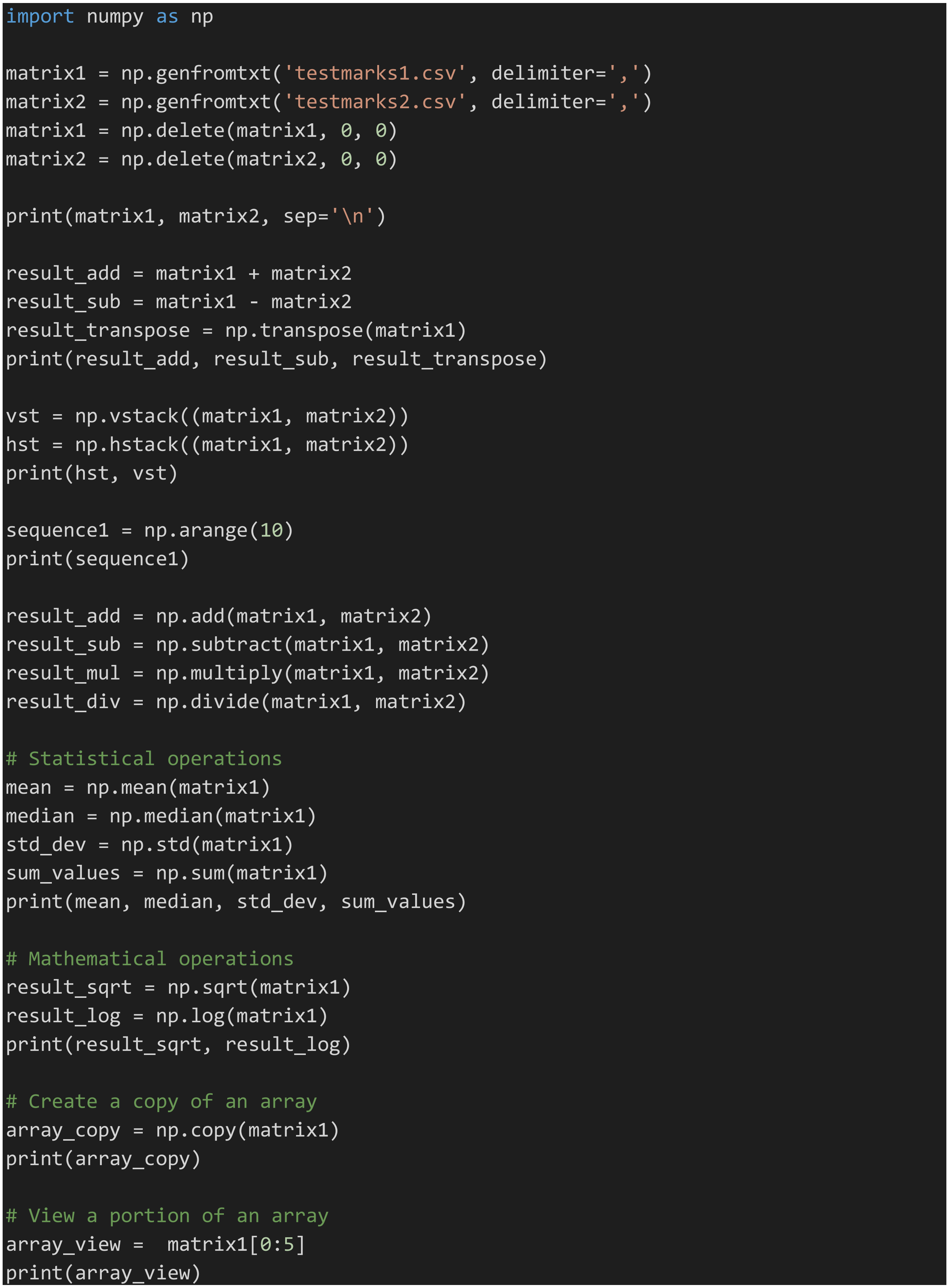
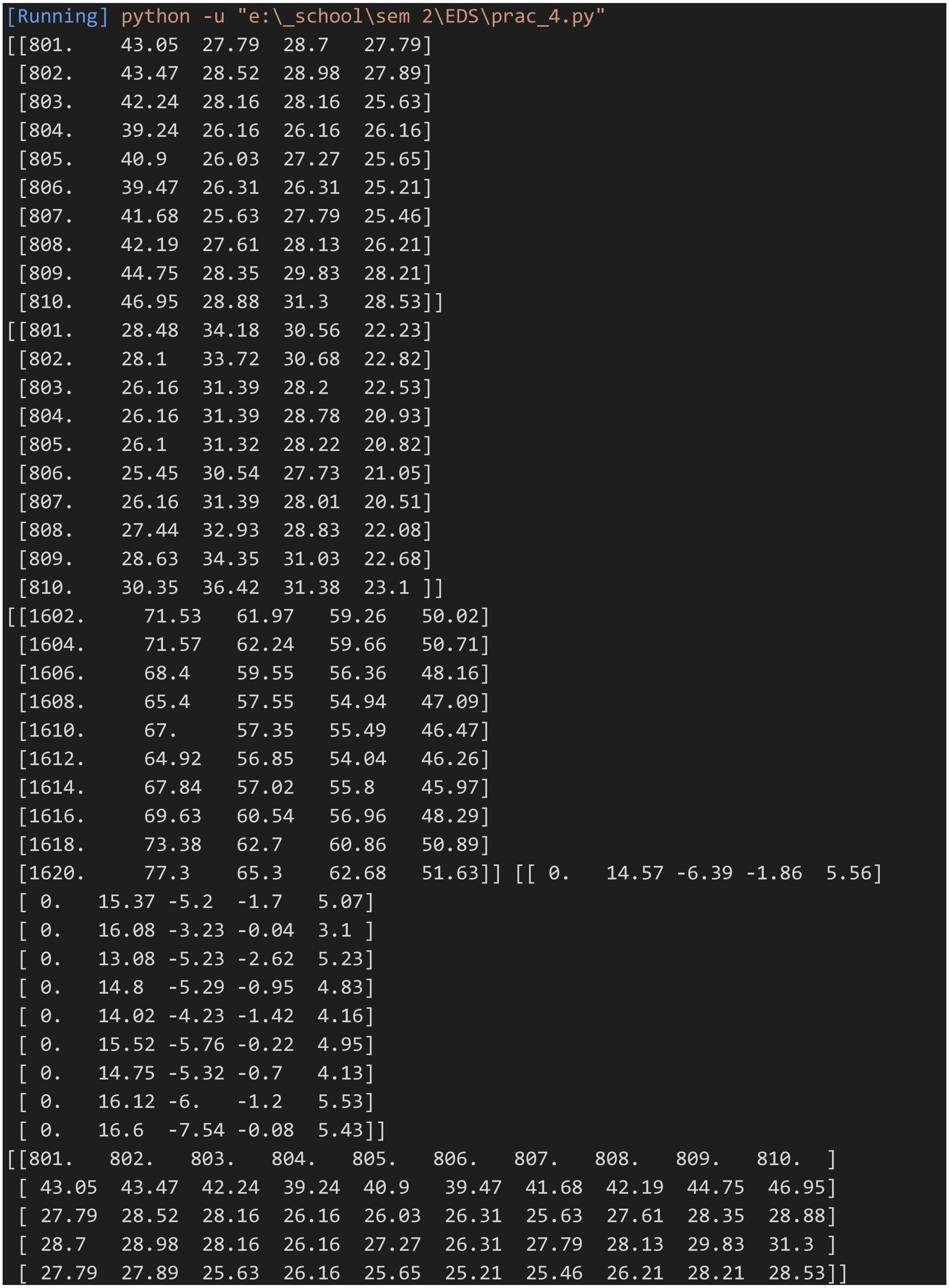
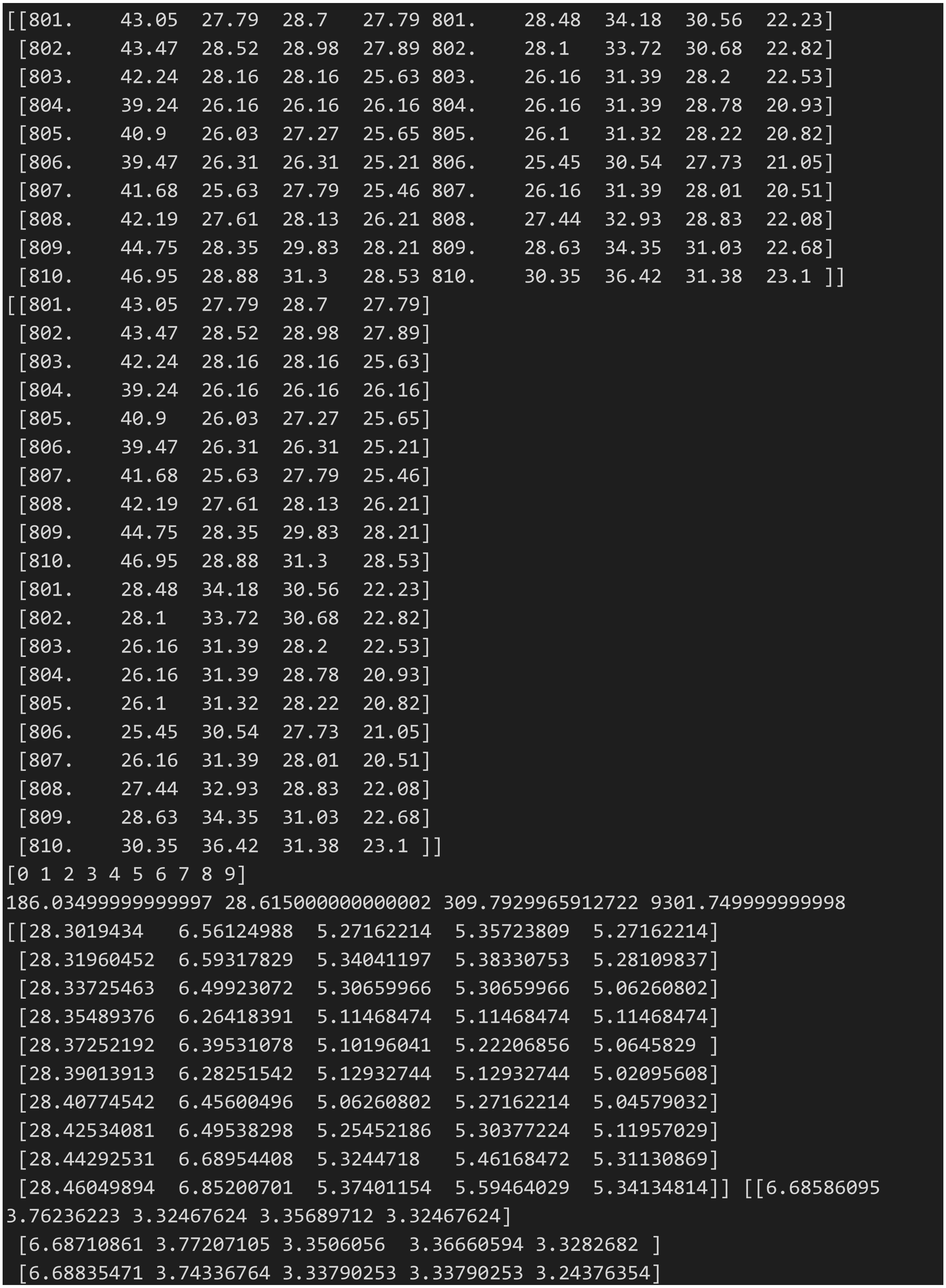


```python
import numpy as np

matrix1 = np.genfromtxt('testmarks1.csv', delimiter=',')
matrix2 = np.genfromtxt('testmarks2.csv', delimiter=',')
matrix1 = np.delete(matrix1, 0, 0)
matrix2 = np.delete(matrix2, 0, 0)

print(matrix1, matrix2, sep='\n')

result_add = matrix1 + matrix2
result_sub = matrix1 - matrix2
result_transpose = np.transpose(matrix1)
print(result_add, result_sub, result_transpose)

vst = np.vstack((matrix1, matrix2))
hst = np.hstack((matrix1, matrix2))
print(hst, vst)

sequence1 = np.arange(10)
print(sequence1)

result_add = np.add(matrix1, matrix2)
result_sub = np.subtract(matrix1, matrix2)
result_mul = np.multiply(matrix1, matrix2)
result_div = np.divide(matrix1, matrix2)

# Statistical operations
mean = np.mean(matrix1)
median = np.median(matrix1)
std_dev = np.std(matrix1)
sum_values = np.sum(matrix1)
print(mean, median, std_dev, sum_values)

# Mathematical operations
result_sqrt = np.sqrt(matrix1)
result_log = np.log(matrix1)
print(result_sqrt, result_log)

# Create a copy of an array
array_copy = np.copy(matrix1)
