

GITHUB LINK-

<https://github.com/Harishwar-reddi/ICP-3>

YOUTUBE LINK-

https://youtu.be/PD_ul4x010o

Q1

```
import numpy as np

# 1. Using NumPy create random vector of size 15 having only Integers in the range 1-20.
vector = np.random.randint(1, 21, size=15)

# a. Reshape the array to 3 by 5.
array_3x5 = vector.reshape(3, 5)

# b. Print array shape.
print("Shape of the array:", array_3x5.shape)

# c. Replace the max in each row by 0.
for i in range(array_3x5.shape[0]):
    max_val_index = array_3x5[i].argmax()
    array_3x5[i][max_val_index] = 0

print("Modified Array:")
print(array_3x5)
```

```
Shape of the array: (3, 5)
Modified Array:
[[ 9  3  7  0 16]
 [ 3  0  5  6 11]
 [ 0 12  1 11  1]]
```

Q2

```
[9] import numpy as np
array = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9],
                  [10, 11, 12]], dtype=np.int32)

# Print array shape
print("Shape of the array:", array.shape)

# Print type of the array
print("Type of the array:", type(array))

# Print data type of the array
print("Data type of the array:", array.dtype)
```

```
Shape of the array: (4, 3)
Type of the array: <class 'numpy.ndarray'>
Data type of the array: int32
```

Q3

```
[4] import numpy as np

array = np.array([[0, 1, 2], [3, 4, 5]])

# sum of the diagonal elements
diagonal_sum = np.sum(array.diagonal())

print("Sum of diagonal elements:", diagonal_sum)
```

Sum of diagonal elements: 4

Q4

```

import numpy as np

# a. To create an array of odd and even numbers between 10 to 70.
evens = np.arange(10, 71, 2)
odds = np.arange(11, 71, 2)
print("Even numbers:", evens)
print("Odd numbers:", odds)

# b. To perform at-least three element-wise mathematical operations using two arrays of the same size.
arr1 = np.array([1, 2, 3])
arr2 = np.array([4, 5, 6])

sum_arr = np.add(arr1, arr2)
diff_arr = np.subtract(arr1, arr2)
product_arr = np.multiply(arr1, arr2)

print("Sum of arrays:", sum_arr)
print("Difference of arrays:", diff_arr)
print("Product of arrays:", product_arr)

# c. To sort a given array by row and column in ascending order.
array = np.array([[5.54, 3.38, 7.99], [3.54, 4.38, 6.99], [1.54, 2.39, 9.29]])

sorted_by_row = np.sort(array, axis=1)
sorted_by_column = np.sort(array, axis=0)

print("Sorted by row:")
print(sorted_by_row)
print("Sorted by column:")
print(sorted_by_column)

```

```

Even numbers: [10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56
58 60 62 64 66 68 70]
Odd numbers: [11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49 51 53 55 57
59 61 63 65 67 69]
Sum of arrays: [5 7 9]
Difference of arrays: [-3 -3 -3]
Product of arrays: [ 4 10 18]
Sorted by row:
[[3.38 5.54 7.99]
 [3.54 4.38 6.99]
 [1.54 2.39 9.29]]
Sorted by column:
[[1.54 2.39 6.99]
 [3.54 3.38 7.99]
 [5.54 4.38 9.29]]

```

Q5

```
import numpy as np

# Declare the given array using NumPy
Array = np.array([[4, 2, np.nan, 1],
                  [11, 12, 14, 9],
                  [5, np.nan, 1, np.nan]])

# Find missing data by returning Boolean output
missing_data = np.isnan(Array)

print(missing_data)
```

```
[[False False  True False]
 [False False False False]
 [False  True False  True]]
```