



Submission-2



COLLEGE OF ENGINEERING
(AUTONOMOUS)

Name: Bongu Hari Hara Charan

Roll no: 323103310032

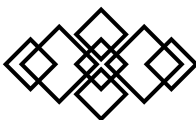
Branch: CSE-1

Semistor: 3

Name of the lab: Adv Python lab

Professor: Satya Keerthi Gorripati

Links:



Week3

Numpy Library:

```
week3a.py X
2nd year Labs > python > Week3 > week3a.py > ...
1 import numpy as np
2
3 l = [1, 2, 3, 4, 5]
4 # Converting list to numpy array and making the data type as float16
5 arr1 = np.array(l, dtype='float16')
6
7 # Converting tuple to numpy array and making the data type as float
8 t = (6, 7, 8, 9, 10)
9 arr2 = np.array(t, dtype=float)
10
11 print(arr1)
12 print(arr2)
13
14 arr3 = np.array([2, 4, 5])
15 print(arr3)
16
17 print(arr1.dtype)
18 print(arr2.dtype) # Default data type is float64
19 print(arr1.size)
20 print(arr1.itemsize)
```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week3> python week3a.py

```
[1.  2.  3.  4.  5.]
[ 6.  7.  8.  9. 10.]
[2 4 5]
float16
float64
5
2
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week3> 
```

```
week3b.py X
2nd year Labs > python > Week3 > week3b.py > ...
1 import numpy as np
2
3 arr2 = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
4 print("****Slicing****")
5 print("Array: ", arr2)
6 print("First row: ", arr2[0])
7 print("First column: ", arr2[:, 0])
8 print("Last column: ", arr2[:, -1])
9 print("****Integer array indexing****")
10 rows = [0, 2]
11 cols = [1, 2]
12 int_indexing = arr2[rows, cols]
13 print("Integer Indexing: ", int_indexing)
14 print("****Boolean array indexing****")
15 bool_indexing = arr2[arr2 > 5]
16 print("Boolean Indexing: ", bool_indexing)
17
```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week3> python week3b.py

```
****Slicing****
Array: [[1 2 3]
 [4 5 6]
 [7 8 9]]
First row: [1 2 3]
First column: [1 4 7]
Last column: [3 6 9]
****Integer array indexing****
Integer Indexing: [2 9]
****Boolean array indexing****
Boolean Indexing: [6 7 8 9]
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week3> 
```

```

week3c.py X
2nd year Labs > python > Week3 > week3c.py ...
1 import numpy as np
2
3 arr2 = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
4
5 print("The array is: ", arr2)
6 print("Maximum element in the array: ", np.max(arr2))
7 print("Minimum element in the array: ", np.min(arr2))
8 print("Maximum element in all rows: ", np.max(arr2, axis=1)) # axis=1 for rows
9 print("Minimum element in all columns: ", np.min(arr2, axis=0)) # axis=0 for columns
10 print("Sum of all elements in the array: ", np.sum(arr2))
11 print("Sum of all elements in each column: ", np.sum(arr2, axis=0))
12 print("Sum of all elements in each row: ", np.sum(arr2, axis=1))
13 print("Cumulative sum of all elements in the array: ", np.cumsum(arr2))
14 print("Cumulative sum of all elements in each column: ", np.cumsum(arr2, axis=0))
15 print("Cumulative sum of all elements in each row: ", np.cumsum(arr2, axis=1))

```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week3> python week3c.py

The array is: [[1 2 3]
[4 5 6]
[7 8 9]]

Maximum element in the array: 9
Minimum element in the array: 1
Maximum element in all rows: [3 6 9]
Minimum element in all columns: [1 2 3]
Sum of all elements in the array: 45
Sum of all elements in each column: [12 15 18]
Sum of all elements in each row: [6 15 24]
Cumulative sum of all elements in the array: [1 3 6 10 15 21 28 36 45]
Cumulative sum of all elements in each column: [[1 2 3]
[5 7 9]
[12 15 18]]
Cumulative sum of all elements in each row: [[1 3 6]
[4 9 15]
[7 15 24]]

