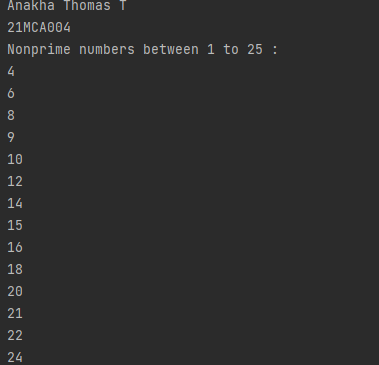
DATA SCEINCE & MACHINE LEARNING:

LAB CYCLE 1

1. Program to Print all non-Prime Numbers in an Interval.

print("Anakha Thomas T")  
print("21MCA004")  
import math  
def is\_not\_prime(n):  
 ans = False  
 for i in range(2, int(math.sqrt(n)) + 1):  
 if n % i == 0:  
 ans = True  
 return ans  
print("Nonprime numbers between 1 to 100:")  
for x in filter(is\_not\_prime, range(1, 101)):  
 print(x)

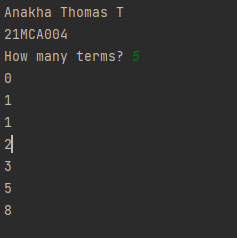
**OUTPUT:**



1. Program to print the first N Fibonacci numbers.

print("Anakha Thomas T")  
print("21MCA004")  
nterms = int(input("How many terms? "))  
n1, n2 = 0, 1  
count = 0  
print(n1)  
print(n2)  
for i in range (0,nterms):  
 count=n1+n2  
 print(count)  
 n1=n2  
 n2=count

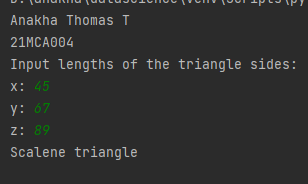
output:



1. Given sides of a triangle, write a program to check whether given triangle is an isosceles, equilateral or scalene.

print("Anakha Thomas T")  
print("21MCA004")  
print("Input lengths of the triangle sides: ")  
x = int(input("x: "))  
y = int(input("y: "))  
z = int(input("z: "))  
  
if x == y == z:  
 print("Equilateral triangle")  
elif x==y or y==z or z==x:  
 print("isosceles triangle")  
else:  
 print("Scalene triangle")

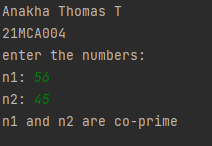
Output:



1. Program to check whether given pair of number is coprime

print("Anakha Thomas T")  
print("21MCA004")  
print("enter the numbers: ")  
n1 = int(input("n1: "))  
n2 = int (input("n2: "))  
for i in range(1,n1):  
 if n1%i==0 and n2%i==0:  
 hcf=i  
if hcf==1:  
 print("n1 and n2 are co-prime")  
else:  
 print("n1 and n2 are not co-prime")

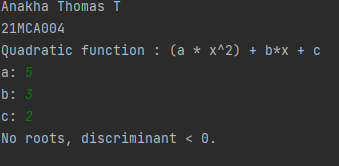
output:



1. Program to find the roots of a quadratic equation (rounded to 2 decimal places)

print("Anakha Thomas T")  
print("21MCA004")  
from math import sqrt  
print("Quadratic function : (a \* x^2) + b\*x + c")  
a = float(input("a: "))  
b = float(input("b: "))  
c = float(input("c: "))  
r = b \*\* 2 - 4 \* a \* c  
if r > 0:  
 num\_roots = 2  
 x1 = (((-b) + sqrt(r)) / (2 \* a))  
 x2 = (((-b) - sqrt(r)) / (2 \* a))  
 print("There are 2 roots: %f and %f" % (x1, x2))  
elif r == 0:  
 num\_roots = 1  
 x = (-b) / 2 \* a  
 print("There is one root: ", x)  
else:  
 num\_roots = 0  
 print("No roots, discriminant < 0.")  
 exit()

output:

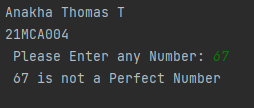


6. Program to check whether a given number is perfect number or not (sum of factors

=number)

print("Anakha Thomas T")  
print("21MCA004")  
Number = int(input(" Please Enter any Number: "))  
Sum = 0  
for i in range(1, Number):  
 if(Number % i == 0):  
 Sum = Sum + i  
if (Sum == Number):  
 print(" %d is a Perfect Number" %Number)  
else:  
 print(" %d is not a Perfect Number" %Number)

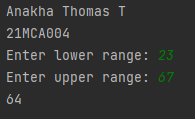
output:



7.Program to display Armstrong numbers unto 1000

print("Anakha Thomas T")  
print("21MCA004")  
lower = int(input("Enter lower range: "))  
upper = int(input("Enter upper range: "))  
  
for num in range(lower, upper + 1):  
 sum = 0  
 temp = num  
 while temp > 0:  
 digit = temp % 10  
 sum += digit \*\* 3  
 temp //= 10  
 if num == sum:  
 print(num)

