete this problem using C.

3) Write a short program that reverses a string using Objective-C.

4) In Objective-C write a simple class that represents a student. A student should have a first name, a last name, a major and a year (freshman, senior, etc.). Your student class should have two methods. One that allows the student to state his or her name and another that will allow the student to greet a person given a name.