PPL Lab Assignment 4

Jaynam Modi. PG-43. G3. Sep 03, 2020

- Problem Statement : Given a square matrix of N rows and columns, find out whether it is symmetric or not.
- Objectives :

1. To learn about the Matrix in Python programming language.
2. To understand the concept of interpreted language working.

- Theory :

- Transpose Matrix & Symmetric Matrix : the transpose of a matrix is an operator which flips a matrix over its diagonal; that is, it switches the row and column indices of the matrix A by producing another matrix, A square matrix whose transpose is equal to itself is called a symmetric matrix.
- Matrix in python : Numpy is the core library for scientific computing in Python

It provides a high-performance multidimensional array object, and tools for working with these arrays.

1. Concept of matrix manipulation using Numpy : In Python we can solve the different matrix manipulations and operations. Numpy Module provides different methods for matrix operations.

- add( ) - add elements of two matrices.
- subtract( ) - subtract elements of two matrices.
- divide( ) - divide elements of two matrices.
- multiply( ) - multiply elements of two matrices.
- dot( ) - It performs matrix multiplication, does not element wise

multiplication.

- sqrt( ) - square root of each element of matrix.

- sum(x,axis) - add to all the elements in matrix. Second argument is optional, it is used when we want to compute the column sum if axis is 0 and row sum if axis is 1.

- "T" - It performs transpose of the specified matrix.

- Algorithm :

- Problem Solution :

1. Declare a matrix of N rows and columns
2. Define a function to create a transpose of given matrix

3. Then check with the two for loops if ith row and jth column are present in the given range that is N.

4. Compare given matrix with transpose matrix if they are similar then return symmetric otherwise return asymmetric.

5. Exit

- Sample Code :

```
# Simple Python code for check a matrix is
# symmetric or not.

# Fills transpose of mat[N][N] in tr[N][N]
def transpose(mat, tr, N):

for i in range(N): for j in range(N):

tr[i][j] = mat[j][i]

# Returns true if mat[N][N] is symmetric,
else false def isSymmetric(mat, N):
```

```
tr = [ [0 for j in range(len(mat[0])) ] for i
in range(len(mat)) ] transpose(mat, tr, N)

for i in range(N):


for j in range(N):
if (mat[i][j] != tr[i][j]): return False

return True

# Driver code

mat = [ [ 1, 3, 5 ], [ 3, 2, 4 ], [ 5, 4, 1 ] ]

if (isSymmetric(mat, 3)): print "Yes"

else:

print "No"
```

- Platform : Ubuntu
- Input : 1 2 3 2 1 4 3 4 3
- Output : Yes

- Conclusion: We have studied and executed program successfully

- FAQs:

1. What are the key features of python?

> Python has clean object-oriented design, provides enhanced process control capabilities, and possesses strong integration and text processing capabilities and its own unit testing framework, all of which contribute to the increase in its speed and productivity. Python is considered a viable option for building complex multi-protocol network applications.

2. What is the difference between python arrays and list

> Numpy is the core library for scientific computing in Python. It provides a high-performance multidimensional array object, and tools for working with these arrays. A numpy array is a grid of values, all of the same type, and is indexed by a tuple of nonnegative integers. The number of dimensions is the rank of the array; the shape of an array is a tuple of integers giving the size of the array along each

dimension.

The Python core library provided Lists. A list is the Python equivalent of an array, but is resizeable and can contain elements of different types.

3. What are functions in python?

> A function is a block of organized, reusable code that is used to perform a single, related action. Functions provide better modularity for your application and a high degree of code reusing.

As you already know, Python gives you many built-in functions like print(), etc. but you can also create your own functions. These functions are called user-defined functions.

4. what is the purpose of for and if statement in python

> A for loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string).

> The if..else statement evaluates test expression and will execute the body of if only when the test condition is True. If the

condition is False, the body of else is executed. Indentation is used to separate the blocks.

Practice Assignments:

1. Python program to create matrix using Numpy
2. python program to Add two matrices
3. python program to multiply two matrices
4. Python program to find row-wise maximum element of matrix

```python

# PPL Lab Assignment 4, PG43 Jaynam Modi, G3

# Write a Python Program to input a Numpy Array and print it's transpose and find out whether it is Symmetric or not.

import numpy as np


def printmatrix( arr ):
    for i in range(arr.shape[0]):
        for j in range(arr.shape[1]):
            print("    ", arr[i][j], end = "")
        print("\n")

def issymetric( arr ):
    flag = True
    t = arr.transpose()
    for i in range(arr.shape[0]):
        for j in range(arr.shape[1]):
            if arr[i][j] != t[i][j]:
                flag = False
    return flag


i, j = (int(a) for a in input(" > Enter dimensions of the Array : ").split(" ") if a != "")
inpar = np.array([int(x) for x in input(" > Enter Elements of {} × {} matrix : ".format(i, j)).split(" ") if x != ""]).reshape((i, j))

print(" > Stored Array : \n")
printmatrix(inpar)

transar = inpar.transpose()

print(" > Transpose of the Array : \n")
printmatrix(transar)

if issymetric(inpar):
    print(" > The given Array is symmetric.")
else:
    print(" > The given Array is NOT symmetric.")


# PRACTICE PROBLEMS.

# 1. Python program to Add two matrices.

def addMatrix(x, y):
    sumMatrix = x + y
    return sumMatrix

# 2. Python program to multiply two matrices.

def productMatrix(x, y):
    return numpy.matmul(x, y)

# 3. Python program to find row-wise maximum element of matrix.

def maxElementInRow(arr):
    for x in range(arr.shape[0]):
        m = arr[x][0]
        for y in range(arr.shape[1]):
            if arr[x][y] > m:
                m = arr[x][y]
        print(" > Greatest element in Row {} is : {}".format(x+1, m))

```

```
u0_a362@localhost:~$ cd github/assignments/PPL7
u0_a362@localhost:~/github/assignments/PPL$ python ppl_assignment_4.py
 > Enter dimensions of the Array : 3 3
 > Enter Elements of 3 × 3 matrix : 1 2 3 2 1 4 3 4 3
 > Stored Array :

    1     2     3

    2     1     4

    3     4     3

 > Transpose of the Array :

    1     2     3

    2     1     4

    3     4     3

 > The given Array is symmetric.
```