Machine Learning

22AIE213

Assignment-1

Team 17:

Y Nishanth (AIE22165)

Meghraj G K (AIE22129)

Akshat Kuttan (AIE22104)

Odd Assignment:

**Q1)**

* Firstly a count variable is defined and two for loops are run to access 2 elements of the list at a time.
* If the sum of both the elements accessed is 10 then the count variable is incremented.
* Then the count value is divided by 2 as it counts the same pair twice and is printed.

**Pseudo Code:**

Set count to

For each element i in arr:

For each element j in arr:

If (i + j) equals 10:

Increment count by 1

Print "The count value is:", int(count/2)

**Q2)**

* Firstly the no. of elements in the list is input and an empty list with all zeroes is created.
* Two for loops are used to access 2 elements of the list simultaneously and they are compared.
* The highest difference is then stored into the variable and is printed.

**Pseudo Code:**

Input: n (number of elements in the list)

Set inp\_list to a list of n elements, initialized to 0

Print "Enter the elements"

For i in range from 0 to n-1:

Input element and store it in inp\_list[i]

Set diff to 0

For each element i in inp\_list:

For each element j in inp\_list:

If (int(i) - int(j)) is greater than diff:

Set diff to (int(i) - int(j))

Print "The range of the given list of real numbers is:", diff

**Q3)**

* A function called matrix\_multiply is created for the matrix multiplication where a copy of the user input matrix is created.
* The result is returned to the main function where all the information like the matrix and the power is input from the user.
* Then the matrix multiplication function is called for power-1 times using a for loop.

**Pseudo Code:**

Function third\_question():

MatrixA = []

size = Input("Enter the size of the matrix")

For i in range from 0 to size - 1:

row\_list = []

For j in range from 0 to size - 1:

number = Input("Enter element in row {i+1} column {j+1}")

Append number to row\_list

Append row\_list to MatrixA

power = Input("Enter the power to which you want the matrix: ")

If power > 0:

For \_ in range from 0 to power - 2:

MatrixA = matrix\_multiply(MatrixA, size)

Print "The final matrix is:"

For i in range from 0 to size - 1:

For j in range from 0 to size - 1:

Print MatrixA[i][j], end=" "

Print newline

Function matrix\_multiply(matA, size):

matB = Copy each row of matA # Create a copy of MatrixA

result = Create a size x size matrix filled with zeros

For i in range from 0 to size - 1:

For j in range from 0 to size - 1:

For k in range from 0 to size - 1:

result[i][j] += matA[i][k] \* matB[k][j]

Return result

**Q4)**

* The required sentence is input from the user, an empty dictionary and a count variable is initiated.
* A for loop is run to access each and every element in the sentence and is checked whether it exists in the dictionary and is checked whether if it is an alphabet.
* Then, another loop is run to check whether both the letters are the same. If yes then count variable is incremented.
* Then the dictionary is updated with the alphabet and the count value and the maximum value out of it is accessed and printed.

**Pseudo Code:**

Input: sentence (string)

Set an empty dictionary dict

Set count to 0

For each character i in sentence:

If i is not in dict keys and i is an alphabet:

For each character j in sentence:

If i equals j:

Increment count by 1

Add i as a key to dict with value count

Set count to 0

Set value to the maximum value in dict

For each key i in dict:

If the value associated with i in dict is equal to value:

Set key to i

Print "The highest recurring alphabet is", key, "and it occurs", value, "times"

**EVEN Question:**

**Question 1:**

Set vowels = 0 # Initialize vowels variable

Set consonants = 0 # Initialize consonant variable

Input a string

Convert the string to lowercase

Remove spaces from the string

For each character in the string:

If the character is (a, e, i, o, u):

Vowels+=1

Else:

Increment the consonant count

Print "The number of vowels:", vowels

Print "The number of consonants:", consonants

**Description:**

In this code the variables are set as counters to count the vowels and the consonants, after inputting the string it is converted to lowercase and removes the empty spaces. The given string is then put through a for-loop and the string is compared to a list of vowels and the count increases in the vowels variables if there is a vowel or else the consonants variableincreases. The value of the vowels and consonants is then printed.

**Question 2:**

Initialize an empty list matA

Initialize an empty list matB

Input the number of rows

Input the number of columns

For each row in the range of rows:

Initialize an empty list matrowA

For each column in the range of columns:

Input an element and append it to matrowA

Append matrowA to matA

For each row in the range of rows:

Initialize an empty list matrowB

For each column in the range of columns:

Input an element and append it to matrowB

Append matrowB to matB

Define a function multiply(matA, matB):

Initialize an empty result matrix with dimensions len(matA) rows by len(matB[0]) columns, filled with zeros

For each row i in the range of length of matA:

For each column j in the range of length of matB[0]:

For each k in the range of length of matB:

Update result[i][j] by adding the product of matA[i][k] and matB[k][j]

Return the result matrix

Call multiply function with matA and matB and store the result in a variable called answer

For each row in answer:

Print the row

**Description:**

Two empty lists are created termed matA and matB which are the matrices. The rows and columns of the matrix is taken from the user. A nested for loop(i and j) is used to enter the values to the matrix, num is used to take the elements from the user and is later appended to a matArow and that list is appened to a matA list similar is done for matB. A function multiple is defined where a empty result list is initialised woth zeros, a nested for loop(I,j,k) is used to multiply the two matrices. The output is then printed as rows using for loop.

**Question 3:**

Initialize an empty list called list1

Initialize an empty list called list2

Input the number of elements in the lists

For each element in the range of the number of elements:

Input an element and append it to list1

For each element in the range of the number of elements:

Input an element and append it to list2

Define a function called commonele(list1, list2):

Initialize an empty list called common

For each element i in list1:

For each element j in list2:

If i is equal to j:

Append i to the common list

Return the common list

Print the result of calling commonele function with list1 and list2 as arguments

**Description:**

2 empty lists are initialised, elements are accepted in both the lists, later a function is defined called commonele, by running the lists through a nested for loop(i,j) which acts as pointers to both the list the values in the list is then compared and a empty list called common is also initialled. If both the list have common elements it will stored in common and is printed.

**Question 4:**

Input the number of rows

Input the number of columns

Define a function called transmatrix(rows, cols):

Initialize an empty list called matrix

Initialize an empty list called tramat

For each row in the range of rows:

Initialize an empty list called matrow

Initialize an empty list called tra\_matrow

For each column in the range of columns:

Input a number and append it to matrow and tra\_matrow

Append matrow to matrix

Append tra\_matrow to tramat

Return matrix and a list comprehension to transpose the matrix

Print the result of calling the transmatrix function with rows and columns as arguments

**Description:**

The values of rows and columns are taken from the user a function called transmatrix, a list called matrow and tramat to store the values. A nested for loop(i,j) is used to add the values to the matrix as rows and values are appended to matrow and tramat and the rows are then appended to matrix and tramat which are the two list. While printing the [j][i] is printed of the tramat and the original matrix is also returned.