print(array_copy)

# View a portion of an array
array_view =  matrix1[0:5]
print(array_view)
```

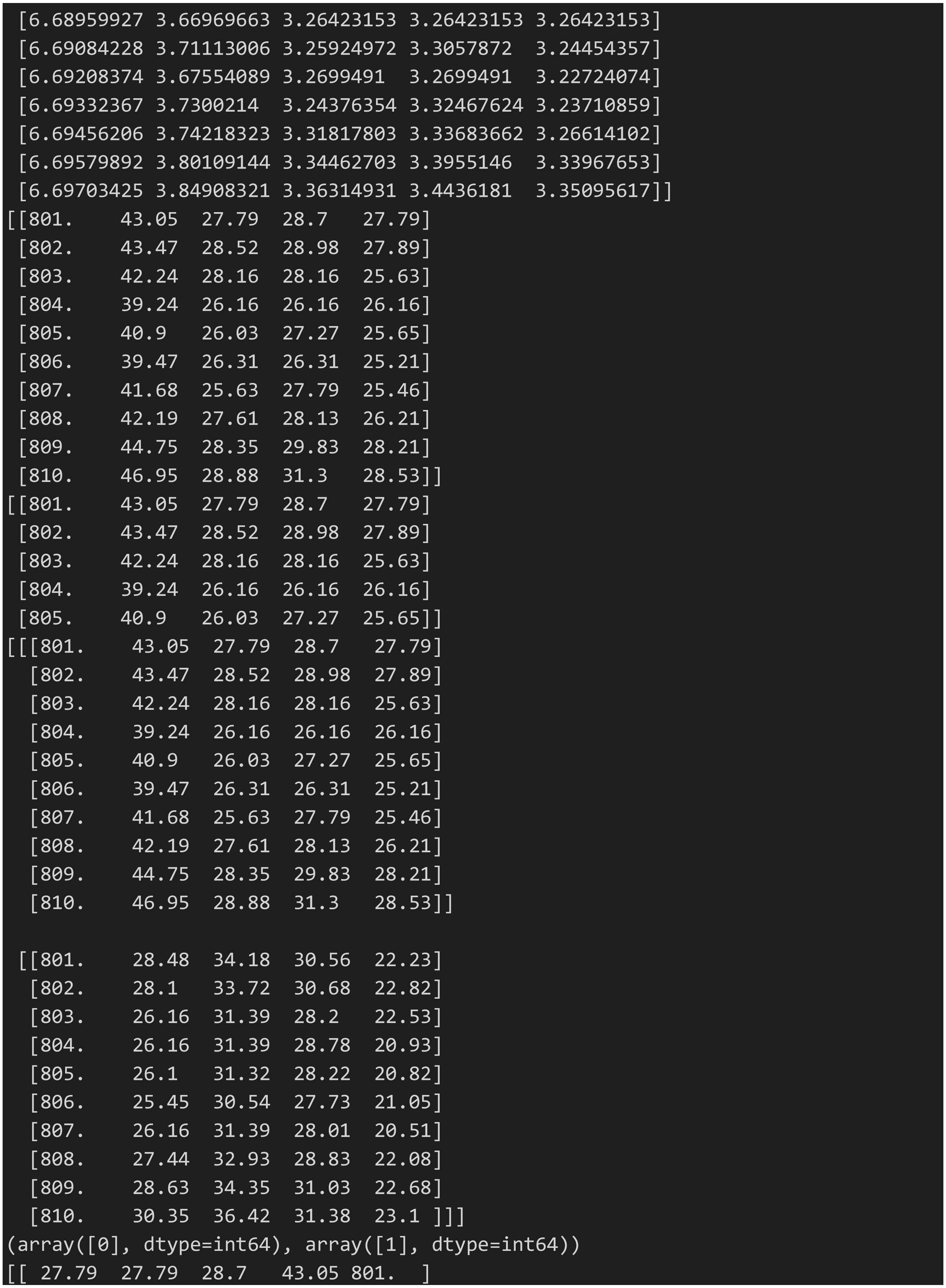
```
# Data stacking
stacked_array = np.stack((matrix1, matrix2))
print(stacked_array)
# Searching
indices = np.where(matrix1 == 43.05)
print(indices)

# Sorting
sorted_array = np.sort(matrix1)
print(sorted_array)

# Counting
count = np.count_nonzero(matrix1)
print(count)
```



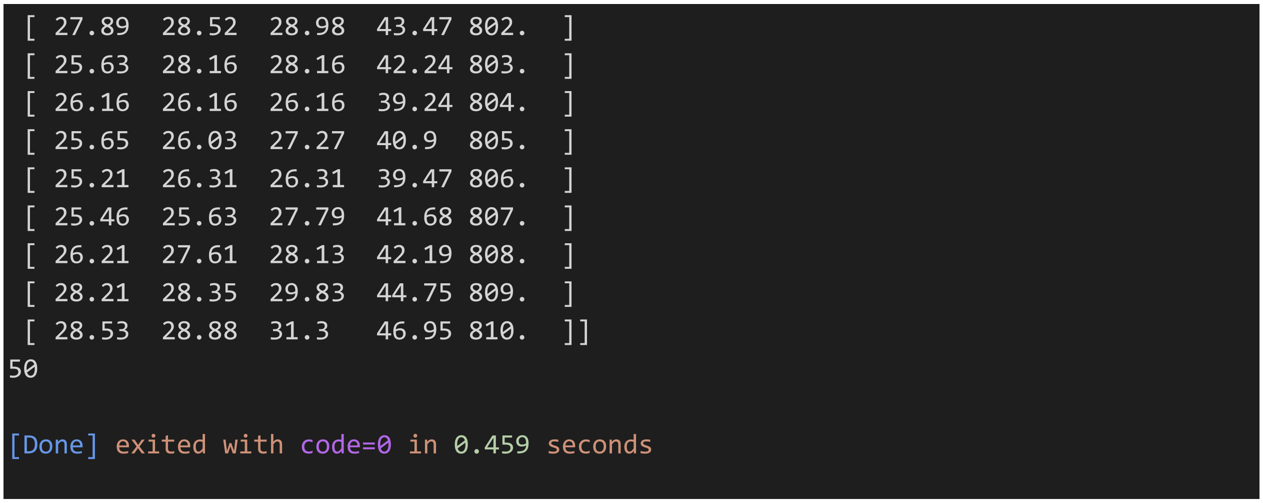
```
[Running] python -u "e:\_school\sem 2\EDS\prac_4.py"
[[801.    43.05  27.79  28.7   27.79]
 [802.    43.47  28.52  28.98  27.89]
 [803.    42.24  28.16  28.16  25.63]
 [804.    39.24  26.16  26.16  26.16]
 [805.    40.9   26.03  27.27  25.65]
 [806.    39.47  26.31  26.31  25.21]
 [807.    41.68  25.63  27.79  25.46]
 [808.    42.19  27.61  28.13  26.21]
 [809.    44.75  28.35  29.83  28.21]
 [810.    46.95  28.88  31.3   28.53]]
[[801.    28.48  34.18  30.56  22.23]
 [802.    28.1   33.72  30.68  22.82]
 [803.    26.16  31.39  28.2   22.53]
 [804.    26.16  31.39  28.78  20.93]
 [805.    26.1   31.32  28.22  20.82]
 [806.    25.45  30.54  27.73  21.05]
 [807.    26.16  31.39  28.01  20.51]
 [808.    27.44  32.93  28.83  22.08]
