

#Task 1:

*#Write a Python program to find the maximum and minimum value of a given
#flattened array.*

import numpy as np

arr1 = np.array([[0,1],[2,3]])

arr1

max1 = np.max(arr1)

min1 = np.min(arr1)

print('The maximum value of the above-flattened array:')

print(max1)

print('A minimum value of the above-flattened array:')

print(min1)

The maximum value of the above-flattened array:

3

A minimum value of the above-flattened array:

0

#Task 2

#Write a NumPy program to compute the median of flattened given array.

#Note: First array elements raised to powers from second array

arr2 = np.array([[0, 1, 2, 3, 4, 5],[6, 7, 8, 9, 10, 11]])

arr5= np.power(arr2[0],arr2[1])

arr2[0] = arr5

print (arr2)

arr = arr2.flatten()

print(arr)

median = np.median(arr)

print('Median of said array:')

print(median)

[[0 1 256 19683 1048576 48828125]

[6 7 8 9 10 11]]

[0 1 256 19683 1048576 48828125 6 7
 8 9 10 11]

Median of said array:

9.5

#Task 3

#Write a NumPy program to compute the mean, standard deviation, and variance
#of a given array along the second axis.

#Sample output:

#Original array:

#[0 1 2 3 4 5]

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

```
print(arr)
```

```
mean = np.mean(arr)
```

```
std_dev = np.std(arr)
```

```
variance = np.var(arr)
```

```
print("Mean: ",mean)
```

```
print("std: ",std_dev)
```

```
print("variance: ",variance)
```

```
[0 1 2 3 4 5]
```

```
Mean:  2.5
```

```
std:  1.707825127659933
```

```
variance:  2.9166666666666665
```

#Task 4

#Write a Python program to count number of occurrences of each value in a given
#array of non-negative integers.

#Note: bincount() function count number of occurrences of each value in an array
#of non-negative integers in the range of the array between the minimum and
#maximum values including the values that did not occur.

#Sample Output:

#Original array:

```
#[0, 1, 6, 1, 4, 1, 2, 2, 7]
```

#Number of occurrences of each value in array:

```
#[1 3 2 0 1 0 1 1]
```

#code bellow

```

arr = np.array([0, 1, 6, 1, 4, 1, 2, 2, 7])

occ = np.bincount(arr)
print("Number of occurrences of each value in array:")
print(occ)

Number of occurrences of each value in array:
[1 3 2 0 1 0 1 1]

#Write a NumPy program to compute the histogram of nums against the bins.
#Sample Output:
#nums: [0.5 0.7 1. 1.2 1.3 2.1]
#bins: [0 1 2 3]
#Result: (array([2, 3, 1], dtype=int64), array([0, 1, 2, 3]))

```

```

import matplotlib.pyplot as plt
nums = [0.5, 0.7, 1, 1.2, 1.3, 2.1]
bins= [0 ,1 , 2 , 3]
plt.figure(figsize=(6, 4))
x,y = np.histogram(nums,bins)
plt.hist(nums, bins, color='blue', alpha=0.5)

print('result :',(x ,y))

plt.show()

result : (array([2, 3, 1], dtype=int64), array([0, 1, 2, 3]))

result : (array([2, 3, 1], dtype=int64), array([0, 1, 2, 3]))

```

