

Name: Rohan Babulnath Kamble

Class: S. Y. B. Sc. CS

PRN NO: 2020420004

Subject Code: USCS405

Subject Name: Linear Algebra using Python

Practical – 1

Aim :- Write a program that demonstrates the following. i)

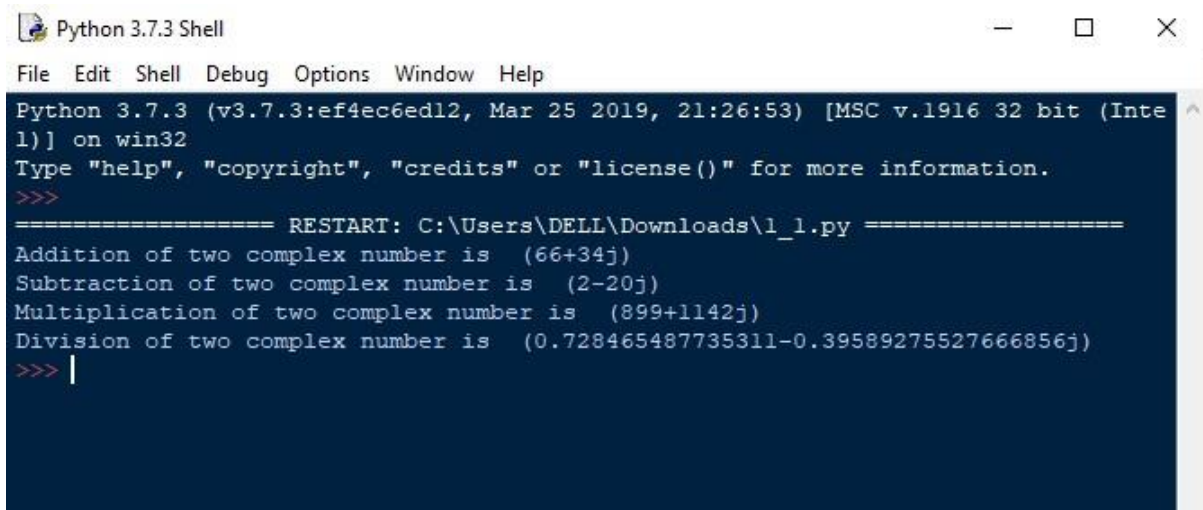
Addition of two complex numbers.

ii) **Displaying the conjugate of a complex number.** iii) **Plotting a set of complex numbers.** iv) **Creating a new plot by rotating the given number by a degree 90 , 180 , 270 degrees and also by scaling by a number $a = \frac{1}{2}$, $a = \frac{1}{3}$, $a = 2$ etc.**

Program Code :- i)

```
c1=34+7j c2=32+27j
```

```
print("Addition of two complex number is ", c1+c2) print("Subtraction  
of two complex number is ", c1-c2) print("Multiplication of two  
complex number is ", c1*c2) print("Division of two complex number  
is ", c1/c2) Output:-
```

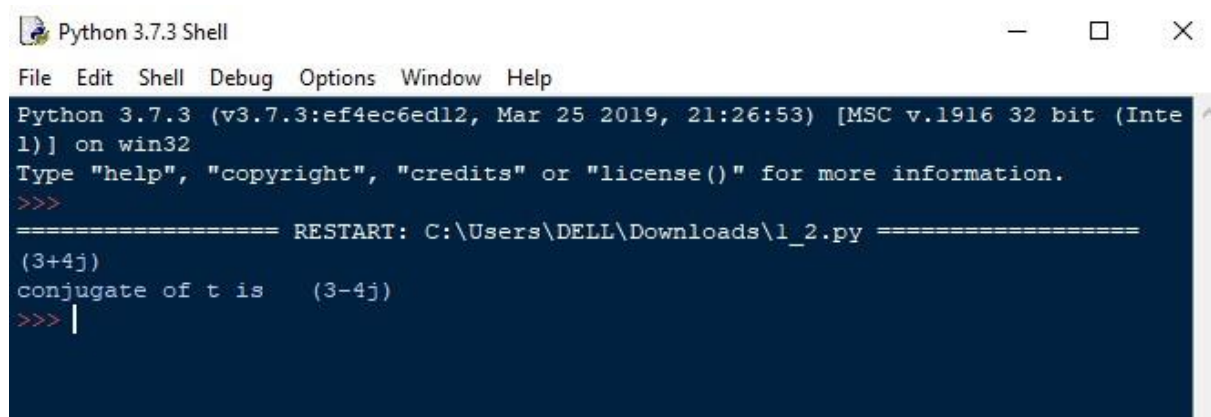


```
Python 3.7.3 Shell
File Edit Shell Debug Options Window Help
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\DELL\Downloads\l_1.py =====
Addition of two complex number is (66+34j)
Subtraction of two complex number is (2-20j)
Multiplication of two complex number is (899+1142j)
Division of two complex number is (0.728465487735311-0.3958927552766856j)
>>> |
```