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week3> |

```

week3d.py X
2nd year Labs > python > Week3 > week3d.py ...
1 import numpy as np
2
3 arr = np.array([[1, 2, 3], [4, 5, 6]])
4 print("The array is: ", arr)
5 print("The dimensions of the array is: ", arr.ndim)
6
7 print("*** Changing dimension type of the array ***")
8 arr = np.array([[1, 2, 3], [4, 5, 6]], ndmin=5)
9 print("The dimensions of the array is: ", arr.ndim)
10 print("The array is: ", arr)
11 print("The shape of the array is: ", arr.shape)
12
13 print("*** Changing the shape of the array ***")
14 arr = arr.reshape(3, 2)
15 print(arr)
16 print("The shape of the array is: ", arr.shape)
17 print("The size of the array is: ", arr.size)
18 print("The data type of the array is: ", arr.dtype)
19
20 print("*** Changing the data type of the array ***")
21 arr = arr.astype(float)
22 print(arr)
23 print("The data type of the array is: ", arr.dtype)
24 print("The item size of the array is: ", arr.itemsize)

```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week3> python week3d.py

The array is: [[1 2 3]
[4 5 6]]

The dimensions of the array is: 2
*** Changing dimension type of the array ***
The dimensions of the array is: 5
The array is: [[[[[1 2 3]
[4 5 6]]]]]]

The shape of the array is: (1, 1, 1, 2, 3)
*** Changing the shape of the array ***
[[1 2]
[3 4]
[5 6]]

The shape of the array is: (3, 2)
The size of the array is: 6
The data type of the array is: int64
*** Changing the data type of the array ***
[[1. 2.]
[3. 4.]
[5. 6.]]

The data type of the array is: float64
The item size of the array is: 8

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week3> |

Week4

Numpy Library(Linear Algebra):

```
week4a.py X
2nd year Labs > python > Week4 > week4a.py > ...
1 import numpy as np
2
3 arr = np.array([[1, 2, 3], [4, 5, 6], [7, 10, 9]])
4 print("The matrix is:", arr)
5 print("The rank of the matrix is", np.linalg.matrix_rank(arr))
6 print("The trace of the matrix is", np.trace(arr))
7 print("The determinant of the matrix is", np.linalg.det(arr))
```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week4> python week4a.py
The matrix is: [[ 1  2  3]
 [ 4  5  6]
 [ 7 10  9]]
The rank of the matrix is 3
The trace of the matrix is 15
The determinant of the matrix is 11.999999999999999
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week4>
```

```
week4b.py X
2nd year Labs > python > Week4 > week4b.py > ...
1 import numpy as np
2 arr = np.array([[1, 2, 3],[2, 3, 4],[4, 5, 6]])
3 print("The matrix is: ", arr)
4 a, b = np.linalg.eig(arr)
5 print("The eigenvalues of the matrix are ", a)
6 arr = np.array([[1, 2, 3], [2, 3, 4], [4, 5, 6]])
7 print("The matrix is: ", arr)
8
9 a, b = np.linalg.eig(arr)
10 print("The eigenvalues of the matrix are ", a)
11 print("The eigenvectors of the matrix are ", b)
```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week4> python week4b.py
The matrix is: [[1 2 3]
 [2 3 4]
 [4 5 6]]
The eigenvalues of the matrix are [ 1.08309519e+01 -8.30951895e-01  1.01486082e-16]
The matrix is: [[1 2 3]
 [2 3 4]
 [4 5 6]]
The eigenvalues of the matrix are [ 1.08309519e+01 -8.30951895e-01  1.01486082e-16]
The eigenvectors of the matrix are [[ 0.34416959  0.72770285  0.40824829]
 [ 0.49532111  0.27580256 -0.81649658]
 [ 0.79762415 -0.62799801  0.40824829]]
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week4>
```

```

week4c.py X
2nd year Labs > python > Week4 > week4c.py > ...
1 import numpy as np
2
3 vector_a = np.array([1, 2, 3])
4 vector_b = np.array([4, 5, 6])
5 matrix_a = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
6 matrix_b = np.array([[9, 8, 7], [6, 5, 4], [3, 2, 1]])
7
8 print("Vector a:", vector_a)
9 print("Vector b:", vector_b)
10 print("Dot product of vector a and b:", np.dot(vector_a, vector_b)) # 1*4 + 2*5 + 3*6
11 print("\nMatrix a:\n", matrix_a)
12 print("Matrix b:\n", matrix_b)
13 print("Dot product of matrix a and b:\n", np.dot(matrix_a, matrix_b)) # matrix multiplication
14 print("\nInner product of vector a and b:", np.inner(vector_a, vector_b))
15 print("Inner product of matrix a and b:\n", np.inner(matrix_a, matrix_b))
16 print("\nOuter product of vector a and b:\n", np.outer(vector_a, vector_b))
17 print("Outer product of matrix a and b:\n", np.outer(matrix_a, matrix_b))
18 print("\nMatrix a raised to the power 2:\n", np.linalg.matrix_power(matrix_a, 2))