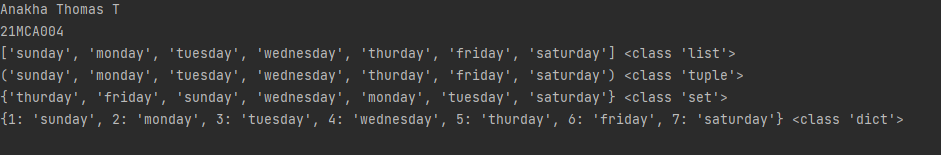
output:



8. Store and display the days of a week as a List, Tuple, Dictionary, Set. Also Demonstrate different ways to store values in each of them. Display its type also.

list1=["sunday","monday","tuesday","wednesday","thurday","friday","saturday"]  
tuple=("sunday","monday","tuesday","wednesday","thurday","friday","saturday")  
set={"sunday","monday","tuesday","wednesday","thurday","friday","saturday"}  
dict={1:"sunday",2:"monday",3:"tuesday",4:"wednesday",5:"thurday",6:"friday",7:"saturday"}  
print(list1,type(list1))  
print(tuple,type(tuple))  
print(set,type(set))  
print(dict,type(dict))

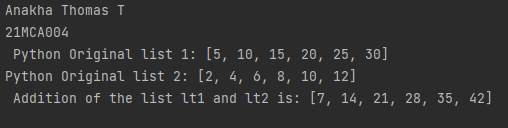
Output:



9. Write a program to add elements of given 2 lists.

print("Anakha Thomas T")  
print("21MCA004")  
l1 = [5, 10, 15, 20, 25, 30]  
l2 = [2, 4, 6, 8, 10, 12]  
print(" Python Original list 1: " + str(l1))  
print("Python Original list 2: " + str(l2))  
res\_lt = []  
for x in range(0, len(l1)):  
 res\_lt.append(l1[x] + l2[x])  
print(" Addition of the list lt1 and lt2 is: " + str(res\_lt))

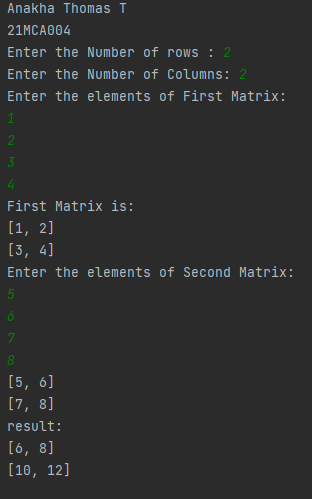
Output:



10. Write a program to find the sum of 2 matrices using nested List.

print("Anakha Thomas T")  
print("21MCA004")  
rows = int(input("Enter the Number of rows : "))  
column = int(input("Enter the Number of Columns: "))  
print("Enter the elements of First Matrix:")  
matrix\_a = [[int(input()) for i in range(column)] for i in range(rows)]  
print("First Matrix is: ")  
for n in matrix\_a:  
 print(n)  
print("Enter the elements of Second Matrix:")  
matrix\_b = [[int(input()) for i in range(column)] for i in range(rows)]  
for n in matrix\_b:  
 print(n)  
result = [[0 for i in range(column)] for i in range(rows)]  
for i in range(rows):  
 for j in range(column):  
 result[i][j] = matrix\_a[i][j] + matrix\_b[i][j]  
print("result: ")  
for r in result:  
 print(r)

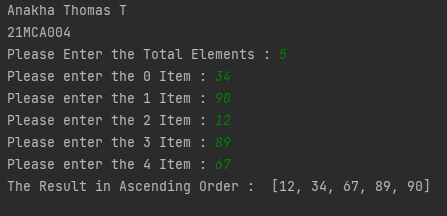
Output:



11. Write a program to perform bubble sort on a given set of elements.

print("Anakha Thomas T")  
print("21MCA004")  
a = []  
number = int(input("Please Enter the Total Elements : "))  
for i in range(number):  
 value = int(input("Please enter the %d Item : " %i))  
 a.append(value)  
  
for i in range(number -1):  
 for j in range(number - i - 1):  
 if(a[j] > a[j + 1]):  
 temp = a[j]  
 a[j] = a[j + 1]  
 a[j + 1] = temp  
print("The Result in Ascending Order : ", a)

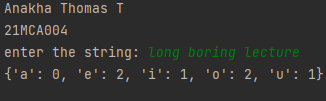
Output:



12. Program to find the count of each vowel in a string (use dictionary)

print("Anakha Thomas T")  
print("21MCA004")  
a= input("enter the string: ")  
a= a.casefold()  
count= {x:sum([1 for char in a if char == x]) for x in 'aeiou'}  
print(count)

Output:



13. Write a Python program that accept a positive number and subtract from this

Number the sum of its digits and so on. Continues this operation until the number is