ii)

```
t=3+4j print(t)
m=t.conjugate()
print("conjugate of t is ",m)
```

Output:-



```
Python 3.7.3 Shell
File Edit Shell Debug Options Window Help
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\DELL\Downloads\l_2.py =====
(3+4j)
conjugate of t is (3-4j)
>>> |
```

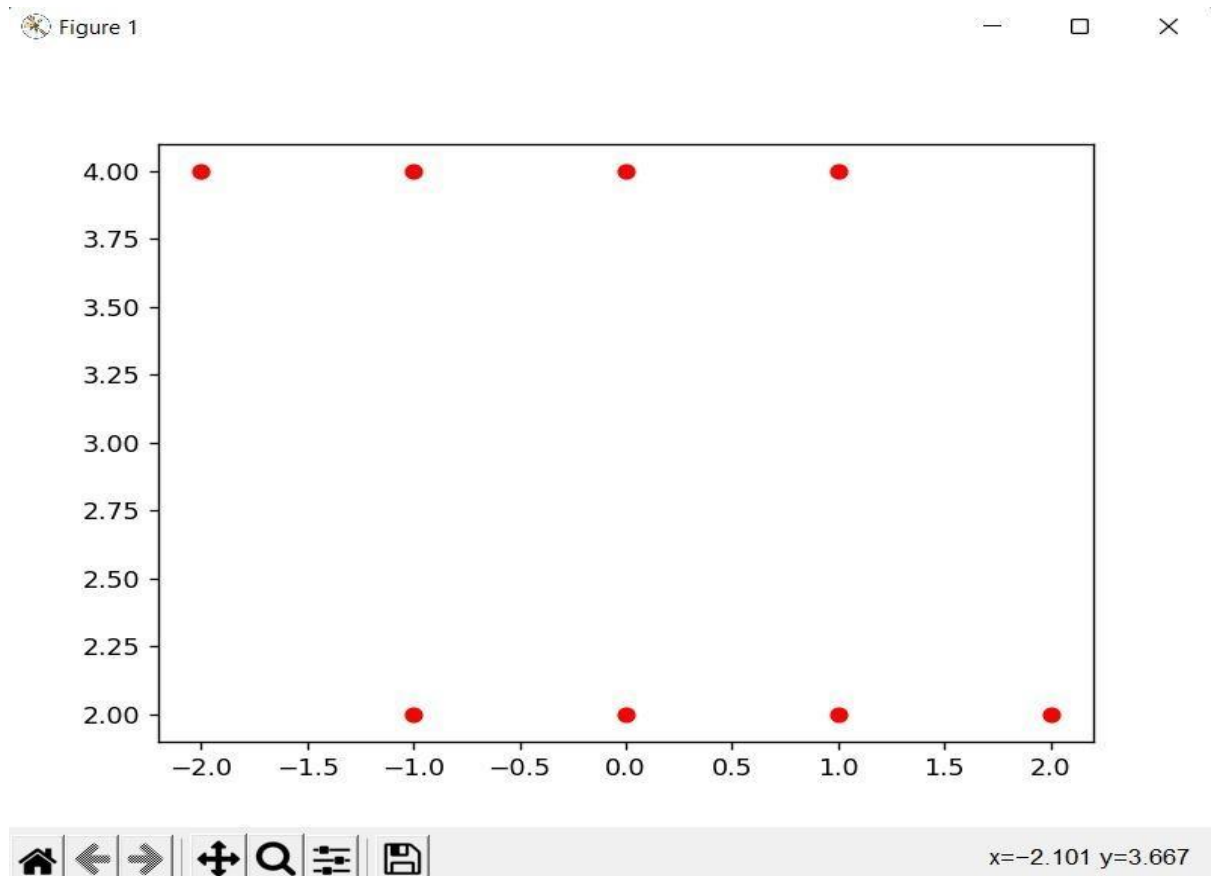
iii) import matplotlib.pyplot

as plt

```
x=3+2j
a=[-2+4j,-1+2j,0+2j,1+2j,2+2j,-1+4j,0+4j,1+4j]
A=[x.real for x in a] B=[x.imag
for x in a]
```

```
plt.scatter(A,B,color="red")  
plt.show()
```

