Quiz 1 var 2 - (5 points) – 40min

#1 (String) – 1.5 points

Hangman: Let’s say the word the player has to guess is “EVAPORATE”. For this exercise, write the logic that asks a player to guess a letter and displays letters in the clue word that were guessed correctly. For now, let the player guess an infinite number of times until they get the entire word. As a bonus, keep track of the letters the player guessed and display a different message if the player tries to guess that letter again. Remember to stop the game when all the letters have been guessed correctly! Don’t worry about choosing a word randomly or keeping track of the number of guesses the player has remaining - we will deal with those in a future exercise.

An example interaction can look like this:

```
>>> Welcome to Hangman!
_ _ _ _ _ _ _ _ _
>>> Guess your letter: S
Incorrect!
>>> Guess your letter: E
E _ _ _ _ _ _ _ E
...
```

And so on, until the player gets the word.

#2 (math) –1 point

suppose we have an array 50 98 17 79 which after sorting becomes 17 50 79 98. Now:

17 was at 3-rd place initially

50 was at 1-st place initially

79 was at 4-th place initially

98 was at 2-nd place initially

so result is 3 1 4 2

Initial data will contain array size at first line and array values itself in the second (integers separated by spaces).

Answer should contain initial indexes of the array members after they are reordered by sorting.

Example:

input data:

4

50 98 17 79

answer: 3 1 4 2

#3(Dictionary) - 1 points

Dictionaries provide a convenient way to store structured data.
Here is an example dictionary:
d=[{'name':'Todd', 'phone':'555-1414', 'email':'todd@mail.net'}, {'name':'Helga', 'phone':'555-1618', 'email':'helga@mail.net'}, {'name':'Princess', 'phone':'555-3141', 'email':''}, {'name':'LJ', 'phone':'555-2718', 'email':'lj@mail.net'}]
Write a program that reads through any dictionary like this and prints the following:
(a) All the users whose phone number ends in an 8
(b) All the users that don't have an email address listed

#4 (Input/Output from a file) – 0.75 points

You are given a file called class_scores.txt, where each line of the file contains a one word username and a test score separated by spaces, like below:
Anuar 83
Erasyl 86
Write code that scans through the file, adds 5 points to each test score, and outputs the usernames and new test scores to a new file, scores2.txt.

#5 (List) – 0.5 points
Create a Python set such that it shows the element from both lists in a pair. Consider the case when the lists have different length.
Example:
first_list = [2, 3, 4, 5, 6, 7, 8]
second_list = [4, 9, 16, 25, 36, 49, 64]
Result is {(6, 36), (8, 64), (4, 16), (5, 25), (3, 9), (7, 49), (2, 4)}

#6 (List + Dict) – 0.25 points
Iterate a given list and check if a given element exists as a key's value in a dictionary. If not, delete it from the list

Example:
roll_number = [47, 64, 69, 37, 76, 83, 95, 97]
sample_dict = {'Jhon':47, 'Emma':69, 'Kelly':76, 'Jason':97}
After removing unwanted elements from list [47, 69, 76, 97]