```

```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week4> python week4c.py
Vector a: [1 2 3]
Vector b: [4 5 6]
Dot product of vector a and b: 32

Matrix a:
[[1 2 3]
 [4 5 6]
 [7 8 9]]
Matrix b:
[[9 8 7]
 [6 5 4]
 [3 2 1]]
Dot product of matrix a and b:
[[ 30 24 18]
 [ 84 69 54]
 [138 114 90]]

Inner product of vector a and b: 32
Inner product of matrix a and b:

```

Output:

```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
Vector a: [1 2 3]
Vector b: [4 5 6]
Dot product of vector a and b: 32

Matrix a:
[[1 2 3]
 [4 5 6]
 [7 8 9]]
Matrix b:
[[9 8 7]
 [6 5 4]
 [3 2 1]]
Dot product of matrix a and b:
[[ 30 24 18]
 [ 84 69 54]
 [138 114 90]]

Inner product of vector a and b: 32
Inner product of matrix a and b:
[[ 46 28 10]
 [118 73 28]
 [190 118 46]]

Outer product of vector a and b:
[[ 4 5 6]
 [ 8 10 12]
 [12 15 18]]
Outer product of matrix a and b:
[[ 9 8 7 6 5 4 3 2 1]
 [18 16 14 12 10 8 6 4 2]
 [27 24 21 18 15 12 9 6 3]
 [36 32 28 24 20 16 12 8 4]
 [45 40 35 30 25 20 15 10 5]
 [54 48 42 36 30 24 18 12 6]
 [63 56 49 42 35 28 21 14 7]
 [72 64 56 48 40 32 24 16 8]
 [81 72 63 54 45 36 27 18 9]]

Matrix a raised to the power 2:
[[ 30 36 42]
 [ 66 81 96]
 [102 126 150]]
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week4>

```

```

week4d.py X
2nd year Labs > python > Week4 > week4d.py > ...
1 import numpy as np
2
3 # Example system of equations:
4 # 2x + 3y = 8
5 # 3x + 4y = 11
6
7 # Coefficient matrix (A)
8 A = np.array([[2, 3],
9               [3, 4]])
10
11 # Result matrix (B)
12 B = np.array([8, 11])
13
14 # Solve the system of equations
15 solution = np.linalg.solve(A, B)
16
17 # Print the solution (values of x and y)
18 print("Solution: ", solution)
19
20 #Example system of Matrix
21
22 # Define the coefficient matrix A
23 A = np.array([[3, 1, -1],
24               [2, 4, 1],
25               [-1, 2, 5]])
26
27 # Define the result matrix B
28 B = np.array([4, 1, 1])
29
30 # Solve the system Ax = B
31 x = np.linalg.solve(A, B)
32
33 # Print the solution vector x
34 print("Solution for x is: ", x)

```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week4> python week4d.py
Solution: [1. 2.]
Solution for x is: [ 2. -1.  1.]
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week4>

```

Week5

Numpy Advanced:

```

week5a.py X
2nd year Labs > python > Week5 > week5a.py > ...
1 import numpy as np
2 from PIL import Image
3
4 # Create a 500x500 white image using numpy
5 white_array = np.ones((500, 500, 3), dtype=np.uint8) * 255
6
7 # Convert the numpy array to a PIL image
8 white_image = Image.fromarray(white_array)
9
10 # Save the white image to a file
11 white_image.save("white.png")
12
13 # Create a 500x500 black image with the red channel set to 255
14 red_array = np.zeros((500, 500, 3), dtype=np.uint8)
15 red_array[:, :, 0] = 255 # Set the red channel to 255
16
17 # Convert the numpy array to a PIL image
18 red_image = Image.fromarray(red_array)
19
20 # Save the red image to a file and display it
21 red_image.save("red.png")
22 red_image.show()
23

```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week5> python week5a.py
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week5>

```

white.png X

2nd year Labs > python > Week5 > white.png

white.png

red.png X

2nd year Labs > python > Week5 > red.png


```

week5cpy X
2nd year Labs > python > Week5 > week5cpy > ...
1 import numpy as np
2
3 # Sorting
4 a = np.array([[12, 5], [10, 35]])
5 print("Along y-axis:", np.sort(a, axis=0))
6 print("Along x-axis:", np.sort(a, axis=1))
7 print("Flattened array:", np.sort(a, axis=None))
8
9 # Searching
10 arr = np.array([1, 2, 3, 3, 4, 5, 6, 4, 5])
11 x = np.where(arr == 4)
12 print("Index where 4 is present:", x)
13 arr1 = np.array([1, 2, 3, 4, 6, 7, 8, 9, 10])
14 y = np.searchsorted(arr1, 4)
15 print("Index where 4 should be inserted:", y)
16
17 # Counting
18 arr2 = np.array([[1, 2, 3, 4, 5], [6, 7, 8, 9, 10], [1, 2, 3, 4, 5], [6, 7, 8, 9, 10]])
19 count_4 = np.count_nonzero(arr2 == 4)
20 print("Number of times 4 appears in arr2:", count_4)
21 unique, counts = np.unique(arr2, return_counts=True)
22 count_dict = {int(k): int(v) for k, v in zip(unique, counts)}
23 print("Count of each unique element in arr2:", count_dict)
24 unique1 = np.unique(arr2)
25 print("Unique elements are:", unique1)
26 count1 = np.unique(arr2, return_counts=True)
27 print(count1)

```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week5> python week5cpy
Along y-axis: [[10  5]
 [12 35]]
Along x-axis: [[ 5 12]
 [10 35]]
Flattened array: [ 5 10 12 35]
Index where 4 is present: (array([4, 7]),)
Index where 4 should be inserted: 3
Number of times 4 appears in arr2: 2
Count of each unique element in arr2: {1: 2, 2: 2, 3: 2, 4: 2, 5: 2, 6: 2, 7: 2, 8: 2, 9: 2, 10: 2}
Unique elements are: [ 1  2  3  4  5  6  7  8  9 10]
(array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10]), array([2, 2, 2, 2, 2, 2, 2, 2, 2, 2]))
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week5>

```

```

week5d.py X
2nd year Labs > python > Week5 > week5d.py > ...
1 import numpy as np
2
3 # Python program to demonstrate the use of reshape() function in numpy
4 a = np.array([[1, 2, 3, 4, 5], [6, 7, 8, 9, 10]])
5 print("Original array:", a)
6 print("Shape of the array:", a.shape)
7
8 newa = a.reshape(5, 2)
9 print("Reshaped array:", newa)
10 print("Shape of the reshaped array:", newa.shape)

```

PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```

PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week5> python week5d.py
Original array: [[ 1  2  3  4  5]
 [ 6  7  8  9 10]]
Shape of the array: (2, 5)
Reshaped array: [[ 1  2]
 [ 3  4]
 [ 5  6]
 [ 7  8]
 [ 9 10]]
Shape of the reshaped array: (5, 2)
PS C:\Users\bhhc1\Desktop\Hari Hara Charan\323103310032\2nd year Labs\python\Week5>

```