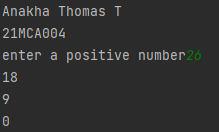
Positive (e.g.: 256->2+5+6=13

256-13=243

243-9=232……..

print("Anakha Thomas T")  
print("21MCA004")  
num=int(input("enter a positive number"))  
digsum=0  
new\_num=num  
while new\_num >= digsum:  
 list1 = [int(x) for x in str(new\_num)]  
 for i in list1:  
 digsum=digsum+i  
 new\_num=num-digsum  
 print(new\_num)  
print(new\_num-new\_num)

Output:

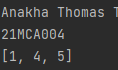


14. Write a Python program that accepts a 10 digit mobile number, and find the digits

Which are absent in a given mobile number.

print("Anakha Thomas T")  
print("21MCA004")  
def absent\_digits(n):  
 all\_nums = set([0,1,2,3,4,5,6,7,8,9])  
 n = set([int(i) for i in n])  
 n = n.symmetric\_difference(all\_nums)  
 n = sorted(n)  
 return n  
print(absent\_digits([9,8,3,2,2,0,9,7,6,3]))

Output:

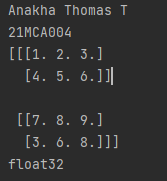


LAB CYCLE 2

1. Create a three dimensional array specifying float data type and print it.

print("Anakha Thomas T")  
print("21MCA004")  
import numpy as np  
arr = np.array([[[1,2,3],[4,5,6]],[[7,8,9],[3,6,8]]],dtype='f')  
print(arr)  
print(arr.dtype)

Output:



2. Create a 2 dimensional array (2X3) with elements belonging to complex data type and print it. Also display

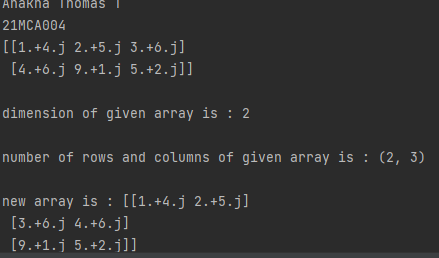
A. the no: of rows and columns

B. dimension of an array

C. reshape the same array to 3X2

print("Anakha Thomas T")  
print("21MCA004")  
import numpy as np  
arr = np.array([  
 [1+4j,2+5j,3+6j],  
 [4+6j,9+1j,5+2j],  
 ],  
 dtype=complex)  
print(arr)  
print("\ndimension of given array is :",arr.ndim)  
print("\nnumber of rows and columns of given array is :",arr.shape)  
newarr = arr.reshape(3,2)  
print("\n new array is :",newarr)

Output:



3. Familiarize with the functions to create

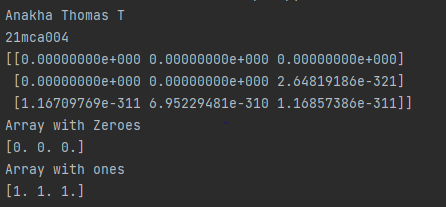
a) An uninitialized array

b) Array with all elements as 1,

c) All elements as 0

print("Anakha Thomas T")  
print("21mca004")  
import numpy as np  
myarr=np.empty((3, 3))  
print(myarr)  
print("Array with Zeroes")  
print(np.zeros(3))  
print("Array with ones")  
print(np.ones(3))

Output:



4. Create a one dimensional array using arrange function containing 10 elements.

Display

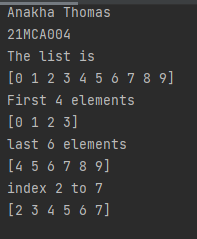
a. First 4 elements

b. Last 6 elements

c. Elements from index 2 to 7

print(("Anakha Thomas"))  
print("21MCA004")  
import numpy as np  
mylist=np.arange(start=0, stop=10)  
print("The list is")  
print(mylist)  
firstfour=mylist[0:4]  
print("First 4 elements")  
print(firstfour)  
lastsix=mylist[4:10]  
print("last 6 elements")  
print(lastsix)  
index=mylist[2:8]  
print("index 2 to 7")  
print(index)

Output:



5. Create a 1D array with arrange containing first 15 even numbers as elements

a. Elements from index 2 to 8 with step 2(also demonstrate the same Using slice function)

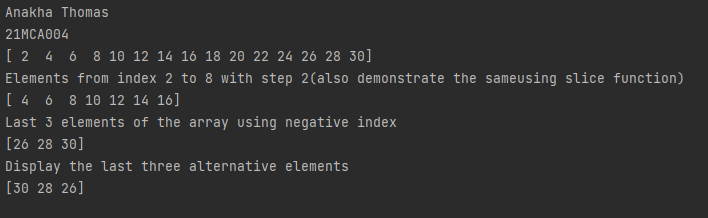
b. Last 3 elements of the array using negative index

c. Alternate elements of the array

d. Display the last 3 alternate elements

print("Anakha Thomas")  
print("21MCA004")  
import numpy as np  
num = np.arange(start=2, stop=32, step=2)  
print(num)  
print("Elements from index 2 to 8 with step 2(also demonstrate the same using slice function)")  
x = slice(1, 8)  
print(num[x])  
print("Last 3 elements of the array using negative index")  
print(num[-3:])  
print("Display the last three alternative elements")  
print(num[15:11:-1])

