

Name : Makwana Vishal Raghavbhai

Class : FYMCA (SEM-2)

Div : B

Roll No: 113

Paper Number : 204

Subject : Python Programming Language

Assignment Number : 7

Dept. of Computer Science,
Veer Narmad South Gujarat University, Surat.
M.C.A. 2nd Semester (2020-2021)
Paper 204: Python Programming Language
Practical Sheet - 7

Q.1. Consider a 4 x 4 NumPy array of your choice. Write a Python program that finds the following and display them in an appropriate format for the given NumPy array: (NOTE: DO NOT USE BUILT-IN FUNCTIONS TO FIND THE MAXIMUM, MINIMUM AND SUM)

- Maximum, Minimum and Sum of all the elements of the matrix
- Maximum, Minimum and Sum of all the elements of each row
- Maximum, Minimum and Sum of all the elements of each column
- Maximum, Minimum and Sum of all the diagonal elements

```
import numpy as np
arr1 = np.arange(16, dtype = np.int_).reshape(4, 4)
sum=0
for i in range(0,4):
    for j in range(0,4):
        arr1[i][j] = int(input("enter the value in
array : "))
print("-----MATRIX-----")
for i in range(0,4):
    for j in range(0,4):
        print(arr1[i][j],end=' ')
        sum = sum + arr1[i][j]
    print()
maximum = arr1[0][0]
for i in range(0,4):
    for j in range(0,4):
        if(arr1[i][j]>maximum):
            maximum = arr1[i][j]
minimum = arr1[0][0]
for i in range(0,4):
    for j in range(0,4):
```

```

        if(arr1[i][j]<minimum):
            minimum = arr1[i][j]
maximum1 = arr1[0][0]
maximum2 = arr1[1][0]
maximum3 = arr1[2][0]
maximum4 = arr1[3][0]
minimum1 = arr1[0][0]
minimum2 = arr1[1][0]
minimum3 = arr1[2][0]
minimum4 = arr1[3][0]
sum1=0
sum2=0
sum3=0
sum4=0
for i in range(0,4):
    for j in range(0,4):
        if (i==0):
            if(arr1[i][j] > maximum1):
                maximum1 = arr1[i][j]
            if(arr1[i][j] <minimum1):
                minimum1 = arr1[i][j]
            sum1 = sum1 + arr1[i][j]
        if (i==1):
            if(arr1[i][j] > maximum2):
                maximum2 = arr1[i][j]
            if(arr1[i][j] < minimum2):
                minimum2 = arr1[i][j]
            sum2 = sum2 + arr1[i][j]
        if (i==2):
            if(arr1[i][j] > maximum3):
                maximum3 = arr1[i][j]
            if(arr1[i][j] < minimum3):
                minimum3 = arr1[i][j]
            sum3 = sum3 + arr1[i][j]
        if (i==3):
            if(arr1[i][j] > maximum4):
                maximum4 = arr1[i][j]
            if(arr1[i][j] < minimum4):
                minimum4 = arr1[i][j]
            sum4 = sum4 + arr1[i][j]
maximum01 = arr1[0][0]
maximum02 = arr1[0][1]
maximum03 = arr1[0][2]
maximum04 = arr1[0][3]

```

```

minimum01 = arr1[0][0]
minimum02 = arr1[0][1]
minimum03 = arr1[0][2]
minimum04 = arr1[0][3]
sum01=0
sum02=0
sum03=0
sum04=0
for i in range(0,4):
    for j in range(0,4):
        if (j==0):
            if(arr1[i][j] > maximum01):
                maximum01 = arr1[i][j]
            if(arr1[i][j] < minimum01):
                minimum01 = arr1[i][j]
            sum01 = sum01 + arr1[i][j]
        if (j==1):
            if(arr1[i][j] > maximum02):
                maximum02 = arr1[i][j]
            if(arr1[i][j] < minimum02):
                minimum02 = arr1[i][j]
            sum02 = sum02 + arr1[i][j]
        if (j==2):
            if(arr1[i][j] > maximum03):
                maximum03 = arr1[i][j]
            if(arr1[i][j] < minimum03):
                minimum03 = arr1[i][j]
            sum03 = sum03 + arr1[i][j]
        if (j==3):
            if(arr1[i][j] > maximum04):
                maximum04 = arr1[i][j]
            if(arr1[i][j] < minimum04):
                minimum04 = arr1[i][j]
            sum04 = sum04 + arr1[i][j]
print("-----MAXIMUM-----")
print("maximum in matrix      :",maximum)
print("maximum from row 1     :",maximum1)
print("maximum from row 2     :",maximum2)
print("maximum from row 3     :",maximum3)
print("minimumum from row 4     :",maximum4)
print("maximum from column 1 :",maximum01)
print("maximum from column 2 :",maximum02)
print("maximum from column 3 :",maximum03)
print("minimumum from column 4 :",maximum04)

