```
#Aim:WAP in python to transpose and find diagonal elements of a matrix
# Branch: Comps
# Year: SE
# Sem: IV
# Subject: Python
# Name: Sidra Shaikh
# UIN: 231P064
# Roll No: 40
from numpy import *
 # accept rows and column
r,c=[int(a) for a in input("Enter rows and column :").split()]


# accept matrix element as a string
str= input(" Enter Matrix Elements :\n")

# convert string into matrix with size r*c
x= reshape(matrix(str),(r,c))
print(" original matrix : ")
print(x)


print(" Transpose matrix : ")
y=x.transpose()
print(y)

print(" Diagonal of a matrix : ")
y=diagonal (x)
print(y)
print("Sum ofdiagonal : ")
print(sum(y))


#Aim:Write a program to perform matrix multiplication.
# Branch: Comps
# Year: SE
# Sem: IV
# Subject: Python
# Name: Sidra Shaikh
# UIN: 231P064
# Roll No: 40
def matrix_multiplication(A, B):
    # Get the dimensions of matrices
    rows_A, cols_A = len(A), len(A[0])
    rows_B, cols_B = len(B), len(B[0])

    # Check if multiplication is possible
    if cols_A != rows_B:
        raise ValueError("Number of columns in A must be equal to number of rows in
B")
```

```
    # Initialize result matrix with zeros
    result = [[0] * cols_B for _ in range(rows_A)]

    # Perform multiplication
    for i in range(rows_A):
        for j in range(cols_B):
            result[i][j] = sum(A[i][k] * B[k][j] for k in range(cols_A))

    return result

# Function to take matrix input
def input_matrix(rows, cols):
    print(f"Enter elements for a {rows}x{cols} matrix:")
    return [[int(input(f"Element [{i+1}][{j+1}]: ")) for j in range(cols)] for i in
range(rows)]

# Taking input from user
rows_A = int(input("Enter number of rows for matrix A: "))
cols_A = int(input("Enter number of columns for matrix A: "))
A = input_matrix(rows_A, cols_A)

rows_B = int(input("Enter number of rows for matrix B: "))
cols_B = int(input("Enter number of columns for matrix B: "))
B = input_matrix(rows_B, cols_B)

# Perform matrix multiplication
try:
    result = matrix_multiplication(A, B)
    print("Resultant Matrix:")
    for row in result:
        print(row)
except ValueError as e:
    print(e)


#Aim:Write a program to perform transpose of a matrix.
# Branch: Comps
# Year: SE
# Sem: IV
# Subject: Python
# Name: Sidra Shaikh
# UIN: 231P064
# Roll No: 40
def transpose_matrix(matrix):
    rows, cols = len(matrix), len(matrix[0])
    transposed = [[matrix[j][i] for j in range(rows)] for i in range(cols)]
    return transposed

# Function to take matrix input
```

```
def input_matrix(rows, cols):
    print(f"Enter elements for a {rows}x{cols} matrix:")
    return [[int(input(f"Element [{i+1}][{j+1}]: ")) for j in range(cols)] for i in
range(rows)]

# Taking input from user
rows = int(input("Enter number of rows for the matrix: "))
cols = int(input("Enter number of columns for the matrix: "))
matrix = input_matrix(rows, cols)

# Compute transpose
transposed_matrix = transpose_matrix(matrix)

# Display result
print("Transposed Matrix:")
for row in transposed_matrix:
    print(row)
```