

---

# Machine Learning Workshop 1 – CSE 2793

## MINOR ASSIGNMENT-4: STRINGS

---

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

# Machine Learning Workshop 1 – CSE 2793

## MINOR ASSIGNMENT-5: LISTS, TUPLES, DICTIONARIES AND SETS

---

- Q 01 Write a Python program to create a list of integers of size N and store random values in it. Find and display the sum and average. Create two more lists that contain the even and odd values from this list.
- Q 02 Write a Python program using a list that reads the integers between 1 and 100 and counts the occurrences of each. Assume the input ends with 0.

Here is a sample run of the program:

```
Enter the integers between 1 and 100: 2 5 6 5 4 3 23 43 2 0
2 occurs 2 times
3 occurs 1 time
4 occurs 1 time
5 occurs 2 times
6 occurs 1 time
23 occurs 1 time
43 occurs 1 time
```

**Note that if a number occurs more than one time, the plural word “times” is used in the output.**

- Q 03 Write a function that finds the smallest element in a list of double values. Write a Python program that prompts the user to enter ten numbers, invokes this function to return the minimum value, and displays the minimum value.

Here is a sample run of the program:

```
Enter ten numbers: 1.9 2.5 3.7 2 1.5 6 3 4 5 2
The minimum number is: 1.5
```

- Q 04 Define a function that eliminates the duplicate values in the list. Write a Python program that reads in ten integers, invokes the method, and displays the result.

Here is the sample run of the program:

```
Enter ten numbers: 1 2 3 2 1 6 3 4 5 2

The distinct numbers are: 1 2 3 6 4 5
```

- Q 05 Define a function that returns the sum of all the elements in a specified column in a matrix. Write a Python program that reads a 3-by-4 matrix and displays the sum of each column.

Here is a sample run:

```
Enter a 3-by-4 matrix row by row:
1.5  2   3   4
```

5.5	6	7	8
9.5	1	3	1

**Sum of the elements at column 0 is 16.5**

**Sum of the elements at column 1 is 9.0**

**Sum of the elements at column 2 is 13.0**

**Sum of the elements at column 3 is 13.0**

- Q 06 Define a function to add two matrices. In order to be added, the two matrices must have the same dimensions and the same or compatible types of elements. Let  $c$  be the resulting matrix. Each element  $c_{ij}$  is  $a_{ij} + b_{ij}$ . For example, for two  $3 \times 3$  matrices  $a$  and  $b$ ,  $c$  is

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} + \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} = \begin{pmatrix} a_{11} + b_{11} & a_{12} + b_{12} & a_{13} + b_{13} \\ a_{21} + b_{21} & a_{22} + b_{22} & a_{23} + b_{23} \\ a_{31} + b_{31} & a_{32} + b_{32} & a_{33} + b_{33} \end{pmatrix}$$

Write a Python program that prompts the user to enter two  $3 \times 3$  matrices and displays their sum.

- Q 07 Define a function to multiply two matrices.

For example, for two  $3 \times 3$  matrices  $a$  and  $b$ ,  $c$  is

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \times \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{pmatrix}$$

where  $c_{ij} = a_{i1} * b_{1j} + a_{i2} * b_{2j} + a_{i3} * b_{3j}$ .

Write a Python program that prompts the user to enter two  $3 \times 3$  matrices and displays their product.

- Q 08 Write a Python program that randomly fills in 0s and 1s into a 4-by-4 matrix, prints the matrix, and finds the first row and column with the most 1s.

Here is a sample run of the program:

```
0 0 1 1
0 0 1 1
1 1 0 1
1 0 1 0
```

**The largest row index: 2**

**The largest column index: 2**

- Q 09 You are supposed to remove duplicate entries from a list. The constraint is that the original order should be preserved. Write function **remove\_duplicates(values)**.

- Q 10 Given two lists of numbers, each sorted in ascending order, merge them into a result list according to their order. Write function **mergeList(values1, values2)**.

Here are some test cases:

---

Input 1	Input 2	Result
1, 4, 7, 12, 20	10, 15, 17, 33	1, 4, 7, 10, 12, 15, 17, 20, 33
2, 3, 5, 7	11, 13, 17	2, 3, 5, 7, 11, 13, 17
2, 3, 5, 7, 11	7, 11, 13, 17	2, 3, 5, 7, 7, 11, 11, 13, 17
[1, 2, 3]	$\emptyset = []$	[1, 2, 3]

---

- Q 11 Write function **is\_magic\_triangle(values)** that checks whether a sequence of numbers forms a magic triangle. Such a triangle is defined as one where the respective sums of the three sides' values must all be equal. The sequence of numbers must be entered in the form of a list.

