**Haskell Assignment 2**

What do the following functions do?

1. **mystery0 [] = 0**

**mystery0 (a:x) = 1 + mystery0 x**

The function mystery0 is a function that returns the length of a list, string, and list of lists. An empty list will return list 0, while a list with 7 elements inside the list will return 7. When given a string, the function will return the length or number of characters including spaces within the string. When given a list of lists, the function will return the number of lists contained within the encapsulating/main list. For the record, each of the lists within the main/encapsulating list can be empty and of different sizes and it will still be counted as an element and will uptick the length by 1.

1. **mystery1 xs = [y | x <- xs, y <- [x,x] ]**

The function mystery1 takes a list, string, and a list of lists and duplicates each element. When given a list, mystery1 will duplicate each element within the list in succession and return the resulting list. So when given [1,2,3], mystery1 will return [1,1,2,2,3,3]. An empty list will result in an empty list since there is nothing to duplicate. When given a string, mystery1 will duplicate each char within the string in succession and return the resulting string. So when given “hello”, mystery1 will return “hheelllloo”. When given a list of lists, the function will duplicate each list in succession as a whole and return the resulting list of lists. When given [[1,2], [1,2,4], []], mystery1 will return [[1,2], [1,2], [1,2,4], [1,2,4], [], []].

1. **mystery2 f [] = []**

**mystery2 f (a:x) = f a : mystery2 f x**

The function mystery2 takes in a function and list, string, or list of lists then applies the given function on the given list, string, or list of lists to each element, char, or list contained within the list or string and then returns the new and resulting list or string. If mystery2 were to apply to an empty list, then mystery2 will result in an empty list.

1. **mystery3 item [] = []**

**mystery3 item (x:y) | item==x = mystery3 item y**

**| otherwise = x : mystery3 item y**

The function mystery3 takes a list or string and removes all elements that `item` is equal to that is found from within the list or string and then returns the resulting list or string after the removal of all elements equal to `item`. If mystery3 were to apply to an empty list, then mystery3 will result in an empty list.

1. **mysteryd = (\*) 2**

**mysteryr f x = f (f x)**

**mystery4 = mysteryr mysteryd**

The function mystery4 takes in only an integer and multiplies the given integer by 2 twice due to the f(f x) defined by mysteryr which is a function within a function. When given mystery4 1, the function will take the 1 and multiply it by 2 then use that result to repeat the function one more time by multiplying the result by 2 again thus giving us the final result and returning 4. This function doesn’t work on lists, strings, or even floating points or any number that isn’t a whole number. mysteryd is the other function that determines why we are multiplying our inputted integer by 2.

Find a version of append on the web or in the book (or write your own) and test it

1. > **append [7, 8] [2, 3, 4]**

**[7, 8, 2, 3, 4]**

1. You can simply use the ++ operator to append or combine both lists into one.

The expression [7,8] ++ [2,3,4] will give you the same resulting list as append [7,8] [2,3,4]. This can be found in book.

1. Another version of append that can be utilized and was found on the web was

append [] [] = []

append xs [] = xs

append [] xs = xs

append a@(h:first) b@(c:second) = h:merge first b

The function append will take two set of lists and will result in a list that combines all the elements in the first list with the second list by placing all the following elements in the second list right after the last element in the first list. Then there are three rules/expressions within the append function that states if both are empty lists than append will return an empty list. While the other two rules/expression state if either one of the lists given is empty while the other list isn’t an empty list, return the list that isn’t empty.