Output:



6. Create a 2 Dimensional array with 4 rows and 4 columns.

a. Display all elements excluding the first row.

b. Display all elements excluding the last column.

c. Display the elements of 1 st and 2 nd column in 2 nd and 3 rd row.

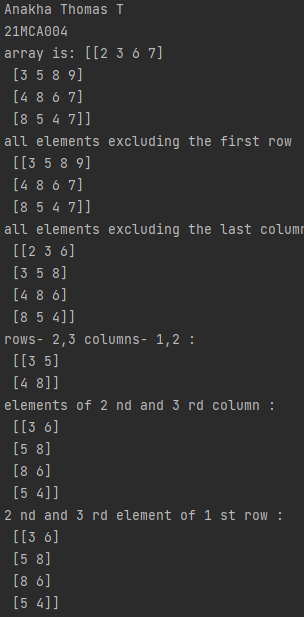
d. Display the elements of 2 nd and 3 rd column.

e. Display 2 nd and 3 rd. element of 1 st row.

f. Display the elements from indices 4 to 10 in descending order (use –values).

print("Anakha Thomas T")  
print("21MCA004")  
import numpy as np  
arr = np.array([[2,3,6,7],  
 [3,5,8,9],  
 [4,8,6,7],  
 [8,5,4,7]])  
print("array is:", arr)  
print("all elements excluding the first row\n", arr[1:4])  
print("all elements excluding the last column\n", arr[:,0:3])  
print("rows- 2,3 columns- 1,2 : \n", arr[1:3,0:2])  
print("elements of 2 nd and 3 rd column : \n", arr[:,1:3])  
print("2 nd and 3 rd element of 1 st row :\n", arr[0:,1:3])

Output:



7. Create two 2D arrays using array object and

a. Add the 2 matrices and print it

b. Subtract 2 matrices

c. Multiply the individual elements of matrix

d. Divide the elements of the matrices

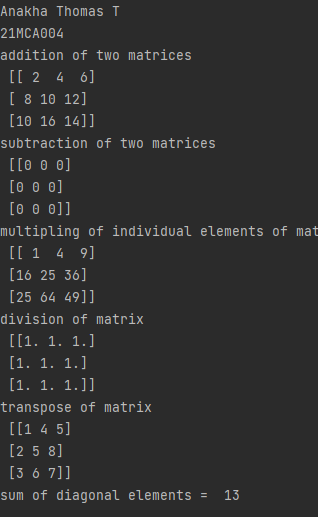
e. Perform matrix multiplication

f. Display transpose of the matrix

g. Sum of diagonal elements of a matrix

print("Anakha Thomas T")  
print("21MCA004")  
import numpy as n  
array1=n.array([[1,2,3],[4,5,6],[5,8,7]])  
array2=n.array([[1,2,3],[4,5,6],[5,8,7]])  
print("addition of two matrices\n",n.add(array1,array2))  
print("subtraction of two matrices\n",n.subtract(array1,array2))  
print("multipling of individual elements of matrix\n",n.multiply(array1,array2))  
print("division of matrix\n",n.divide(array1,array2))  
print("transpose of matrix\n",array1.T)  
print("sum of diagonal elements = ",sum(n.diag(array1)))

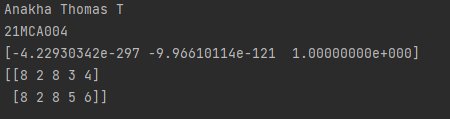
Output:



8. Demonstrate the use of insert () function in 1D and 2D array.

print("Anakha Thomas T")  
print("21MCA004")  
import numpy as np  
empt\_aray=np.empty([2])  
empt\_aray1=np.array([[2,3,4],[2,5,6]])  
print(np.insert(empt\_aray,2,[1]))  
print(np.insert(empt\_aray1,[0,1],8,axis=1))

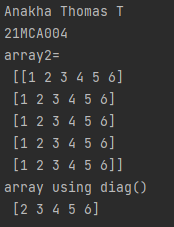
Output:



9. Demonstrate the use of diag () function in 1D and 2D array.

print("Anakha Thomas T")  
print("21MCA004")  
import numpy as np  
arr1=np.array([1,2,3,4,5,6])  
arr2=np.array([[1,2,3,4,5,6],[1,2,3,4,5,6],[1,2,3,4,5,6],[1,2,3,4,5,6],[1,2,3,4,5,6]])  
print("array2=\n",arr2)  
print("array using diag()\n",np.diag(arr2,k=1))

