Import numpy as np

#1. Define and print a 6 dimentional vector

X=np.array([[1,1,1,1,1,1],[2,2,2,2,2,2],[3,3,3,3,3,3],[4,4,4,4,4,4],[5,5,5,5,5,5],[6,6,6,6,6,6]])

Print(x)

#2. Print the transpose of the above vector

X=np.array([[1,1,1,1,1,1],[2,2,2,2,2,2],[3,3,3,3,3,3],[4,4,4,4,4,4],[5,5,5,5,5,5],[6,6,6,6,6,6]]).T

Print(X)

#3. Define two non square matrices such that they can be mulplied.

A=np.array([[1,2,6],[8,2,9]])

B=np.array([[2,5],[8,1],[2,6]])

Print(a)

Print(b)

#4. Print the shape of the above matrices

Print(a.shape)

Print(b.shape)

#5. Print the product of above two matrices (do so without using the inbuilt functions).

Z=np.array([np.zeros(3)]\*3)

For I in range(len(a)):

For j in range(len(b[1])):

For k in range(len(b)):

Z[i][j] += a[i][k] \* b[k][j]

Print(Z)

#6. Define two non square matrices of same order and print their sum.

P=np.array([[5,3,9],[9,2,4],[9,4,7]])

Q=np.array([[9,3,6],[5,2,8],[8,1,2]])

Print(p+q)

#7. Define a square matrix A.

A=np.array([[7,3],[9,2]])

Print(A)

#8. Print the transpose of A.

Print(A.T)

#9. Print the identity matrix of the above order I.

I=np.array([[1,0],[0,1]])

Print(I)

#10. Verify A.I = I.A for matrix multiplication.

X=A@I

Print(“A.I = “,X)

Y=I@A

Print(“I.A = “,Y)

Print(“ Therefore, A.I = I.A”)

#11. Define another square matrix of the same order as A.

M=np.array([[4,1],[7,2]])

Print(m)

#12. Print the product of the matrices as matrix multiplication

Print(A@m)

#13. Print the product of the matrices by element wise multiplication

Print(np.multiply(A,m))

#14. Calculate and print the inverse of A.

C=np.linalg.det(A)

Print(“The determinant of A is”,c)

If c!=0:

Print(“The inverse is “,np.linalg.inv(A))

Else:

Print(“Inserve does not exist”)