```

```
print ("-----MINIMUM-----")
```

```
print("minimum in matrix      :", minimum)
print("minimum from row 1     :", minimum1)
print("minimum from row 2     :", minimum2)
print("minimum from row 3     :", minimum3)
print("minimum from row 4     :", minimum4)
print("minimum from column 1  :", minimum01)
print("minimum from column 2  :", minimum02)
print("minimum from column 3  :", minimum03)
print("minimum from column 4  :", minimum04)
print ("-----SUM-----")
print("sum of  matrix          :", sum)
print("sum of  row 1           :", sum1)
print("sum of  row 2           :", sum2)
print("sum of  row 3           :", sum3)
print("sum of  row 4           :", sum4)
print("sum of column 1         :", sum01)
print("sum of column 2         :", sum02)
print("sum of column 3         :", sum03)
print("sum of column 4         :", sum04)
```

Q.2. Consider a 1-D NumPy array of 10 elements, where each element is temperature in degrees Celsius. Write a Python program to Convert it to an array containing temperature in degrees Fahrenheit. The relation between Celsius and Fahrenheit is $C / 5 = (F - 32) / 9$. Display them in an appropriate format.

```
import numpy as np

mat = np.random.randint(15, 60, 10)

def convert():
    fahrenheit = []
    for i in mat:
        fahrenheit.append((i * 9/5) + 32)
        # (0°C × 9/5) + 32 = 32°F
    return fahrenheit

if __name__ == '__main__':
    fahrenheit = convert()
    for i in range(len(mat)):
        print(str(mat[i]) + "°C", "=>",
              str(fahrenheit[i]) + "°F")
```

Q.3. Consider a 4 x 3 NumPy array and a 3 x 4 NumPy array. Write a Python program to perform the Matrix Multiplication of these two NumPy arrays. DO NOT USE THE BUILT-IN OPERATOR TO FIND THE MATRIX MULTIPLICATION.

```
import numpy as np

n = 12
r1, c1 = 4, 3
r2, c2 = 3, 4
a = np.random.randint(0, 10, n).reshape(r1, c1)
b = np.random.randint(0, 10, n).reshape(r2, c2)
c = np.zeros((r1, c2), int)

def mul():
    if c1 == r2:
        for k in range(c2):
            for i in range(c2):
                for j in range(r2):
                    c[k][i] = c[k][i] + a[k][j] * b[j][i]

    return c

if __name__ == '__main__':
    print(a, b, sep='\n')
    print("-".ljust(20, '-'))
    print(mul())
```

Q.4. Consider a NumPy array where each row represents data of a student and there are 10 such rows. The data of a student consists of rollno, name, city and age. i.e. each row contains rollno, name, city and age of a student. Write a Python program to find the following:

- Maximum, Minimum and Average age of all the students living in a particular city (take input of name of a city from the user)
- Maximum, Minimum and Average age of all the students whose name starts with the letter 'A'
- Maximum, Minimum and Average age of all the students having rollno > n (take input of n from the user)

```
import numpy as np
import sys

from numpy.core.numeric import roll, rollaxis

data_type = [('rollno', 'int'), ('name', 'U25'),
              ('city', 'U25'), ('age', 'int'), ]
data = [(1, 'Vishal', 'surat', 21), (2, 'Hardik',
    'surat', 15), (3, 'Maulik', 'amreli', 23), (4,
    'Ashish', 'surat', 16), (5, 'Jaydeep', 'ahemdabad',
    22), (6, 'Arshit', 'Rajkot', 13), (7, 'Raj', 'amreli',
    23), (8, 'Rahul', 'ahemdabad', 21), (9, 'Arvind',
    'surat', 20), (10, 'Vairag', 'surat', 21),]

student = np.array(data, dtype=data_type)

def print_data(*args):
    print("Min = ", args[0])
    print("Max = ", args[1])
    print("Avg = ", round(args[2], 2))

def all_min_max_avg(student):
    print(" First ".center(25, '-'))
    print(student)
    age = student["age"]
    print_data(min(age), max(age), sum(age)/len(age))
```



```

def city_wise(student):
    print(" Second ".center(25, '-'))
    city_name = input("Enter city name: ")
                    .lower().strip()
    if city_name in student["city"]:
        data = student[student["city"] == city_name]
        print(data)
        age = data["age"]
        print_data(min(age), max(age),
                    sum(age)/len(age))
    else:
        print(city_name, " is not the correct
                        value...")

def start_with_A(student):
    print(" Third ".center(25, '-'))
    data = student[[True if i.startswith('A') else
False for i in student["name"]]]
    print(data)
    age = data["age"]
    print_data(min(age), max(age), sum(age)/len(age))

def rollno_greater(student):
    print(" Four ".center(25, '-'))
    try:
        roll_no = int(input("Enter roll no: "))
    except ValueError as e:
        print(e)
        sys.exit()

    if roll_no in student["rollno"]:
        data = student[student["rollno"] > roll_no]
        print(data)
        age = data["age"]
        print_data(min(age), max(age),
                    sum(age)/len(age))
    else:
        print(roll_no, " is not the correct value...")

if __name__ == '__main__':
    all_min_max_avg(student)

```

```
city_wise(student)  
start_with_A(student)  
rollno_greater(student)
```

Q.5. Consider the NumPy array used in Q.4. Write a Python program to do the following:

- Display details of all the students living in a particular city (take input of name of a city from the user) in appropriate format
- Display details of all the students having age greater than N (take input of N from the user) in appropriate format

```
import numpy as np
import sys

data_type = [('rollno', 'int'), ('name', 'U25'),
              ('city', 'U25'), ('age', 'int'), ]
data = [(1, 'Vishal', 'surat', 21), (2, 'Hardik',
' surat', 15), (3, 'Maulik', 'amreli', 23), (4,
'Ashish', 'surat', 16), (5, 'Jaydeep', 'ahemdabad',
22), (6, 'Arshit', 'Rajkot', 13), (7, 'Raj',
'amreli', 23), (8, 'Rahul', 'ahemdabad', 21), (9,
'Arvind', 'surat', 20), (10, 'Vairag', 'surat', 21),]

student = np.array(data, dtype=data_type)

def print_data(data):
    print('-'.center(80, '-'))

    print("{:^20}{:^20}{:^20}{:^20}".format(*data.dtype.
names))

    print("{:^20}{:^20}{:^20}{:^20}".format(*['-
'*len(i) for i in data.dtype.names]))
    for i in data:
        print("{:^20}{:^20}{:^20}{:^20}".format(*i))

def city_wise(student):
    print(" First ".center(25, '-'))
    city_name = input("Enter city name:
").lower().strip()
    if city_name in student["city"]:
        data = student[student["city"] == city_name]
```

```

        print_data(data)
    else:
        print(city_name, " is not the correct
                    value...")

def rollno_greater(student):
    print(" Second ".center(25, '-'))
    try:
        roll_no = int(input("Enter roll no: "))
        if roll_no in student["rollno"]:
            data = student[student["rollno"] > roll_no]
            print_data(data)
        else:
            print(roll_no, " is not the correct
                        value...")
    except ValueError as e:
        print(e)
        sys.exit()

if __name__ == '__main__':
    city_wise(student)
    rollno_greater(student)

```