 [809.    28.63  34.35  31.03  22.68]
 [810.    30.35  36.42  31.38  23.1 ]]
[[1602.     71.53   61.97   59.26   50.02]
 [1604.     71.57   62.24   59.66   50.71]
 [1606.     68.4    59.55   56.36   48.16]
 [1608.     65.4    57.55   54.94   47.09]
 [1610.     67.     57.35   55.49   46.47]
 [1612.     64.92   56.85   54.04   46.26]
 [1614.     67.84   57.02   55.8    45.97]
 [1616.     69.63   60.54   56.96   48.29]
 [1618.     73.38   62.7    60.86   50.89]
 [1620.     77.3    65.3    62.68   51.63]] [[ 0.   14.57 -6.39 -1.86  5.56]
 [ 0.   15.37 -5.2  -1.7   5.07]
 [ 0.   16.08 -3.23 -0.04  3.1 ]
 [ 0.   13.08 -5.23 -2.62  5.23]
 [ 0.   14.8  -5.29 -0.95  4.83]
 [ 0.   14.02 -4.23 -1.42  4.16]
 [ 0.   15.52 -5.76 -0.22  4.95]
 [ 0.   14.75 -5.32 -0.7   4.13]
 [ 0.   16.12 -6.   -1.2   5.53]
 [ 0.   16.6  -7.54 -0.08  5.43]]
[[801.   802.   803.   804.   805.   806.   807.   808.   809.   810.  ]
 [ 43.05  43.47  42.24  39.24  40.9   39.47  41.68  42.19  44.75  46.95]
 [ 27.79  28.52  28.16  26.16  26.03  26.31  25.63  27.61  28.35  28.88]
 [ 28.7   28.98  28.16  26.16  27.27  26.31  27.79  28.13  29.83  31.3 ]
