**COS 120 - Intro to Computational Problem Solving  
Lab 13 - Recursion Review and OOP**

**L13-1.** Write a recursive function to implement integer division as a series of subtractions. Utilize print statements to demonstrate how the recursion actually works. Assume positive dividends and divisors.

**L13-2.** Write a recursive function to implement modulus as a series of subtractions. Utilize print statements to demonstrate how the recursion actually works. Assume positive terms.

**L13-3.** Implement a class called Time that allows the creation of objects containing attributes of hours, minutes and seconds in the range of 0-23, 0-59 and 0-59 respectively. If illegal values are passed to the constructor for any of the attributes (instance variables), assign a value of zero to the attribute.

**L13-4.** Write an accessor method getTime that returns the time in hh:mm:ss format.

**L13-5.** Write an accessor method getCivilianTime that returns the time as hh:mm:ss AM or PM.

**L13-6.** Write individual accessor methods for hours, minutes and seconds.

**L13-7.** Write individual mutator methods that allow the user to change hours, minute or seconds, but only to valid values. (Mutator methods often start with the word set).

**L13-8.** Write a mutator method incrementTime that increments the time by one second. Be sure to handle all possible outcomes of such an increment.

**L13-9.** Demonstrate that the current version of the time class is not enforcing information hiding (as simple as a direct modification of one of the attribute values). Now, m**odify the class so that information hiding is enforced.** Demonstrate this using the same code you used to demonstrate lack of information hiding.

**L13-10.** Write a timeDiff method that given another time object as a parameter, returns the difference between the two times as a list containing hours, minutes and seconds. The difference should always be positive.

**L13-11.** Write test code that exercises all of the methods, and demonstrates they work correctly for all unique classes of test values.