

# LAB 02: Types and Objects

CS211 – Data Structures and Algorithms

Usman Institute of Technology

Fall 2020

- **How to submit:**

- Online: Submit on your respective MS Team.

1. Create a parent class **Array** which takes two parameters to initialize: **rows and cols** and write functions in Python whose parameters and return value are given below.

- a) Add a constructor of the class that initializes a list containing rows \* cols elements. All elements must be declared 0 by default.

```
class Array:
    def __init__(self, rows, cols):
        // your code goes here
```

Example:

```
A = Array(3,3) # for a 3 x 3 array
```

- b) Add a function **SetValue** which takes three parameters i, j, and v, for row, column, and value respectively. The function set the value at i<sup>th</sup> row and j<sup>th</sup> column.

The function is supposed to convert these two dimensions value into a linear dimension.

The following equation can be used for conversion:

$\text{Location} = i * R + j$	(R = number of elements in a ROW)
-------------------------------	-----------------------------------

```
def SetValue(self, i, j, v):
    // your code goes here
```

Example:

```
A = Array(3,3)
A.SetValue(0,0,5)
A.SetValue(0,1,15)
```

- c) Add a function **GetValue** which takes two parameters i and j and returns the value for i<sup>th</sup> row and j<sup>th</sup> column. You have to convert two dimensional values into a single dimension value.

```
def GetValue(self, i, j):
    // your code goes here
```

Example:

```
A = Array(3,3)
A.SetValue(0,0,5)
A.SetValue(0,1,15)
print(A.GetValue(0,0))
print(A.GetValue(0,1))
```

- d) Add a function **PrintValues()** that print the values of the array in Row and Column format.

```
def PrintValues(self):
    // your code goes here
Example:
A = Array(3,3)
A.SetValue(0,0,5)
A.SetValue(0,1,15)
A.PrintValues()
```

- e) Add a function **SubValues()** that takes two parameters array1 and array2 and returns a array object containing difference of two given matrices.

```
def SubValues(array1, array2):
    // your code goes here
Example:
a1 = Array(2,2)
a2 = Array(2,2)
a3 = a1.SubValues(a1,a2)
```

- f) Add a function **MultValues()** that takes two parameters Matrix A and Matrix B and returns a matrix containing multiplication of two given matrices.

```
def MultValues(array1, array2):
    // your code goes here
Example:
a1 = Array(2,2)
a2 = Array(2,2)
a3 = a1.MultValues(a1,a2)
```

- g) Add a function **Transpose()** that returns a matrix containing the transpose of the matrix.

```
def Transpose():
    // your code goes here
Example:
a1 = Array(2,2)
a2 = a1.Transpose()
```

Example: Matrix A =  $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$

Transpose =  $\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$

## 2. Create a Python script by using the following functions by importing NumPy Library of Python.

### 1. Create a Numpy Array.

```
import numpy as np

array1 = np.array([[1,2,3,4],[5,6,7,8]], dtype=np.int64)
print(array1)
```

### 2. Create an array of ones

```
x = np.ones((3,4), dtype=np.int64)
print(x)
```

### 3. Create an array of zeros

```
y = np.zeros((2,3,4), dtype=np.int16)
print(y)
```

### 4. Create an array with random values

```
array2 = np.random.random((2,2))
print(array2)
```

### 5. Create a full array

```
array3 = np.full((3,3), 7)
print(array3)
```

### 6. Create an identity matrix

```
array4 = np.identity(3, dtype=np.int64)
print(array4)
```

### 7. Find sum of two matrices

```
add = np.add(x, y)
```

```
print(add)
```

#### 8. Find difference of two matrices

```
diff = np.subtract(x,y)  
print(diff)
```

#### 9. Find product of two matrices

```
mult = np.multiply(x,y)  
print(mult)
```

#### 10. Find division of two matrices

```
div = np.divide(y,x)  
print(div)
```

#### 11. Find remainder of two matrices

```
rem = np.remainder(y,x)  
print(rem)
```

#### 12. Check if two arrays are equal

```
result = np.array_equal(x,y)  
print(result)
```