 [ 27.79  27.89  25.63  26.16  25.65  25.21  25.46  26.21  28.21  28.53]]
```



```
[[801.    43.05  27.79  28.7   27.79 801.    28.48  34.18  30.56  22.23]
 [802.    43.47  28.52  28.98  27.89 802.    28.1   33.72  30.68  22.82]
 [803.    42.24  28.16  28.16  25.63 803.    26.16  31.39  28.2   22.53]
 [804.    39.24  26.16  26.16  26.16 804.    26.16  31.39  28.78  20.93]
 [805.    40.9   26.03  27.27  25.65 805.    26.1   31.32  28.22  20.82]
 [806.    39.47  26.31  26.31  25.21 806.    25.45  30.54  27.73  21.05]
 [807.    41.68  25.63  27.79  25.46 807.    26.16  31.39  28.01  20.51]
 [808.    42.19  27.61  28.13  26.21 808.    27.44  32.93  28.83  22.08]
 [809.    44.75  28.35  29.83  28.21 809.    28.63  34.35  31.03  22.68]
 [810.    46.95  28.88  31.3   28.53 810.    30.35  36.42  31.38  23.1 ]]
[[801.    43.05  27.79  28.7   27.79]
 [802.    43.47  28.52  28.98  27.89]
 [803.    42.24  28.16  28.16  25.63]
 [804.    39.24  26.16  26.16  26.16]
 [805.    40.9   26.03  27.27  25.65]
 [806.    39.47  26.31  26.31  25.21]
 [807.    41.68  25.63  27.79  25.46]
 [808.    42.19  27.61  28.13  26.21]
 [809.    44.75  28.35  29.83  28.21]
 [810.    46.95  28.88  31.3   28.53]
