Module 4
Swift Development

Swift 5 Collections:

Problem1

- Implement a function which receives a list and returns True if the list is already sorted ascendingly and returns 0 if the list is not sorted ascendingly.

Problem2

- Write a function which receives a list and returns True if the list is "Partially sorted" and returns False if the list is not "Partially Sorted". A list is "Partially sorted" if and only if there exists an item in the list which if removed, the list will become a sorted list. For instance the following list is "Partially sorted":
- [1,2,5,10,6,8,9] This is partially sorted because it is not originally sorted but if we remove 10 from the list, then the list is sorted.

Problem3

- Write a function which takes no input parameter and allow the user to enter words as many as the user wants until the user enters an empty word. When the user enters a word, the function will add the word to a list which was originally empty. Before the function adds the word to the list, it should check whether the same word had been already added to the list or not. If not, then the word is added to the list and if yes, the word should not be added to the list. The function will eventually return the list of words.
    - *Note: Only use List to solve this problem*

Problem4

- Write a Swift function which receives 2 mathematical equations and adds these two equations and prints and retunes the result. The mathematical equations that are inputs of this function have the following format / specifications:
    - The maximum degree of the equation is 10.
    - The equation can have only 2 variables: X and Y.
    - The syntax of the equation is like: Example: X^4+ 5X^2 + Y^3+Y^2+1

- Only use lists to solve this problem

Problem5

- Write a Swift function which receives 2 lists as its input parameters and checks whether one of the list is included in another input list. A list A is considered to be included in List B, if all elements in A are elements in B.

Problem6

- Write a function which has no input parameter. The function asks the user to enter numbers. The user can enter repeated numbers but if the user entered a number which was already entered, the function will provide an error message to the user and ask the user to enter another one. The user can enter numbers as long as s/he has not entered 0. Once a 0 is entered, the function returns the sum of all distinct numbers the user had entered.

Problem7

- Write a function which lets the user enter English words. The user can keep entering English words as long as the user has not entered the word “exit”. Once the user enters “exit” the function will return and print the list of all distinct words starts with English alphabets. Like:
  A: Ali, apple, …
  B: Bob, book
  … until Z

Problem8

- Design and implement a function which has one input parameter which is a number which is greater than 50, called num. Then the function will create a dictionary whose keys are 2 and 3 and 4 and 5 and 6 and 7 and 8 and 9. Then the function calculates the values for each of the above keys. The value for a key is all the numbers between 2 and input “num” that are divisible by the key. The function eventually will print the result.
- Hint: Use a dictionary whose values are lists.
- Example:
  num = 20
  2: [2,4,6,8,10,12,14,16,18,20]
  3: [3,6,9,12,15,18]
  4: [4,8,12,16,20]
  5: [5,10,15,20]
  6: [6,12,18]

7: [7, 14]
8: [8, 16]
9: [9, 18]

Problem9

- Write a function which receives a list and returns a number. In the list, all numbers have been repeated twice except one number that is repeated once. The function should return the number that is repeated once and return it.

Problem10

- Write a Swift function which receives 3 lists as its input parameters and combines the lists and remove repeated numbers from the combined list and return the combined list. For instance, if the input is [1,2,3,4,2,3] and [3,4,6,7] and [-1,0,23,4] the result is [1,2,3,4,6,7,-1,0,23]
- Note, the order the lists are combined together does not matter.