Here is a sample run:

```
>>> is_magic6([1, 5, 3, 4, 2, 6])
True
```

```
>>> is_magic6([1, 2, 3, 4, 5, 6])
False
```

- Q 12 Create a tuple containing the names of five countries and display the whole tuple. Ask the user to enter one of the countries that have been shown to them and then display the index number (i.e. position in the list) of that item in the tuple.
- Q 13 Add to program in Q 12 to ask the user to enter a number and display the country in that position.
- Q 14 Write a program that prints up to  $n^{\text{th}}$  Fibonacci number by using a dictionary.
- Q 15 Write a program that prints a histogram of frequencies of characters occurring in a message supplied by the user.

---

# Machine Learning Workshop 1 – CSE 2793

## MINOR ASSIGNMENT-6: RECURSION, FILES AND EXCEPTIONS

---

- Q 01 Write function **fib\_rec(n)** that recursively computes Fibonacci numbers.
- Q 02 Calculate the sum of the digits of a number recursively. Write recursive function **calc\_sum\_of\_digits(value)** for this purpose.
- Q 03 Write function **gcd(a, b)** that computes the greatest common divisor (GCD) recursively.
- Q 04 Write function **lcm(a, b)** that computes the lowest common multiplier (LCM) recursively.
- Q 05 Write recursive function **reverse\_string(text)** that flips the letters of the text passed in.
- Q 06 Write function **to\_binary(n)** that recursively converts the given positive integer into a textual binary representation. No call to **int(x, base)** may be used.
- Q 07 Write conversions to octal and hexadecimal numbers by implementing the corresponding functions **to\_octal(n)** and **to\_hex(n)**. Again, no call to **int(x, base)** may be used.
- Q 08 Write recursive function **power\_of(value, exponent)** that exponentiates the given positive integer with the positive number specified as second parameter.  
For example, the call **power\_of(4, 2)** should return the square of 4, so it computes  $4^2 = 16$ . You may not use the built-in functionality **pow()** or the operator **\*\***.
- Q 09 A palindrome is a word that reads the same from the front and the back. You can extend this definition to the digits of a number. Write recursive function **is\_number\_palindrome(number)** but without converting the number into a string and then using string functionalities like **[::-1]**.
- Q 10 Write a recursive function **harmonic\_sum(n)** to calculate the harmonic sum of first n terms.  
Note: The harmonic sum is the sum of reciprocals of the positive integers. For example, if  $n = 4$ , the output should be  $(1 + 1/2 + 1/3 + 1/4) = 2.0833$
- Q 11 Write a Python function that takes two file names, file1 and file2 as input. The function should read the contents of the file file1 line by line and should write them to another file file2 after adding a newline at the end of each line.
- Q 12 Write a Python function that reads a file file1 and displays the number of words and the number of vowels in the file.

Q 13 Write a Python function that takes two files of equal size as input from the user. The first file contains weights of items, and the second file contains corresponding prices. Create another file that should contain price per unit weight for each item.

Q 14 What will be the output produced on executing function `inverse1` when the following input is entered as the value of variable `num`:

- (a)5                      (b)0                      (c)2.0                      (d)x                      (e)None

```
Def inverse1():  
    try:  
        num = input('Enter the number: ')  
        num = float(num)  
        inverse = 1.0 / num  
    except ValueError:  
        print('ValueError')  
    except TypeError:  
        print('TypeError')  
    except ZeroDivisionError:  
        print('ZeroDivisionError')  
    except:  
        print('Any other Error')  
    else:  
        print(inverse)  
    finally:  
        print('Function inverse completed')
```

Q 15 Examine the following function `percentage`:

Determine the output for the following function calls:

- (a) `percentage(150.0, 200.0)`  
(b) `percentage(150.0, 0.0)`  
(c) `percentage('150.0', '200.0')`

```
def percentage(marks, total):  
    try:  
        percent = (marks / total) * 100  
    except ValueError:  
        print('ValueError')  
    except TypeError:  
        print('TypeError')  
    except ZeroDivisionError:  
        print('ZeroDivisionError')  
    except:  
        print('Any other Error')  
    else:  
        print(percent)  
    finally:  
        print('Function percentage completed')
```