EXPERIMENT NO. 10

Roll No. 2049

Q. Programs on Universal Function(ufunc)

1. ADD

Input:

```
import numpy as np arr1 =
np.array([30,54,34,12,65,23]) arr2 =
np.array([5,78,43,86,16,93]) newarr =
np.add(arr1, arr2)
print(newarr)
```

Output:

```
[ 35 132  77  98  81 116]
```

2. SUBTRACT

Input:

```
import numpy as np arr1 =
np.array([30,54,34,12,65,23]) arr2 =
np.array([5,78,43,86,16,93]) newarr =
np.subtract(arr1, arr2) print(newarr)
```

Output:

```
[ 25 -24  -9 -74  49 -70]
```

3. MULTIPLY

Input:

```
import numpy as np arr1 =
np.array([30,54,34,12,65,23]) arr2 =
np.array([5,78,43,86,16,93]) newarr =
np.multiply(arr1, arr2) print(newarr)
```

Output:

```
[ 150 4212 1462 1032 1040 2139]
```

4. DIVIDE

Input:

```
import numpy as np arr1 =
np.array([30,54,34,12,65,23])
arr2 =
np.array([4,78,43,86,16,93])
newarr = np.divide(arr1, arr2)
print(newarr)
```

Output:

```
[7.5        0.69230769 0.79069767 0.13953488 4.0625     0.24731183]
```

5. ROUNDING DECIMAL

Truncation function-

Input:

```
import numpy as np arr =
np.trunc([-7.8344, 9.2955])
print(arr)
```

Output:

```
[-7.  9.]
```

Round function-

Input:

```
import numpy as np arr =
np.around([-7.8344, 9.2955])
print(arr)
```

Output:

```
[-8.  9.]
```

6. LOGS

```
print(newarr)
```

At base 2-

Input:

```
import numpy as np

arr = np.arange(1, 10)
print(np.log2(arr))
```

Output:

```
[0.         1.         1.5849625
 2.80735492 3.         3.169925  ]
```

At any Base-

Input:

```
from math import log
import numpy as np
nplog = np.frompyfunc(log, 2, 1)
print(nplog(50, 5))
```

Output:

```
2.4306765580733933
```

7. SUMMATION

```
import numpy as np
arr1 = np.array([6, 10, 16])
arr2 = np.array([9, 20, 29])
newarr = np.sum([arr1, arr2])
print(newarr)
```

Output:

```
90
```

8. PRODUCT

Output:

```
5011200
```

9. DIFFERENCE

Input:

```
import numpy as np
arr = np.array([50,39,19,8,1])
newarr = np.diff(arr)
print(newarr)
```

Output:

```
[-11 -20 -11  -7]
```

10. LCM

Input:

```
import numpy as np
arr = np.array([4,8,16])
x = np.lcm.reduce(arr)
print(x)
```

Output:

```
16
```

11. GCD

Input:

```
import numpy as np
arr = np.array([40,28,36,74,90])
x = np.gcd.reduce(arr)
print(x)
```

Input:

```
2
```

```
import numpy as np
arr1 = np.array([6, 10, 16])
arr2 = np.array([9, 20, 29])
newarr = np.prod([arr1, arr2])
```

12. TRIGNOMETRY INTERSECTION-

Input:

```
import numpy as np arr =
np.array([np.pi/2, np.pi/3,
np.pi/4, np.pi/5]) x =
np.sin(arr) print(x)
```

Output:

```
[1.         0.8660254  0.70710678 0.5
```

Input:

```
import numpy as np arr1 =
np.array([29,43,15,75,34]) arr2 =
np.array([65,23,87,15,26]) newarr
= np.intersect1d(arr1, arr2,
assume_unique=True) print(newarr)
```

Output:

```
[15]
```

13. HYPERBOLIC FUNCTION

Input:

```
import numpy as np x = np.sinh(np.pi/2)
print(x)
```

Output:

```
2.3012989023072947
```

14. SET OPERATIONS

UNION-

Input:

```
import numpy as np arr1 =
np.array([29,43,15,75,34]) arr2
= np.array([65,23,87,15,26])
newarr = np.union1d(arr1, arr2)
print(newarr)
```

Output:

```
[15 23 26 29 34 43 65 75 87]
```