CISC106 – Lab 08

This lab is due 08/03/2014 11:55 pm (Submission via Sakai)

- Please do all the assignment in one file named lab08.py. This is an individual assignment, so please do all the work accordingly.
- Use comments to separate your programs for each question.
- Use design recipe to write functions and test cases. You need to provide at least 3 test cases for each function you implement.
- The problems are worth 90 points + 10 points for attending the lab session.

Problem 1: (10 points)
Write a function named ***addToPhoneBook*** that gets three parameters (name : String, phoneNumber : String, phonebook : Dictionary) and adds the name-phoneNumber pair to phonebook dictionary. If insertion operation is successful function returns True, otherwise it returns False. For phonebook dictionary, name is the key and phoneNumber is the value.

Problem 2: (10 points)
Write a function named ***deleteFromPhoneBook*** that gets two parameters (name: String, phonebook : Dictionary) and deletes the key-value pair that's key is name. If deletion operation is successful function returns True, else it returns False.

Problem 3: (10 points)
Write a function named ***updatePhoneBook*** that gets three parameters (name: String, phoneNumber: String, phonebook : Dictionary) and updates the name-phoneNumber pair in phonebook dictionary. If update operation is successful function returns True, else it returns False.

Problem 4: (10 points)
Write a function named ***getPhoneNumber*** that gets two parameters (name: String, phonebook: Dictionary) and returns the phone number if name exists in phonebook. If name does not exist in phonebook, it returns empty string.

Problem 5: **(10 points)**
Write a function named ***mergePhoneBooks*** that gets two parameters (phonebook1: Dictionary, phonebook2: Dictionary) and returns a merged phonebook. If same names exist in both phonebooks and their phone numbers are different, your function should use the phone number in phonebook1.

Problem 6: (10 points)
Write a function named ***createUserNumSet*** that does not get any parameter but asks to user to enter 10 numbers. If user enters any number that is already entered,

function prints “You already entered that number” and asks to user to enter a new number. After getting 10 unique numbers function returns a **Set** that consists of numbers that user entered. You do not need to write test cases for this function.

Problem 7: (10 points)
Write a function named ***deleteFromNumSet*** that gets a parameter (numsSet : Set) , prints maximum and minimum numbers in the set and asks to user to enter numbers between maximum and minimum numbers. If the user enters a number that is in the numsSet, the function prints “The number is in the Set” and deletes the number from numsSet. If user enters a number that is not in the numsSet, the function prints “The number is not in the Set”. The function keeps asking to user to enter numbers until numsSet is empty.

Problem 8: (20 points)
Write a class named **Car** that has following attributes; color (string), make (string), model (string), year (integer), mpg (float), fuel_capacity (float), fuel_gauge (float, values can be between 0-1) and the following methods
get_gas_consumption: Gets a parameter (distance : float) and returns amount of gas that car consumes.
is_safe_trip: Gets a parameter (distance : float) and returns True if the car can complete the trip with the existing gas in the fuel tank. It returns False otherwise.
get_gas_amount_in_fuel_tank: Function does not get any parameter but returns the amount of gas in its fuel tank.