**!**pip install cupy-cuda11x

USE\_GPU **=** **False**

**if** USE\_GPU:

**import** cupy **as** xp

**else**:

**import** numpy **as** xp

*# User Input*

N **=** int(input("Enter size of vector/matrix (N): "))

*# Input Vectors*

print("Enter elements for Vector A:")

A **=** xp**.**array([int(input(f"A[{i}]: ")) **for** i **in** range(N)])

print("Enter elements for Vector B:")

B **=** xp**.**array([int(input(f"B[{i}]: ")) **for** i **in** range(N)])

*# Vector Addition*

C **=** A **+** B

print("\nVector A:", A)

print("Vector B:", B)

print("Addition (A + B):", C)

*# Input Matrices*

print("\nEnter elements for Matrix D:")

D **=** xp**.**zeros((N, N), dtype**=**int)

**for** i **in** range(N):

**for** j **in** range(N):

D[i, j] **=** int(input(f"D[{i}][{j}]: "))

print("Enter elements for Matrix E:")

E **=** xp**.**zeros((N, N), dtype**=**int)

**for** i **in** range(N):

**for** j **in** range(N):

E[i, j] **=** int(input(f"E[{i}][{j}]: "))

*# Matrix Multiplication*

F **=** D **@** E

print("\nMatrix D:")

print(D)

print("\nMatrix E:")

print(E)

print("\nMultiplication Result (D x E):")

print(F)

Output

Enter size of vector/matrix (N): 3

Enter elements for Vector A:

A[0]: 1

A[1]: 2

A[2]: 3

Enter elements for Vector B:

B[0]: 4

B[1]: 5

B[2]: 6

Vector A: [1 2 3]

Vector B: [4 5 6]

Addition (A + B): [5 7 9]

Enter elements for Matrix D:

D[0][0]: 1

D[0][1]: 2

D[0][2]: 3

D[1][0]: 4

D[1][1]: 5

D[1][2]: 6

D[2][0]: 7

D[2][1]: 8

D[2][2]: 9

Enter elements for Matrix E:

E[0][0]: 9

E[0][1]: 8

E[0][2]: 7

E[1][0]: 6

E[1][1]: 5

E[1][2]: 4

E[2][0]: 3

E[2][1]: 2

E[2][2]: 1

Matrix D:

[[1 2 3]

[4 5 6]

[7 8 9]]

Matrix E:

[[9 8 7]

[6 5 4]

[3 2 1]]

Multiplication Result (D x E):

[[ 30 24 18]

[ 84 69 54]

[138 114 90]]