Yog Chaudhary

11727095

ADTA 5550: Deep Learning with Big Data

Professor: Dr. Hamidreza Moradi

University of North Texas

Jan 29, 2024

In [1]:

```
print("Welcome ADTA 5550 Deep Learning with big data")
```

```
Welcome ADTA 5550 Deep Learning with big data
```

Q1. Create a vector (1D array) of size 20. All the elements are initialized with 0 (zero) except for the

8th element that is set with the value 8.

In [2]:

```
vector = [0 if i != 7 else 8 for i in range(20)]
print(vector)
```

```
[0, 0, 0, 0, 0, 0, 0, 8, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

Q2. Create a vector of size 16 with random values ranging from 0 to 63, print the vector, then sort it

and print the vector again

In [3]:

```
import random

# Create a vector of size 16 with random values
vector = [random.randint(0, 63) for _ in range(16)]

# Print the original vector
print("Original Vector:", vector)

# Sort the vector
vector.sort()

# Print the sorted vector
print("Sorted Vector:", vector)
```

```
Original Vector: [61, 37, 59, 20, 43, 47, 31, 8, 37, 44, 7, 10, 7, 19, 29, 31]
Sorted Vector: [7, 7, 8, 10, 19, 20, 29, 31, 31, 37, 37, 43, 44, 47, 59, 61]
```

Q3. Create a 5x5 matrix with values ranging from 0 to 24.

In [4]:

```
import numpy as np

# Create a 5x5 matrix with values ranging from 0 to 24
matrix = np.arange(25).reshape(5, 5)

# Print the matrix
print(matrix)
```

```
[[ 0  1  2  3  4]
 [ 5  6  7  8  9]
 [10 11 12 13 14]
 [15 16 17 18 19]
 [20 21 22 23 24]]
```

Q4. Create an 8x8 array with random values, then find the min and max values stored in this matrix

In [5]:

```
import numpy as np
```

In [6]:

```
random_array=np.random.rand(8,8)
```

In [7]:

```
min=np.min(random_array)
```

In [8]:

```
maxi=np.max(random_array)
```

In [9]:

```
print(random_array)
```

```
[[0.32430815 0.12924265 0.15902582 0.938683   0.61377919 0.26913415
  0.47337032 0.78330483]
 [0.10751246 0.82912372 0.14906958 0.02043026 0.3947379  0.18346553
  0.21672932 0.25627099]
 [0.61149468 0.20123634 0.2895593  0.7141168  0.64219482 0.59589227
  0.91231885 0.69853699]
 [0.1664031  0.11754229 0.68081051 0.04271628 0.66173316 0.5801633
  0.80056273 0.25591484]
 [0.82995601 0.04988948 0.87074648 0.71913119 0.00563231 0.9162301
  0.1086788  0.24499845]
 [0.10241213 0.32672512 0.26171743 0.81910861 0.51827273 0.38829735
  0.72801411 0.77348132]
 [0.16936178 0.24014503 0.4700479  0.6835266  0.79504418 0.05461901
  0.05113817 0.37529729]
 [0.93141782 0.88641289 0.23941205 0.63213365 0.00559098 0.91968707
  0.7851192  0.18853933]]
```

In [10]:

```
print(min)
```

```
0.005590981246440529
```

In [11]:
```
print(maxi)
```

```
0.9386830014748025
```

Q5, Create a vector of size 32 that is initialized with random values inside the range (0, 99) and then find the mean of all the initial values.

In [12]:
```
import numpy as np
```

In [14]:
```
ran_vector=np.random.randint(0,100,size=32)
```

In [19]:
```
mean_val=np.mean(ran_vector)
```

In [21]:
```
print("Random Vector")
```

```
Random Vector
```

In [22]:
```
print(ran_vector)
```

```
[ 8 75 24 60  3 54 55 35 10 77 74  7  0 85 35 25 21 40 68 26 24 32 48 51
 65 35 17 83 79  4 50 32]
```

In [23]:
```
print(mean_val)
```

```
40.6875
```