

Q1. Create two 3×3 matrices using the random function in Numpy and perform the following operations.  
 è Product (prod)  
 è Multiplication (multiply)  
 è Dot Product (dot)

```
In [ ]: from numpy import *
x = np.random.randint(1,10,(3,3))
print(x)

print()
y = np.random.randint(1,10,(3,3))
print(y)

# PRODUCT
print()
print("PRODUCT : ")
prod = matmul(x, y)
print(prod)

# DOT
print()
print("DOT PRODUCT : ")
prod = dot(x, y)
print(prod)

# MULTIPLY
print()
print("MULTIPLICATION : ")
prod = multiply(x, y)
print(prod)
```

```
[[7 1 8]
 [4 8 3]
 [9 6 6]]
```

```
[[7 2 3]
 [1 4 1]
 [2 9 4]]
```

```
PRODUCT :
[[66 90 54]
 [42 67 32]
 [81 96 57]]
```

```
DOT PRODUCT :
[[66 90 54]
 [42 67 32]
 [81 96 57]]
```

```
MULTIPLICATION :
[[49  2 24]
 [ 4 32  3]
 [18 54 24]]
```

Q2. Perform the following set operations using the Numpy functions.

- è Union
- è Intersection
- è Set difference
- è XOR

```
In [ ]: set1 = np.array([1,2,3,4,5])
        set2 = np.array([7,8,9,0,4])

        print("UNION : ", union1d(set1,set2))
        print("INTERSECTIOPN", intersect1d(set1,set2))
        print("SET DIFFERENCE", setdiff1d(set1,set2))
        print("XOR", bitwise_xor(set1,set2))
```

```
UNION : [0 1 2 3 4 5 7 8 9]
INTERSECTIOPN [4]
SET DIFFERENCE [1 2 3 5]
XOR [ 6 10 10  4  1]
```

Q3. Create a 1D array using Random function and perform the following operations.

- è Cumulative sum
- è Cumulative Product
- è Discrete difference (with n=3)
- è Find the unique elements from the array

```
In [ ]: x = random.randint(1,10,(6))

        print(x)
        print("Cumulative sum :", cumsum(x))
        print("Cumulative Product :", cumprod(x))
        print("Unique : ", unique(x))
        print("Discrete Difference : ", diff(x,n=3))
```

```
[3 7 2 6 3 1]
Cumulative sum : [ 3 10 12 18 21 22]
Cumulative Product : [  3  21  42 252 756 756]
Unique : [1 2 3 6 7]
Discrete Difference : [ 18 -16  8]
```

Q4. Create two 1D array and perform the Addition using zip(), add() and user defined function (frompyfunc())

```
In [ ]: x = array([1,2,3,4,5])
        y = array([7,8,9,0,12])

        print("Using zip()", [n+m for n, m in zip(x,y)])
        print("Using + ", x+y)
        print("User def function : ", add(x, y))

        def add(set1, set2):
            for i in range(len(set1)):
                set1[i] += set2[i]

            return set1
```

```
Using zip() [8, 10, 12, 4, 17]  
Using + [ 8 10 12  4 17]  
User def function : [ 8 10 12  4 17]
```

### Q5. Find the LCM (Least Common Multiple) and GCD (Greatest Common Divisor) of an array of elements using reduce().

```
In [ ]: from functools import reduce  
import numpy as np  
from math import gcd  
  
input_str = input("Enter values for the array comma-separated: ")  
a = np.array([int(x) for x in input_str.split(',')])  
  
def calculate_lcm(x, y):  
    return x * y // gcd(x, y)  
  
def calculate_gcd(x, y):  
    while y != 0:  
        x, y = y, x % y  
    return x  
  
lcm = reduce(calculate_lcm, a)  
gcd = reduce(calculate_gcd, a)  
  
print("LCM:", lcm)  
print("GCD:", gcd)
```

```
LCM: 56  
GCD: 2
```