 [801.    28.48  34.18  30.56  22.23]
 [802.    28.1   33.72  30.68  22.82]
 [803.    26.16  31.39  28.2   22.53]
 [804.    26.16  31.39  28.78  20.93]
 [805.    26.1   31.32  28.22  20.82]
 [806.    25.45  30.54  27.73  21.05]
 [807.    26.16  31.39  28.01  20.51]
 [808.    27.44  32.93  28.83  22.08]
 [809.    28.63  34.35  31.03  22.68]
 [810.    30.35  36.42  31.38  23.1 ]]
[0 1 2 3 4 5 6 7 8 9]
186.03499999999997 28.615000000000002 309.7929965912722 9301.749999999998
[[28.3019434   6.56124988  5.27162214  5.35723809  5.27162214]
 [28.31960452  6.59317829  5.34041197  5.38330753  5.28109837]
 [28.33725463  6.49923072  5.30659966  5.30659966  5.06260802]
 [28.35489376  6.26418391  5.11468474  5.11468474  5.11468474]
 [28.37252192  6.39531078  5.10196041  5.22206856  5.0645829 ]
 [28.39013913  6.28251542  5.12932744  5.12932744  5.02095608]
 [28.40774542  6.45600496  5.06260802  5.27162214  5.04579032]
 [28.42534081  6.49538298  5.25452186  5.30377224  5.11957029]
 [28.44292531  6.68954408  5.3244718   5.46168472  5.31130869]
 [28.46049894  6.85200701  5.37401154  5.59464029  5.34134814]] [[6.68586095
3.76236223 3.32467624 3.35689712 3.32467624]
 [6.68710861 3.77207105 3.3506056  3.36660594 3.3282682 ]
