EDS ASSIGNMENT 3

NAME- APOORVA SINGH

DIVISION- C

ROLL NO. - 301

PRN- 202201070030

Prepare/Take datasets for any real-life application. Read a dataset into an array. Perform the following operations on it:

1. **Perform all matrix operations**
2. **Horizontal and vertical stacking of Numpy Arrays**
3. **Custom sequence generation**
4. **Arithmetic and Statistical Operations, Mathematical Operations, Bitwise Operators**
5. **Copying and viewing arrays**
6. **Data Stacking, Searching, Sorting, Counting, Broadcasting**

CSV:

https://drive.google.com/file/d/1PRaO6gMfoDDkRyIdh63JfRnXv8MGiQiw/view?usp=sharing

CODE:

```
#EDS ASSIGNMENT 3 : "NUMPY OPERATIONS"


import numpy as np

array = np.loadtxt('/content/drive/MyDrive/scores.csv', delimiter=',' ,skiprows=1)
print(array)
math_scores =[]
science_scores=[]
```

```
english_scores=[]


for i in array:
  math_scores.append(int(i[1]))
  science_scores.append(int(i[2]))
  english_scores.append(int(i[3]))


#converting list into array

arr_ms=np.array(math_scores)
arr_ss=np.array(science_scores)
arr_es=np.array(english_scores)

#displaying the array

print("* MATH SCORES: ",arr_ms)
print("* SCIENCE SCORES: ",arr_ss)
print("* ENGLISH SCORES: ",arr_es)

# Addition

total_scores = arr_ms + arr_ss + arr_es
print("1.Addition :",total_scores)

# Subtraction

math_minus_english = arr_ms - arr_es
print("2.Subtraction :",math_minus_english)

# Multiplication

science_times_2 = arr_ss * 2
print("3.Multiplication :",science_times_2)

# Division

english_divided_by_math = arr_es / arr_ms
print("4.Division :",english_divided_by_math)

# Transpose

english_transposed = np.transpose(arr_es)
print("5.Transpose :",english_transposed)


# Horizontal stacking

horizontal_stack = np.hstack((arr_ms, arr_ss, arr_es))
print("6.Horizontal stacking :",horizontal_stack)

# Vertical stacking
```

```
vertical_stack = np.vstack((arr_ms, arr_ss, arr_es))
print("7.Vertical stacking :",vertical_stack)


# Generate sequence of science score indices 0 to 4 along with values

indices = np.arange(len(arr_ss))

# Access data using the generated indices

for i in indices:
    print("8.Science score at index", i, ":", arr_ss[i])



# Copying arrays

math_scores_copy = arr_ms.copy()
print("9.Copying arrays :",math_scores_copy)

# Viewing arrays

science_scores_view = arr_ss.view()
print("10.Viewing arrays :",science_scores_view)



# Data Stacking

data_stack = np.stack((arr_ms, arr_ss, arr_es), axis=1)
print("11.Data Stacking :",data_stack)

# Searching

index_of_92 = np.where(arr_ms == 92)
print("12.Searching :",index_of_92)

# Sorting

sorted_math_scores = np.sort(arr_ms)
print("13.Sorting :",sorted_math_scores)

# Counting

count_88 = np.count_nonzero(arr_es == 88)
print("14.Counting :",count_88)

# Broadcasting

broadcasted_sum = arr_ms + 10
print("15.Broadcasting :",broadcasted_sum)
```

OUTPUT:

```
[[ 1. 90. 82. 88.]
 [ 2. 85. 95. 92.]
 [ 3. 92. 88. 85.]
 [ 4. 78. 79. 90.]
 [ 5. 88. 91. 94.]]
* MATH SCORES:  [90 85 92 78 88]
* SCIENCE SCORES:  [82 95 88 79 91]
* ENGLISH SCORES:  [88 92 85 90 94]
1.Addition : [260 272 265 247 273]
2.Subtraction : [  2  -7   7 -12  -6]
3.Multiplication : [164 190 176 158 182]
4.Division : [0.97777778 1.08235294 0.92391304 1.15384615 1.06818182]
5.Transpose : [88 92 85 90 94]
6.Horizontal stacking : [90 85 92 78 88 82 95 88 79 91 88 92 85 90 94]
7.Vertical stacking : [[90 85 92 78 88]
 [82 95 88 79 91]
 [88 92 85 90 94]]
8.Science score at index 0 : 82
8.Science score at index 1 : 95
8.Science score at index 2 : 88
8.Science score at index 3 : 79
8.Science score at index 4 : 91
9.Copying arrays : [90 85 92 78 88]
10.Viewing arrays : [82 95 88 79 91]
11.Data Stacking : [[90 82 88]
 [85 95 92]
 [92 88 85]
 [78 79 90]
 [88 91 94]]
12.Searching : (array([2]),)
13.Sorting : [78 85 88 90 92]
14.Counting : 1
15.Broadcasting : [100  95 102  88  98]
```