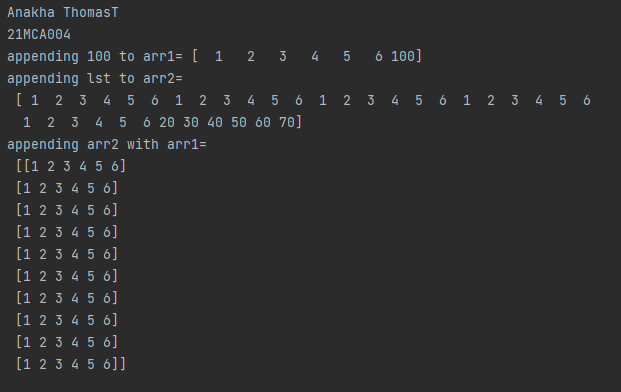
Output:



10. Demonstrate the use of append () function in 1D and 2D Array.

print("Anakha ThomasT")  
print("21MCA004")  
import numpy as np  
arr1=np.array([1,2,3,4,5,6])  
arr2=np.array([[1,2,3,4,5,6],  
 [1,2,3,4,5,6],  
 [1,2,3,4,5,6],  
 [1,2,3,4,5,6],  
 [1,2,3,4,5,6]])  
print("appending 100 to arr1=",np.append(arr1,100))  
lst=[20,30,40,50,60,70]  
print("appending lst to arr2=\n",np.append(arr2,lst))  
print("appending arr2 with arr1=\n",np.append(arr2,arr2,axis=0))

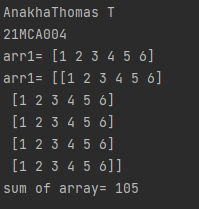
Output:



11. Demonstrate the use of sum () function in 1D and 2D array.

print("AnakhaThomas T")  
print("21MCA004")  
import numpy as np  
arr1=np.array([1,2,3,4,5,6])  
arr2=np.array([[1,2,3,4,5,6],  
 [1,2,3,4,5,6],  
 [1,2,3,4,5,6],  
 [1,2,3,4,5,6],  
 [1,2,3,4,5,6]])  
print("arr1=",arr1)  
print("arr1=",arr2)  
print("sum of array=",np.sum(arr2))

Output:



12. Create a square matrix with random integer values (use randint ()) and use

Appropriate functions to find:

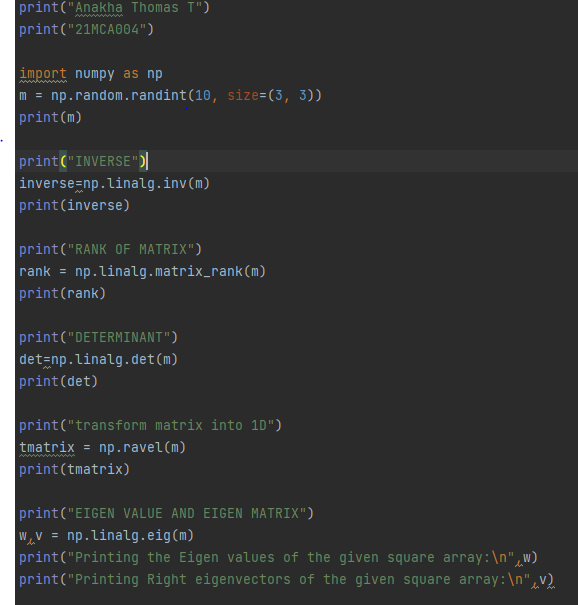
i) Inverse

ii) Rank of matrix

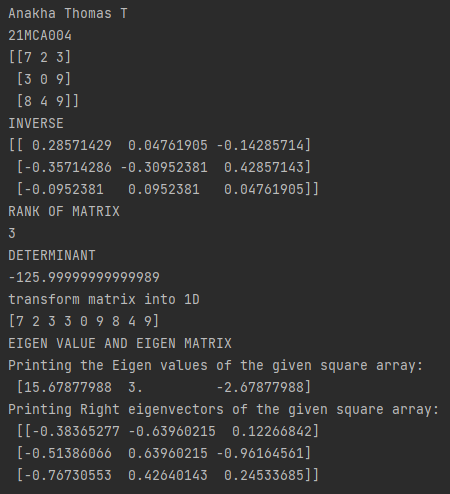
iii) Determinant

iv) Transform matrix into 1D array

v) Eigen values and vectors



Output:



13. Create a matrix X with suitable rows and columns

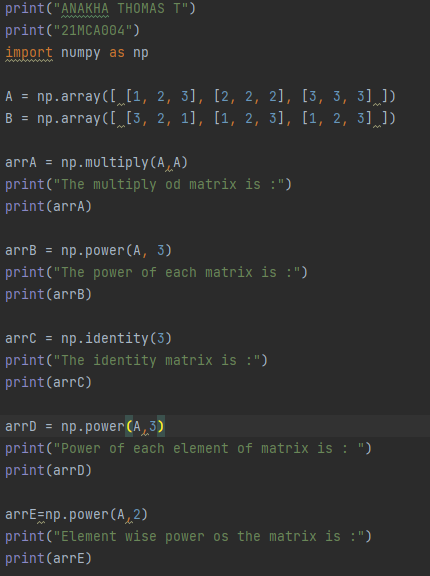
i) Display the cube of each element of the matrix using different methods

(Use multiply (), \*, power (),\*\*)

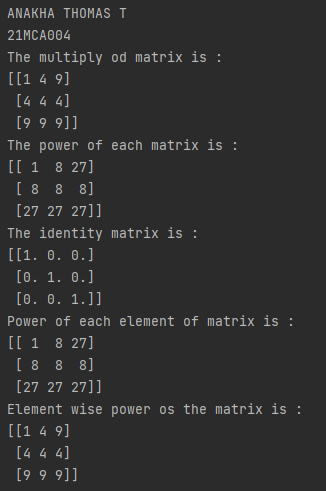
ii) Display identity matrix of the given square matrix.

iii) Display each element of the matrix to different powers.

iv) Create a matrix Y with same dimension as X and perform the operation X 2 +2Y

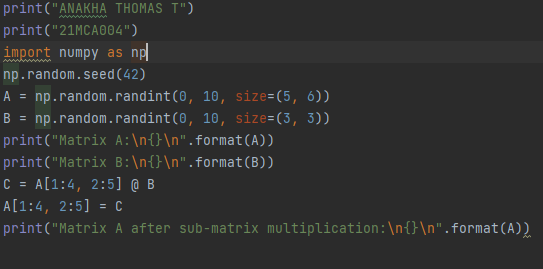


Output:

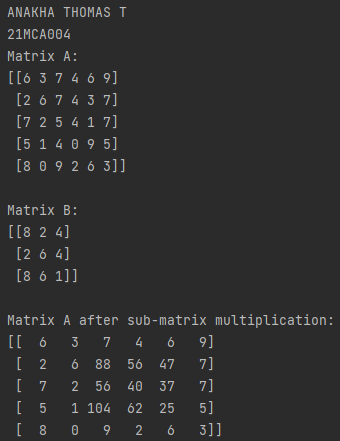


14 Multiply a matrix with a submatrix of another matrix and replace the same in larger

Matrix.

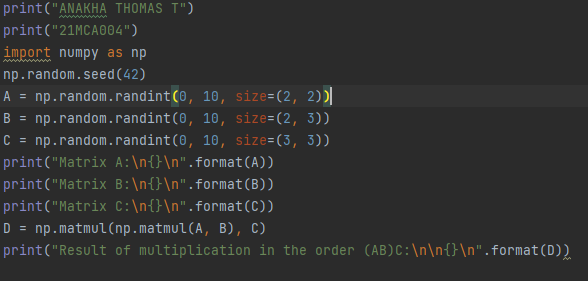


Output:

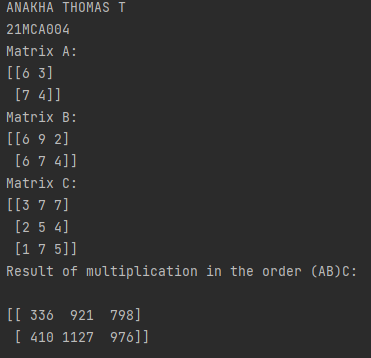


15. Given 3 Matrices A, B and C. write a program to perform matrix multiplication of

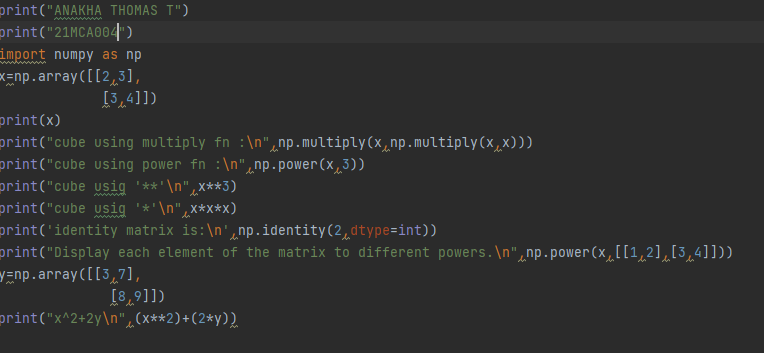
The 3 matrices.