 [6.68835471 3.74336764 3.33790253 3.33790253 3.24376354]
```



```
 [6.68959927 3.66969663 3.26423153 3.26423153 3.26423153]
 [6.69084228 3.71113006 3.25924972 3.3057872  3.24454357]
 [6.69208374 3.67554089 3.2699491  3.2699491  3.22724074]
 [6.69332367 3.7300214  3.24376354 3.32467624 3.23710859]
 [6.69456206 3.74218323 3.31817803 3.33683662 3.26614102]
 [6.69579892 3.80109144 3.34462703 3.3955146  3.33967653]
 [6.69703425 3.84908321 3.36314931 3.4436181  3.35095617]]
[[801.    43.05  27.79  28.7   27.79]
 [802.    43.47  28.52  28.98  27.89]
 [803.    42.24  28.16  28.16  25.63]
 [804.    39.24  26.16  26.16  26.16]
 [805.    40.9   26.03  27.27  25.65]
 [806.    39.47  26.31  26.31  25.21]
 [807.    41.68  25.63  27.79  25.46]
 [808.    42.19  27.61  28.13  26.21]
 [809.    44.75  28.35  29.83  28.21]
 [810.    46.95  28.88  31.3   28.53]]
[[801.    43.05  27.79  28.7   27.79]
 [802.    43.47  28.52  28.98  27.89]
 [803.    42.24  28.16  28.16  25.63]
 [804.    39.24  26.16  26.16  26.16]
 [805.    40.9   26.03  27.27  25.65]]
[[[801.    43.05  27.79  28.7   27.79]
  [802.    43.47  28.52  28.98  27.89]
  [803.    42.24  28.16  28.16  25.63]
