CSC 600 – Assignment 3 – Functional Programming

Due: Sunday, October 22nd. 2017 @ midnight

Due: Tuesday, October 24th. 2017 @ midnight

Description

For the following assignment, you will get practice writing code that adheres to the function programming rules. In Essence, functional programming is a way of writing software with pure functions and immutable values.

When we say pure we mean we do not break the following rules:

1. The function always evaluates to the same result value given the same argument value(s). It cannot depend on any hidden state or value, and it cannot depend on any I/O.
2. Evaluation of the result does not cause any semantically observable side effect or output, such as mutation of mutable objects or output to I/O devices.

Then for the assignment in general, you need assume that values in your program are immutable. This will greatly affect how you build collections and computer certain values.

Submission

Please submit all code to your given repository by the deadline. Please have your answer to each question marked in comments so I know which one to grade. Simply put #1.A and #1.B and so on.

Work to be Done:

1) In Ruby, methods are not what is called first-class citizens. However, procs (lambdas as well) and block logic in Ruby are. For each of the following descriptions below give an example of a proc in Ruby that satisfies the condition. There should be an example for EACH case.
   a) The first class object may be expressed as an anonymous literal value (constant). Show an example of the anonymous function and its use.
   b) The first class object may be stored in variables (i.e. it may have a symbolic name). Show examples of defining and using named functions.
   c) The first class object may be stored in data structures. Show an example of a data structure(e.g. a list) that contains functions.
   d) The first class object may be comparable to other objects for equality. Show an example of comparing functions and lists for equality.
   e) The first class object may be passed as parameter to procedures/functions. Show an example of passing function as an argument to another function.
   f) The first class object may be returned as result from procedures/functions. Show an example of returning a function as a result of another function.

2) Write Ruby code that computes the sum of the first 20 natural numbers whose square value is divisible by 2. This code must adhere to the functional Programming Rules.
3) Write Ruby code that computes all the leaps years between the years 1800 and 2100 *inclusive*. This code must adhere to the functional Programming Rules.
4) Write ruby code that takes a string and collects are the words that contain the vowels a and e. This code must adhere to the functional Programming Rules. Please create your own large string for testing, Leave the string in your code when done.