3.

def common(l1, l2):

set1 = set(l1)

set2 = set(l2)

common\_elements = set1.intersection(set2)

return len(common\_elements)

l1 = list(map(int, input("Enter the elements of the first list").split()))

l2 = list(map(int, input("Enter the elements of the second list").split()))

num = common(l1, l2)

print(f"The number of common elements between the two lists is: {num}")

------------------------------------------------------------------------------------------------------------------------------------

4.

def input\_matrix():

    rows = int(input("Enter the number of rows for the matrix: "))

    cols = int(input("Enter the number of columns for the matrix: "))

    matrix = []

    print("Enter the elements of the matrix, row by row, each row's elements separated by spaces:")

    for i in range(rows):

        while True:

            row = input(f"Row {i+1}: ").split()

            if len(row) != cols:

                print(f"Error: Row {i+1} does not have {cols} columns. Please enter again.")

            else:

                matrix.append(list(map(int, row)))

                break

    return matrix

def transpose\_matrix(matrix):

    rows = len(matrix)

    cols = len(matrix[0])

    transpose = [[0 for \_ in range(rows)] for \_ in range(cols)]

    for i in range(rows):

        for j in range(cols):

            transpose[j][i] = matrix[i][j]

    return transpose

def print\_matrix(matrix):

    for row in matrix:

        print(' '.join(map(str, row)))

matrix = input\_matrix()

transpose = transpose\_matrix(matrix)

print("The transpose of the matrix is:")

print\_matrix(transpose)

--------------------------------------------------------------------------------------------------------------------------------------

1.

def count(s):

    vowels = 'aeiouAEIOU'

    vowel = 0

    consonant = 0

    for char in s:

        if char.isalpha():

            if char in vowels:

                vowel += 1

            else:

                consonant += 1

    return vowel, consonant

input\_string = input("Enter a string: ")

vowels, consonants = count(input\_string)

print(f"Number of vowels: {vowels}")

print(f"Number of consonants: {consonants}")

2.

def input\_matrix(name):

    rows = int(input(f"Enter the number of rows for matrix {name}: "))

    cols = int(input(f"Enter the number of columns for matrix {name}: "))

    matrix = []

    print(f"Enter the elements of matrix {name}, row by row, each row's elements separated by spaces:")

    for i in range(rows):

        while True:

            row = input(f"Row {i+1}: ").split()

            if len(row) != cols:

                print(f"Error: Row {i+1} does not have {cols} columns. Please enter again.")

            else:

                matrix.append(list(map(int, row)))

                break

    return matrix

def multiply\_matrices(A, B):

    if len(A[0]) != len(B):

        return "Error: Matrices A and B cannot be multiplied due to incompatible dimensions."

    result = [[0 for \_ in range(len(B[0]))] for \_ in range(len(A))]

    for i in range(len(A)):

        for j in range(len(B[0])):

            for k in range(len(B)):

                result[i][j] += A[i][k] \* B[k][j]

    return result

def print\_matrix(matrix):

    for row in matrix:

        print(' '.join(map(str, row)))

# Input matrices A and B

try:

    A = input\_matrix('A')

    B = input\_matrix('B')

    # Multiply matrices

    result = multiply\_matrices(A, B)

    if isinstance(result, str):

        print(result)

    else:

        print("The product of matrices A and B is:")

        print\_matrix(result)

except ValueError as e:

    print(e)