  [804.    39.24  26.16  26.16  26.16]
  [805.    40.9   26.03  27.27  25.65]
  [806.    39.47  26.31  26.31  25.21]
  [807.    41.68  25.63  27.79  25.46]
  [808.    42.19  27.61  28.13  26.21]
  [809.    44.75  28.35  29.83  28.21]
  [810.    46.95  28.88  31.3   28.53]]

 [[801.    28.48  34.18  30.56  22.23]
  [802.    28.1   33.72  30.68  22.82]
  [803.    26.16  31.39  28.2   22.53]
  [804.    26.16  31.39  28.78  20.93]
  [805.    26.1   31.32  28.22  20.82]
  [806.    25.45  30.54  27.73  21.05]
  [807.    26.16  31.39  28.01  20.51]
  [808.    27.44  32.93  28.83  22.08]
  [809.    28.63  34.35  31.03  22.68]
  [810.    30.35  36.42  31.38  23.1 ]]]
(array([0], dtype=int64), array([1], dtype=int64))
[[ 27.79  27.79  28.7   43.05 801.  ]
```



```
 [ 27.89  28.52  28.98  43.47 802.   ]
 [ 25.63  28.16  28.16  42.24 803.   ]
 [ 26.16  26.16  26.16  39.24 804.   ]
 [ 25.65  26.03  27.27  40.9  805.   ]
 [ 25.21  26.31  26.31  39.47 806.   ]
 [ 25.46  25.63  27.79  41.68 807.   ]
 [ 26.21  27.61  28.13  42.19 808.   ]
 [ 28.21  28.35  29.83  44.75 809.   ]
 [ 28.53  28.88  31.3   46.95 810.   ]]
50

[Done] exited with code=0 in 0.459 seconds
```