Machine Learning Workshop 1 – CSE 2793
MINOR ASSIGNMENT-4: STRINGS

Q 01   If the input string is test = **'Hello, How are you'**, then
   a. Find the length of the above string
   b. Print only 2 characters from the last word of the string
   c. Find **'hello'** in the above string
   d. Change the lowercase letter to uppercase of the above string and vice versa, Check if the string is in uppercase or not
   e. Separate the words of the string by a comma operator
   f. Replace the word **'Hello'** by **'Hi'**

Q 02   Write a function that takes a string as a parameter and returns a string with every successive repetitive character replaced with a star(*****).
   For example,
   `'balloon'`
   is returned as
   `'bal*o*n'`

Q 03   Write a function that takes two strings and returns **True** if they are anagrams and **False** otherwise. A pair of strings is anagrams if the letters in one word can be arranged to form the second one.

Q 04   Write a python program to replace that character, which is repeated maximum time in each string by '-' (dash).

Q 05   Write a function that takes a sentence as an input parameter and displays the number of words in the sentence.

Q 06   Write a function that takes a sentence as an input parameter and replaces the first letter of every word with the corresponding uppercase letter and rest of the letters in the word by corresponding letters in lowercase without using built-in function.

Q 07   Write a python program to count the number of odd and even length words present in a string.

Q 08   Write a python program to separate an input word according to the vowels present in the string.
   For Example:
   test = **'KlaGt'**
   Result: **['Kl', 'Gt']**

Q 09   Write a Python program to get a string made of the first 2 and last 2 characters of a given string. If the string length is less than 2, return the empty string instead.

Examples:

Sample String: `'w3resource'`
Expected Result: `'w3ce'`

Sample String: `'w3'`
Expected Result: `'w3w3'`

Sample String: `'w'`
Expected Result: `Empty String`

Q 10   Write a Python program to get a string from a given string where all occurrences of its first char have been changed to '$', except the first char itself.

Examples:
Sample String: `'restart'`
Expected Result: `'resta$t'`

Q 11   Write a program that requests a sentence, a word in the sentence, and another word and then displays the sentence with the first word replaced by the second.
Example:
```
Enter a sentence: What you don't know won't hurt you.
Enter word to replace: know
Enter replacement word: owe
What you don't owe won't hurt you.
```

Q 12   Although electronic deposit has become extremely popular, payroll and accounts payable applications often print checks. A serious problem is the intentional alteration of a check amount by someone who plans to cash a check fraudulently.
To prevent a dollar amount from being altered, some computerized check-writing systems employ a technique called check protection. Checks designed for printing by computer typically contain a fixed number of spaces for the printed amount. Suppose a paycheck contains eight blank spaces in which the computer is supposed to print the amount of a weekly paycheck. If the amount is large, then all eight of the spaces will be filled:
```
1,230.60 (check amount)
--------
01234567 (position numbers)
```
On the other hand, if the amount is smaller, then several of the spaces would ordinarily be left blank. For example,

```
399.87
--------
01234567
```

contains two blank spaces. If a check is printed with blank spaces, it's easier for someone to alter the amount. Check-writing systems often insert leading asterisks to prevent alteration and protect the amount as follows:

```
**399.87
--------
01234567
```

Write a script that inputs a dollar amount, then prints the amount in check-protected format in a field of 10 characters with leading asterisks if necessary.

[[Hint: In a format string that explicitly specifies alignment with **<**, **^** or **>**, you can precede the alignment specifier with the fill character of your choice.]]

Q 13   Ask the user to type in a word in upper case. If they type it in lower case, ask them to try again. Keep repeating this until they type in a message all in uppercase.

Q 14   Ask the user to type in a word and then display it backwards on separate lines. For instance, if they type in "Hello" it should display as shown below:
Example:

```
Enter a word: Hello
o
l
l
e
H
>>>
```

Q 15   Pig Latin takes the first consonant of a word, moves it to the end of the word and adds on an "**ay**". If a word begins with a vowel you just add "**way**" to the end. For example, **pig** becomes **igpay**, **banana** becomes **ananabay**, and **aadvark** becomes **aadvarkway**. Create a program that will ask the user to enter a word and change it into Pig Latin. Make sure the new word is displayed in lower case.