# Devyani Beohar

## J068

### ASSIGNMENT 2: NUMPY BASICS

In [1]:
```
import numpy as np
import time
```

In [2]:
```
A=np.array([[0,1,2],[3,4,5],[6,7,8]])
A
```

Out[2]:
```
array([[0, 1, 2],
       [3, 4, 5],
       [6, 7, 8]])
```

In [3]:
```
B=np.array([[10,11,12],[13,14,15],[16,17,18]])
B
```

Out[3]:
```
array([[10, 11, 12],
       [13, 14, 15],
       [16, 17, 18]])
```

In [4]:
```
C=np.array([[20,21,22],[23,24,25],[26,27,28]])
C
```

Out[4]:
```
array([[20, 21, 22],
       [23, 24, 25],
       [26, 27, 28]])
```

## AB!=BA

In [5]:
```
np.dot(A,B)
```

Out[5]:
```
array([[ 45,  48,  51],
       [162, 174, 186],
       [279, 300, 321]])
```

In [6]:
```
np.dot(B,A)
```

Out[6]:
```
array([[105, 138, 171],
       [132, 174, 216],
       [159, 210, 261]])
```

## A(BC)=(AB)C

In [7]:
```
A.dot(B.dot(C))
```

Out[7]:
```
array([[ 3330,  3474,  3618],
       [12078, 12600, 13122],
       [20826, 21726, 22626]])
```

In [8]:
```
(A.dot(B)).dot(C)
```

Out[8]:
```
array([[ 3330,  3474,  3618],
       [12078, 12600, 13122],
       [20826, 21726, 22626]])
```

## A(B+C)=AB + AC

In [10]:
```
d1=np.dot(A,(B+C))
d1
```

Out[10]:
```
array([[120, 126, 132],
       [444, 468, 492],
       [768, 810, 852]])
```

In [11]:
```
d2=np.dot(A,B) + np.dot(A,C)
d2
```

Out[11]:
```
array([[120, 126, 132],
       [444, 468, 492],
       [768, 810, 852]])
```

## AI=IA

In [12]:
```
I=np.identity(3)
I
```

Out[12]:
```
array([[1., 0., 0.],
       [0., 1., 0.],
       [0., 0., 1.]])
```

In [14]:
```
AI=np.dot(A,I)
IA=np.dot(I,A)
AI
```

Out[14]:
```
array([[0., 1., 2.],
       [3., 4., 5.],
       [6., 7., 8.]])
```

In [15]:
```
IA
```

Out[15]:
```
array([[0., 1., 2.],
       [3., 4., 5.],
       [6., 7., 8.]])
```

## INVERSE MATRIX

In [17]:
```
C_inv=np.linalg.inv(C)
C_inv
```

Out[17]:
```
array([[-3.46430741e+14,  6.92861481e+14, -3.46430741e+14],
       [ 6.92861481e+14, -1.38572296e+15,  6.92861481e+14],
       [-3.46430741e+14,  6.92861481e+14, -3.46430741e+14]])
```

In [22]:
```
mat=np.random.randint(100,500,(5,5))
mat
```

Out[22]:
```
array([[444, 261, 267, 496, 423],
       [315, 284, 326, 149, 273],
       [259, 297, 374, 372, 430],
       [340, 126, 283, 234, 408],
       [118, 424, 450, 493, 201]])
```

## COMPARING LOOPS AND INBUILT FUNCTION

In [29]:
```
import timeit
import numpy as np
a = list(range(100))
b = np.array(range(100))
t1 = timeit.timeit("[i + 1 for i in a]", setup="from __main__ import a", number=1000000)
print(t1)
```

```
7.322429900000316
```

In [30]:
```
t2 = timeit.timeit("np.add([1], b)", setup="from __main__ import np, b", number=1000000)
print(t2)
```

```
3.5826240999999754
```

In [ ]:
```
#THIS SHOWS INBUILT FUNCTION IS FASTER
```