Output:-



iv) `import matplotlib.pyplot as plt`

```
s={3+3j,4+3j,2+1j,5+1j,2+1j}
```

```
angle=int(input("Enter the angle rotation"))
```

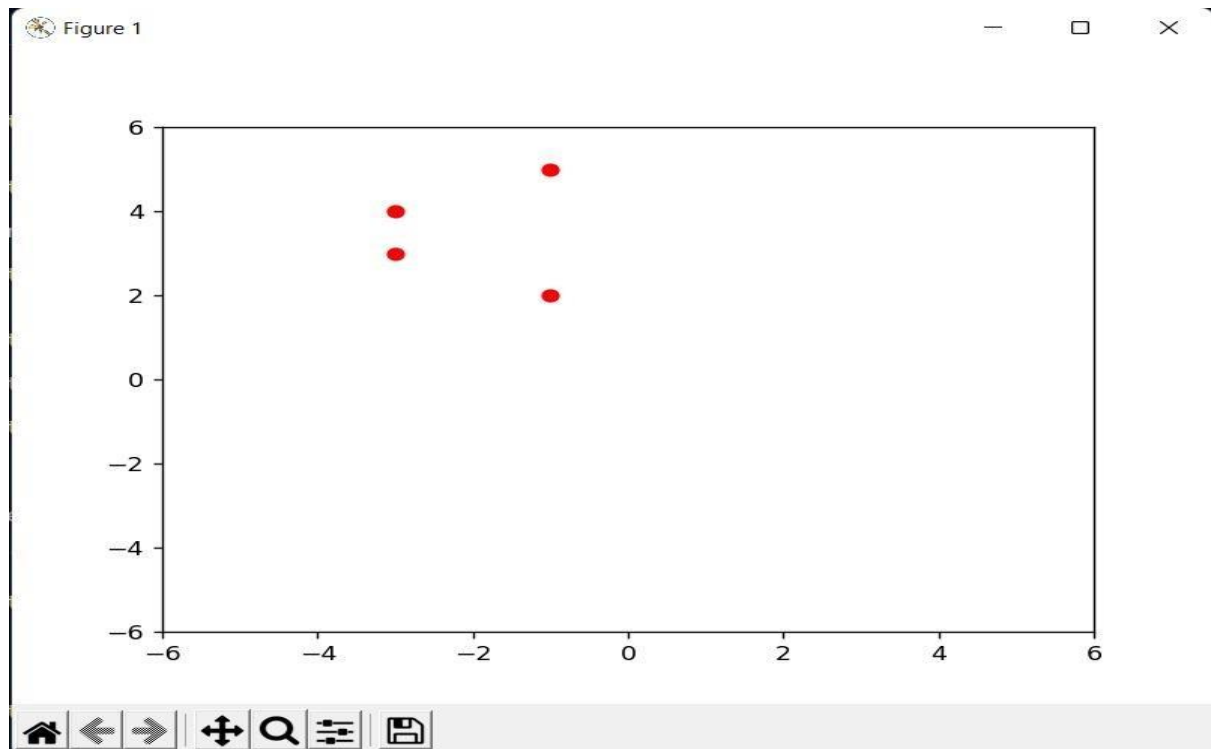
```
if angle==90: s1={x*1j for x in s}
```

```
print(s1) x=[x.real for x in s1]
```

```
y=[x.imag for x in s1] plt.plot(x,y,'ro')
```

```
plt.axis([-6,6,-6,6]) plt.show() else:  
print("invalid angle")
```

Output:-



Practical – 2

Aim:- Write a program to do the following:-

- i) Enter a vector u as a n-list ii) Enter another vector v as a n-list**
- iii) Find the vector $au+bv$ for different values of a and b iv) Find the dot product of u and v.**

Program Code:-

```
import numpy as np
#enter vector as n-list
x=np.array([5,6,7])
y=np.array([1,2,3])
print(x)    print(y)
print("enter values of
a    a    and    b    ")
a=int(input())
b=int(input())
c=a*x+b*y
d=np.dot(x,y)
print("au+bv vector
is    ",c)  print("dot
product is " , d)
```

Output:-

```
De11@DESKTOP-D5GGUQ7 MINGW64 ~/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra using Python/Practicals_files
$ python 2.py
[5 6 7]
[1 2 3]
enter values of a a and b
1
2
au+bv vector is [ 7 10 13]
dot product is 38
```

