**Assignment No.:1C**

**Aim:**

Perform matrix addition subtraction and multiplication using NumPy Library.

**Theory:**

NumPy arrays are powerful data structures in Python that support efficient numerical and matrix operations. They offer advanced mathematical functions and are optimized for performance compared to standard lists.

**Basic Operation:**

* **Addition**: Element-wise addition of two arrays.
  + m1 = np.array([[1,4,7],[2,5,8],[3,4,5]])
  + m2 = np.array([[1,4,5],[1,5,2],[5,6,7]])
  + np.add(m1, m2)
* **Subtraction**: Element-wise subtraction of two arrays.
  + m1 = np.array([[1,4,7],[2,5,8],[3,4,5]])
  + m2 = np.array([[1,4,5],[1,5,2],[5,6,7]])
  + np.subtract(m1, m2)
* **Matrix Multiplication**: Uses the np.dot() function for matrix multiplication.
  + m1 = np.array([[1,4,7],[2,5,8],[3,4,5]])
  + m2 = np.array([[1,4,5],[1,5,2],[5,6,7]])
  + np.dot(m1, m2)
* **Division**: Element-wise division of two arrays.
  + m1 = np.array([[1,4,7],[2,5,8],[3,4,5]])
  + m2 = np.array([[1,4,5],[1,5,2],[5,6,7]])
  + np.divide(m1,m2)
* **Modulo**: Element-wise modulo operation
  + m1 = np.array([[1,4,7],[2,5,8],[3,4,5]])
  + m2 = np.array([[1,4,5],[1,5,2],[5,6,7]])
  + np.mod(m1,m2)
* **Elementwise Multiplication:**
  + m1 = np.array([[1,4,7],[2,5,8],[3,4,5]])
  + m2 = np.array([[1,4,5],[1,5,2],[5,6,7]])
  + np.multiply(m1, m2)

**Features of NumPy:**

* **Array Creation**: NumPy's core feature is the ndarray (N-dimensional array) object. Arrays can be created from Python lists or tuples using np.array(), or using functions like np.zeros(), np.ones(), and np.arange().
* **Array Attributes**: Important attributes include .shape (dimensions), .dtype (data type), and .size (total number of elements).
* **Basic Operations**: NumPy supports element-wise operations like addition, subtraction, multiplication, and division.
* **Statistical Functions**: Functions like np.mean(), np.median(), np.std() calculate mean, median, and standard deviation.

**Algorithm:**

**Step-1:** Start

**Step-2:** Import NumPy as np

**Step-3:** Declare array using np

**Step-4:** Define two matrix

m1 = np.array([[1,4,7],[2,5,8],[3,4,5]])

m2 = np.array([[1,4,5],[1,5,2],[5,6,7]])

**Step-5:** Declare add function np.add(m1, m2))

Store it in result addition

return result

**Step-6:** Declare sub function np.subtract(m1, m2)

Store it in result subtraction

return result

**Step-7:** Declare multiplication function np.dot(m1,m2)

Store it in result result\_dot

return result

**Step-8:** Declare division function np.divide(m1,m2)

Store it in result result\_divide

return result

**Step-9:** Declare modulus function mod(m1,m2)

Store it in result result\_mod

return result

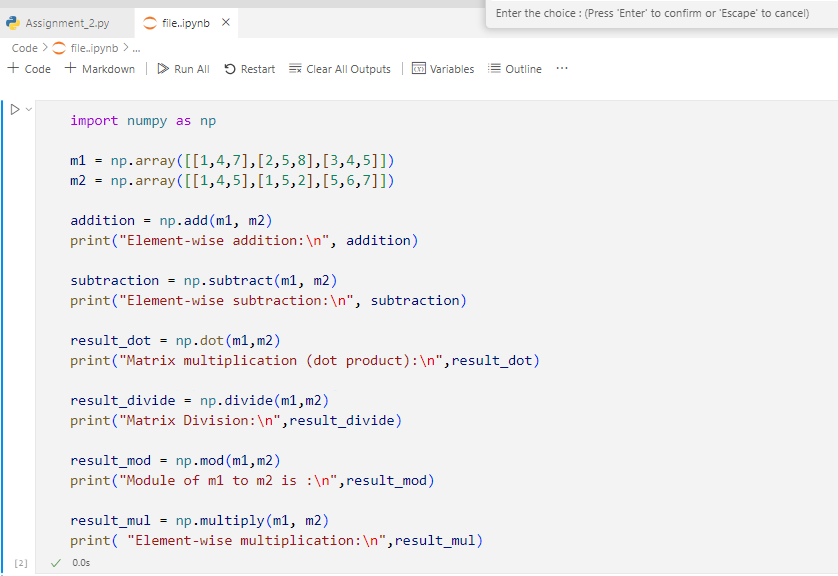
**Step-10:** Stop

**Datasets :**

The code uses two 3x3 NumPy arrays, m1 and m2, to demonstrate operations such as element-wise addition, subtraction, multiplication, division, modulo, and matrix multiplication (dot product). These datasets facilitate various matrix and numerical computations with NumPy functions.

**Conclusion :**

The code effectively demonstrates basic NumPy operations on two 3x3 matrices, showcasing element-wise addition, subtraction, multiplication, division, modulo, and matrix multiplication. By leveraging NumPy's functions, it efficiently performs and prints these operations, highlighting the library's capability for advanced numerical computations.

**Code:**

**Output :**

