Practice exercises: Week 7 (Lists and Tuples)

1. Write a program to create a list called
   Marks = [45, 89, 0, 78, 67, 12, 90, 38, 54, 89]
   Print the average of it. The expected output will be.

```
average is: 56.2
```

2. Write code that prompts the user ask any float value as input continuously and append those values into a created empty List until user press enter key with no input (terminating user input loop). Then display the values of Lists in decreasing order.

3. Create a tuple **"Months"** with names of all 12-month names of the year as its elements. Then prompts the user to enter *month in number* as input and print its corresponding name obtained from aforenoted tuple. The sample run is here:

```
Enter the month number:3
March
>>> %Run p3_w7.py
Enter the month number:12
December
>>> %Run p3_w7.py
Enter the month number:1
January
```

4. Write a program that prompts the user to enter gender details (M or F) of Group A that contains 10 members. The group's gender details must be stored in a List. Then print the total number of Male and Female candidates of the group.

```
Shell
Python 3.7.9 (bundled)
>>> %Run p1_W2.py
  Enter the gender:M
  Enter the gender:M
  Enter the gender:F
  Enter the gender:M
  Enter the gender:F
  Enter the gender:F
  Enter the gender:F
  Enter the gender:M
  Enter the gender:M
  Enter the gender:M
  Number of male candidates: 6
  Number of female candidates: 4
```

5. Write code that prompts the user to enter any English word (upper case or lowercase or mixed) as input and append it to a list in its lower-case form until the user enter the word **"quit"** *(upper /lower /mixed form)* to terminate the loop. You should use while-loop for continuous input process. In particular, if the user enters the word again or already exists in the List then it should not be appended. That is, List should not contain duplicate elements [Hint: not in operator]. Print all elements stored in the list when input loop gets terminated. The sample run is here:

```
>>> %Run p4_w7.py
  Enter a word to add in the list:quit
  []
>>> %Run p4_w7.py
  Enter a word to add in the list:QUit
  []
>>> %Run p4_w7.py
  Enter a word to add in the list:abC
  Enter a word to add in the list:abc
  Enter a word to add in the list:Apple
  Enter a word to add in the list:banana
  Enter a word to add in the list:APPle
  Enter a word to add in the list:Plums
  Enter a word to add in the list:orange
  Enter a word to add in the list:GRAPES
  Enter a word to add in the list:QUIT
  ['abc', 'apple', 'banana', 'plums', 'orange', 'grapes']
>>>
```

6. Two tuples created namely **tup_ Odd = (1, 3, 5, 7, 9)** and **tup_ Even = (2, 4, 6, 8, 10).** Write a program that contains a function **"combineTuples()",** which takes two tuples as parameters and combine those passed two tuples to form the new tuple and it should be sorted in ascending order. Then function should return the new tuple (sorted) to main program where it was called. The new tuple in the main program namely **tup_Odd_Even** should contain all elements of aforenoted tuples in increasing order format. Then print the values of **tup_Odd_Even**. [Hint. *You may need List as well*]. The given input and expected output will be:

```
5
6  #main program
7  tup_Odd = (1, 3, 5, 7, 9)
8  tup_Even = (2, 4, 6, 8, 10)
9
10 tup_Odd_Even = combineTuples(tup_Odd, tup_Even)
11 print(tup_Odd_Even)
```

```
Shell
Python 3.7.9 (bundled)
>>> %Run 'ood even.py'
  (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
>>>
```

7. The **"Names.txt"** contain names of students that submitted the project work on time. Transfer all names to a new list for sorting. The sorted list contents (ascending order) should be written to a new file called "**sortedNames.txt**" file. Then print the elements from the list in descending order but the order of list (ascending) should not be changed. [Hint: use sort() and sorted()]. The expected output will be

```
['Xu', 'Xiaboo', 'Wilson', 'Wang', 'Wali', 'Uolevi', 'Susan', 'Sundar', 'Roopa', 'Peter', 'Nuo', 'Norbu', 'Kathir', 'Kamal', 'Fatimah', 'Erno', 'Erkk
i', 'Domnik', 'Amar', 'Ali']

>>>
```