Practical – 3

Aim:- Write a program to do the following:-

- i) Enter an r by c matrix M (r and c being positive integers)**
- ii) Display M in matrix format**
- iii) Display the rows and columns of the matrix M**
- iv) Find the scalar multiplication of M for a given scalar v**
- v) Find the transpose of the matrix M .**

Program Code:-

```
i) & ii)    import numpy as np
             M=np.array([[1,1,1],[3,4,7],[9,6,3]])

             M

             #matrix M is print("matrix
             M is ",M)

             Y=M[0:1]

             Y

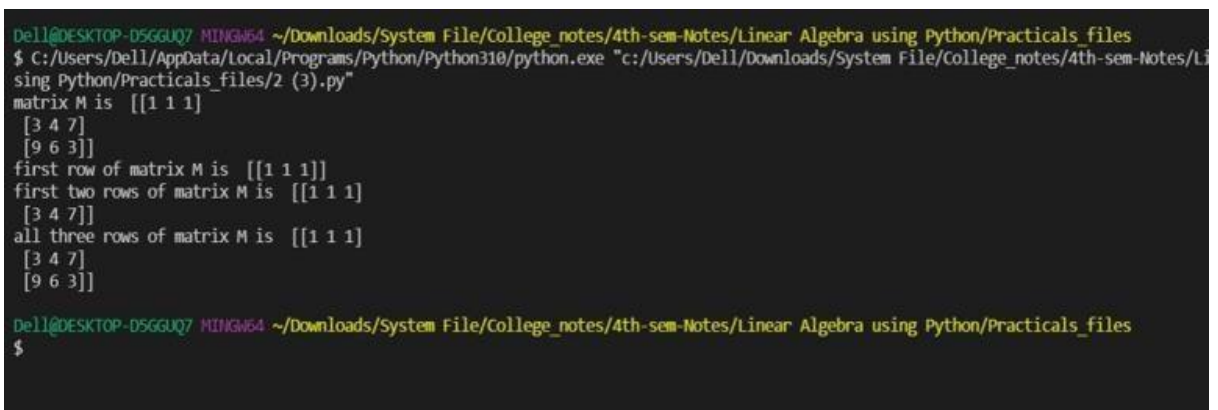
             #first row of matrix M is  print("first
             row of matrix M is ",Y) x=M[0:2]

             #first two rows of matrix M is
             print("first two rows of matrix M is ",x)

             t=M[0:3]

             #all rows of matrix M is print("all
             three rows of matrix M is ",t)
```

Output:-



```
Dell@DESKTOP-05GGUQ7 MINGW64 ~/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra using Python/Practicals_files
$ C:/Users/Dell/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/Dell/Downloads/System File/College_notes/4th-sem-Notes/Li
sing Python/Practicals_files/2 (3).py"
matrix M is  [[1 1 1]
 [3 4 7]
 [9 6 3]]
first row of matrix M is  [[1 1 1]]
first two rows of matrix M is  [[1 1 1]
 [3 4 7]]
all three rows of matrix M is  [[1 1 1]
 [3 4 7]
 [9 6 3]]

Dell@DESKTOP-05GGUQ7 MINGW64 ~/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra using Python/Practicals_files
$
```

iii)

```

import numpy as np
M=np.array([[1,1,1],[3,4,7],[9,6,3]])

M

#matrix M is print("matrix
M is ",M)

Y=M[:,0:1]

Y

#first column of matrix M is
print("first column of matrix M is ",Y)

x=M[:,0:2]

#first two columns of matrix M is print("first
two columns of matrix M is ",x) t=M[:,0:3]

#all columns of matrix M is print("all
three columns of matrix M is ",t)

```

Output:-

```

File "C:\Users\Dell\Downloads\System File\College_notes\4th-sem-Notes\Linear Algebra using Python\Practicals_files\2 (1).py", line 8, in <module>
    a=int(input())
KeyboardInterrupt

Dell@DESKTOP-D5GGUQ7 MINGW64 ~/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra using Python/Practicals_files
$ python "2 (2).py"
matrix M is  [[1 1 1]
 [3 4 7]
 [9 6 3]]
first column of matrix M is  [[1]
 [3]
 [9]]
first two columns of matrix M is  [[1 1]
 [3 4]
 [9 6]]
all three columns of matrix M is  [[1 1 1]
 [3 4 7]
 [9 6 3]]

