Programming 1: Lab 9 : Lists, Sets, Tuples and Dictionaries

Write the python code for the following questions. Handle all the valid and invalid test cases. Write down relevant comments in your code:

1. Take a list of positive integers as input in a list and perform the following operations:
   1. Remove duplicate elements. Do not use sets or any other in-built functions except len().
   2. Remove duplicate elements by typecasting into sets and displaying resulting elements in the same order as they appear in the input list.
2. Use list comprehensions for the following.
   1. Take a list of strings as input along with a sample string. Count the number of strings that contains the sample string as a substring.
   2. Take a list of integers as input. Find the square of only the +ve integers in the list, the rest are converted to 0s.
   3. Take a list of integers as input. Find the square of numbers that are in the range of 10-20, leave the rest as they are in the resultant list.
   4. Take a list of strings as input, convert all the strings into upper case, if they have the first character as lower case, leave the rest as they are.
3. Create a list of student record by taking the student name, rollno and total marks (out of 100) as input. Each record should be saved as a tuple element inside the list.
   1. Display the details of the student, who has the maximum marks
   2. Add an element to each student record corresponding to their Rank. The rank for student with highest marks is 1, and with lowest marks is N. Display the student details in ascending order of their ranks.
4. Represent 2D matrix of dimensions M x N, by taking M, N and matrix elements as input from the user. Perform following operations on these 2D matrices.
   1. Add 2 matrices. Display the values in the input matrices as well as the summation matrix.
   2. Multiply 2 matrices. Note that the input matrices need to be checked for compatible multiplication, i.e., 2 matrices with M x N and N x P dimensions can be multiplied, if number of columns in 1st matrix is equal to the number of rows in the 2nd matrix. Consider all the valid and invalid test cases and display the results.
   3. For a square matrix, check if the matrix is symmetric or not.
   4. Check if the matrix is an upper triangular matrix or a lower triangular matrix.
   5. Find the transpose of an M x N matrix.
   6. Take an NxN matrix with complex values and check if the matrix is a Hermitian matrix.
5. Take a paragraph from a user: calculate frequency of vowels, store it in dictionary. Display the frequency counts in ascending order of the counts.
6. Take numbers from a user until they say “over”. If user enters odd number, add it in dictionary named “Odd” with number as key value and its square and cube as values. If user enters even number, information will be moved to the dictionary named “Even”. Display the results.

   Sample: Even = {2:[4,8], 8:[64,512]}, Odd={1:[1,1], 5:[25,125]}