Q1. Create two 3×3 matrices using the random function in Numpy and perform the following operations.

1. Product (prod)
2. Multiplication (multiply)
3. Dot Product (dot)

In [ ]:
```
import numpy as np

# Create two random 3x3 matrices
matrix1 = np.random.randint(30, size=(3, 3))
matrix2 = np.random.randint(30, size=(3, 3))

# Perform matrix operations

# Product (element-wise multiplication)
product_result = np.prod([matrix1, matrix2], axis=0)

# Multiplication
multiplication_result = np.multiply(matrix1, matrix2)

# Dot Product
dot_product_result = np.dot(matrix1, matrix2)

print("Matrix 1:")
print(matrix1)
print("\nMatrix 2:")
print(matrix2)
print("\nProduct of the MATRIX(element-wise multiplication):")
print(product_result)
print("\nMultiplication of the MAtrix:")
print(multiplication_result)
print("\nDot Product Of two Matrix:")
print(dot_product_result)
```

```
Matrix 1:
[[ 1 13 20]
 [ 5 19 25]
 [27 13  9]]

Matrix 2:
[[ 5 21  0]
 [ 5 19 16]
 [ 6 23 28]]

Product of the MATRIX(element-wise multiplication):
[[  5 273   0]
 [ 25 361 400]
 [162 299 252]]

Multiplication of the MAtrix:
[[  5 273   0]
 [ 25 361 400]
 [162 299 252]]

Dot Product Of two Matrix:
[[ 190  728  768]
 [ 270 1041 1004]
 [ 254 1021  460]]
```

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

1. Union
2. Intersection
3. Set difference
4. XOR

In [ ]:
```
import numpy as np

# Create two arrays
array1 = np.array([11, 22, 33, 44, 55])
array2 = np.array([33, 30, 55, 66, 77])

# Union
union_result = np.union1d(array1, array2)

# Intersection
intersection_result = np.intersect1d(array1, array2)

# Set Difference
set_difference_result = np.setdiff1d(array1, array2)

# XOR (Exclusive OR)
xor_result = np.setxor1d(array1, array2)

print("Array 1:", array1)
print("Array 2:", array2)
print("Union Of the Arrays:", union_result)
print("Intersection of the Array :", intersection_result)
print("Set Difference of the two Array (Array 1 - Array 2):", set_differe
print("XOR:", xor_result)
```

```
Array 1: [11 22 33 44 55]
Array 2: [33 30 55 66 77]
Union Of the Arrays: [11 22 30 33 44 55 66 77]
Intersection of the Array : [33 55]
Set Difference of the two Array (Array 1 - Array 2): [11 22 44]
XOR: [11 22 30 44 66 77]
```

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

1. Cumulative sum
2. Cumulative Product
3. Discrete difference (with n=3)
4. Find the unique elements from the array

In [ ]:
```python
import numpy as np

# Create a random 1D array
array = np.random.randint(50, size=(5))  # You can change the size of the

# Cumulative sum
cumulative_sum_result = np.cumsum(array)

# Cumulative product
cumulative_product_result = np.cumprod(array)

# Discrete difference with n=3
discrete_difference_result = np.diff(array, n=3)

# Find unique elements
unique_elements = np.unique(array)

print("Original Array:")
print(array)
print("\nCumulative Sum of the array:")
print(cumulative_sum_result)
print("\nCumulative Product of the array:")
print(cumulative_product_result)
print("\nDiscrete Difference of the array (n=3):")
print(discrete_difference_result)
print("\nUnique Elements of the Array :")
print(unique_elements)
```

```
Original Array:
[48 22 29 48  1]

Cumulative Sum of the array:
[ 48  70  99 147 148]

Cumulative Product of the array:
[     48    1056   30624 1469952 1469952]

Discrete Difference of the array (n=3):
[-21 -78]

Unique Elements of the Array :
[ 1 22 29 48]
```

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

In [ ]:
```
import numpy as np
x = [1, 2, 3, 4]
y = [4, 5, 6, 7]
z = []

for i, j in zip(x, y):
  z.append(i + j)
print("zip of array x and y ",z)

x = [11, 22, 33, 44]
y = [55, 66, 77, 88]
res = np.add(x, y)
print("addition of element of the 2 arrays",res)


def myadd(x, y):
  return x+y

myadd = np.frompyfunc(myadd, 2, 1)

print("user defined function ",myadd([10, 20, 30, 40], [50, 60, 70, 80]))
```

```
zip of array x and y  [5, 7, 9, 11]
addition of element of the 2 arrays [ 66  88 110 132]
user defined function  [60 80 100 120]
```

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

In [ ]:
```
import numpy as np
from functools import reduce

# Create an array of elements
array = np.array([12, 18, 24, 36, 48])

# Find the LCM using NumPy's lcm.reduce() function
lcm_result = np.lcm.reduce(array)

# Find the GCD using NumPy's gcd.reduce() function
gcd_result = np.gcd.reduce(array)

print("Array:", array)
print("LCM:", lcm_result)
print("GCD:", gcd_result)
```

```
Array: [12 18 24 36 48]
LCM: 144
GCD: 6
```