```

iv)

```

import numpy as np

```



```

M=np.array([[1,1,1],[3,4,7],[9,6,3]])
M
#matrix M is print("matrix M is ",M) a=6
scalar=a*M print("scalar-matrix
multiplication is ",scalar)

```

Output:-

```

Dell@DESKTOP-D5GGUQ7 MINGW64 ~/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra using Python/Practicals_files
$ C:/Users/Dell/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/Dell/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra u
sing Python/Practicals_files/2 (4).py"
matrix M is  [[1 1 1]
 [3 4 7]
 [9 6 3]]
scalar-matrix multiplication is  [[ 6 6 6]
 [18 24 42]
 [54 36 18]]

Dell@DESKTOP-D5GGUQ7 MINGW64 ~/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra using Python/Practicals_files
$ 

```

```

v) x=[[12,7],[4,5],[3,8]]
t=[[0,0,0],[0,0,0]]
print("original matrix")
print(x) print("transpose of
matrix") for i in
range(len(x)):
    for j in
range(len(x[0])):
        t[j][i]=x[i][j]          for r
in t:
                                print(r)

```

Output:-

```
Dell@DESKTOP-05GGK07 MINGW64 ~/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra using Python/Practicals_files
$ C:/Users/Dell/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/Dell/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra u
sing Python/Practicals_files/2 (5).py"
original matrix
[[12, 7], [4, 5], [3, 8]]
transpose of matrix
[12, 0, 0]
[0, 0, 0]
[12, 0, 0]
[7, 0, 0]
[12, 4, 0]
[7, 0, 0]
[12, 4, 0]
[7, 5, 0]
[12, 4, 3]
[7, 5, 0]
[12, 4, 3]
[7, 5, 8]
```

Practical – 4

Aim:- Write a program to do the following:-

- i) Find the vector – matrix multiplication of a r by c matrix M with an c -vector u . ii) Find the matrix – matrix product of M with a c by p matrix N .

Program Code:-

i)

```
import numpy as np x=np.array([1,4,6])
y=np.array([[2,3],[3,4],[4,5]])
print(np.dot(x,y))
```

Output:-

```

Dell@DESKTOP-D5GGUQ7 MINGW64 ~/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra using Python/Practicals_files
$ C:/Users/Dell/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/Dell/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra u
sing Python/Practicals_files/2 (6).py"
[38 49]

Dell@DESKTOP-D5GGUQ7 MINGW64 ~/Downloads/System File/College_notes/4th-sem-Notes/Linear Algebra using Python/Practicals_files
$ █

```

ii)

```

import numpy as np
A=np.array([[3,2,2],[4,1,5],[1,2,3]])
print("matrix A is "A)
B=np.array([[1,2,3],[1,1,1],[2,2,2]])
print("matrix B is "B)
print("multiplication of two matrices A & B is ")
M=([[0,0,0],[0,0,0],[0,0,0]]) for i in
range(len(A)):    for j in range(len(B[0])):
    for k in range(len(B)):
        M[i][j]+=A[i][k]*B[k][j]
    for r in M:
        print(r)

```

Output:-

```
matrix A is [[3 2 2]
[4 1 5]
[1 2 3]]
matrix B is [[1 2 3]
[1 1 1]
[2 2 2]]
multiplication of two matrices A & B is
[3, 0, 0]
[0, 0, 0]
[0, 0, 0]
[5, 0, 0]
[0, 0, 0]
[0, 0, 0]
[9, 0, 0]
[0, 0, 0]
[0, 0, 0]
[9, 6, 0]
[0, 0, 0]
[0, 0, 0]
[9, 8, 0]
[0, 0, 0]
[0, 0, 0]
[9, 12, 0]
[0, 0, 0]
[0, 0, 0]
[9, 12, 9]
[0, 0, 0]
[0, 0, 0]
[9, 12, 11]
[0, 0, 0]
[0, 0, 0]
[9, 12, 15]
[0, 0, 0]
[0, 0, 0]
[9, 12, 15]
[4, 0, 0]
[0, 0, 0]
[9, 12, 15]
[5, 0, 0]
[0, 0, 0]
[9, 12, 15]
[15, 0, 0]
[0, 0, 0]
```

```
[9, 12, 15]  
[15, 19, 23]  
[9, 10, 0]  
[9, 12, 15]  
[15, 19, 23]  
[9, 10, 3]  
[9, 12, 15]  
[15, 19, 23]  
[9, 10, 5]  
[9, 12, 15]  
[15, 19, 23]  
[9, 10, 11]
```