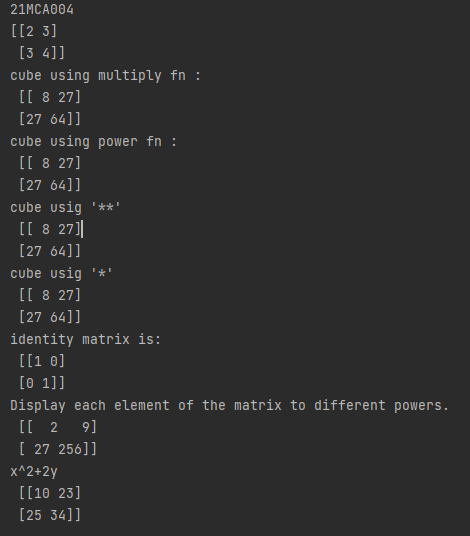
Output:



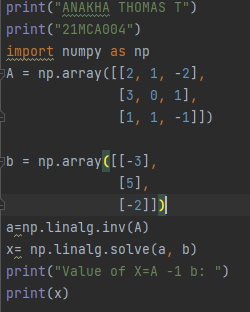
16. Write a program to check whether given matrix is symmetric or Skew Symmetric.



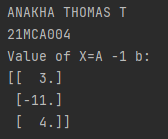
OUTPUT:



17. Write a program to find out the value of X using solve(), given A and b as above

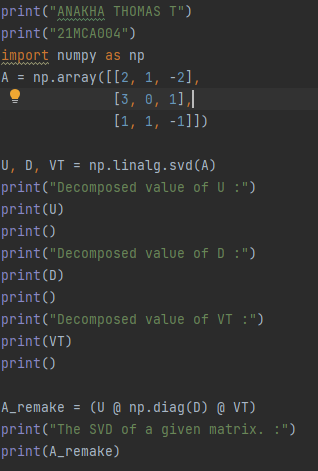


OUTPUT:

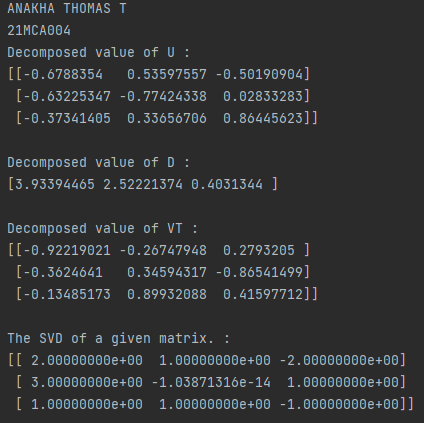


18. Write a program to perform the SVD of a given matrix. Also reconstruct the given matrix

From the 3 matrices obtained after performing SVD.



OUTPUT:



LAB CYCLE-3

1. Sarah bought a new car in 2001 for $24,000. The dollar value of her car changed each year as shown in the table below.

Value of Sarah’s Car

Year Value

2001 $24,000

2002 $22,500

2003 $19,700

2004 $17,500

2005 $14,500

2006 $10,000

2007 $ 5,800

Represent the following information using a line graph with following style properties

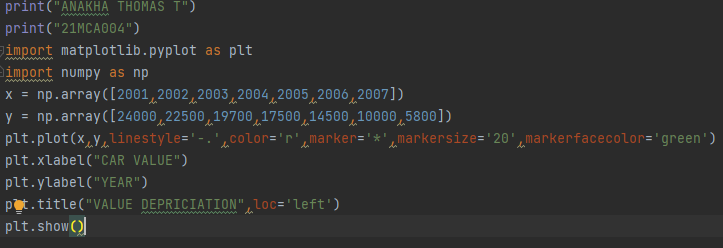
 X- axis - Year

Y –axis - Car Value

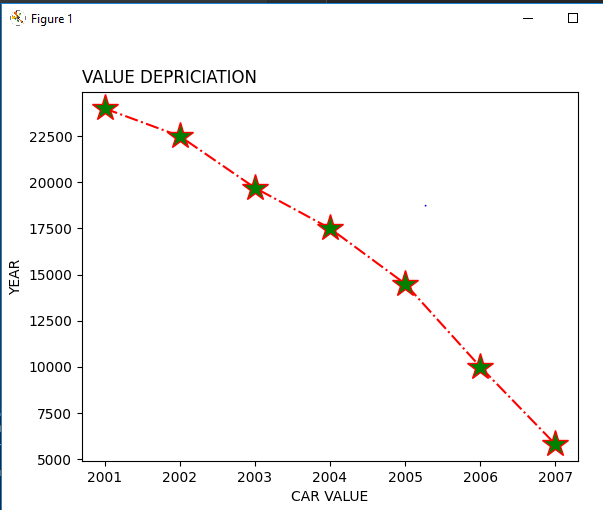
 title –Value Depreciation (left Aligned)

 Line Style dashdot and Line-color should be red

 Point using \* symbol with green color and size 20



OUTPUT:



2. Following table gives the daily sales of the following items in a shop

Day Mon Tues Wed Thurs Fri

Drinks 300 450 150 400 650

Food 400 500 350 300 500

Use subplot function to draw the line graphs with grids (color as blue and line style dotted) for the above information as 2 separate graphs in two rows

a) Properties for the Graph 1:

 X label- Days of week

 Y label-Sale of Drinks

 Title-Sales Data1 (right aligned)

 Line –dotted with cyan color

 Points- hexagon shape with color magenta and outline black

b) Properties for the Graph 2:

 X label- Days of Week

 Y label-Sale of Food

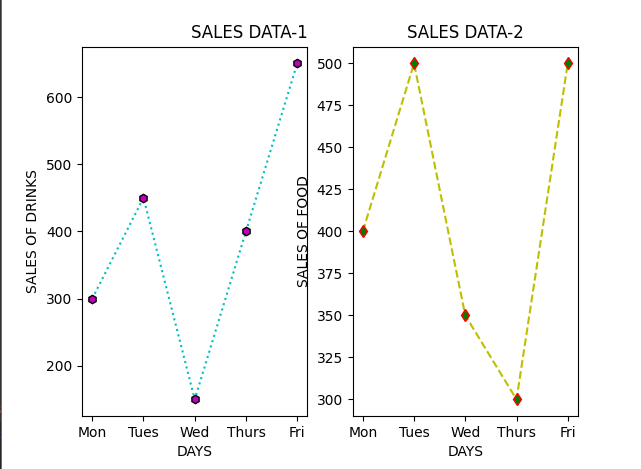
 Title-Sales Data2 (center aligned)

 Line –dashed with yellow color

 Points- diamond shape with color green and outline red



OUTPUT:



3. Create scatter plot for the below data :( use Scatter function)

PRODUCT: Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

AFFORDABLE: 173 153 195 147 120 144 148 109 174 130 172 131

LUXURY: 189 189 105 112 173 109 151 197 174 145 177 161

SUPER LUXURY: 185 185 126 134 196 153 112 133 200 145 167 110

Create scatter plot for each Segment with following properties within one graph

 X Label- Months of Year with font size 18

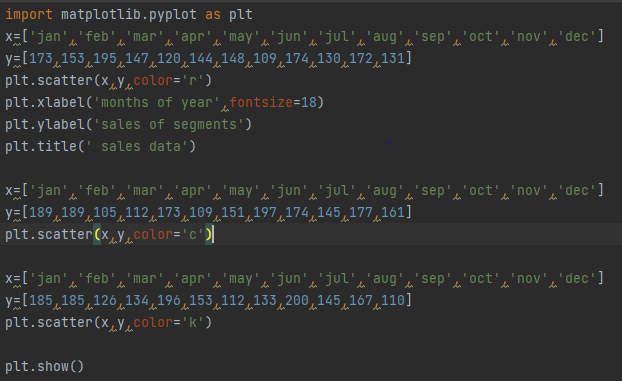
 Y-Label- Sales of Segments

 Title –Sales Data

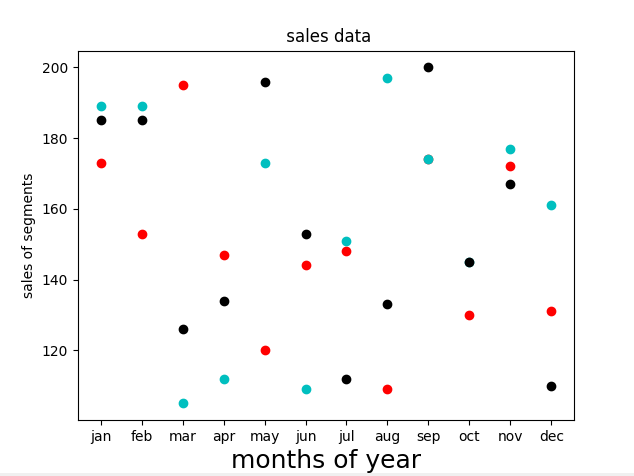
 Color for Affordable segment- pink

 Color for Luxury Segment- Yellow

 Color for Super luxury segment-blue



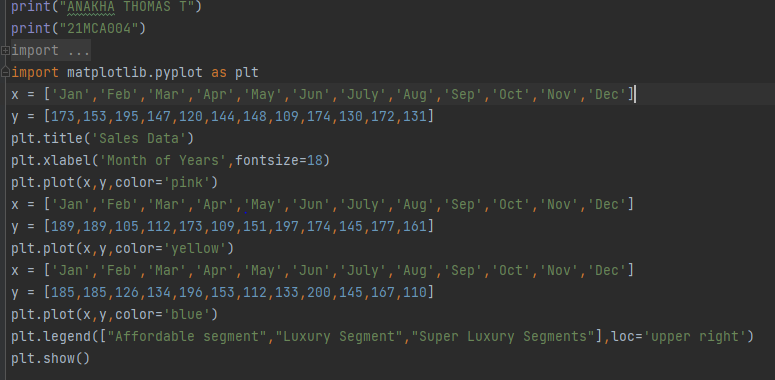
Output:



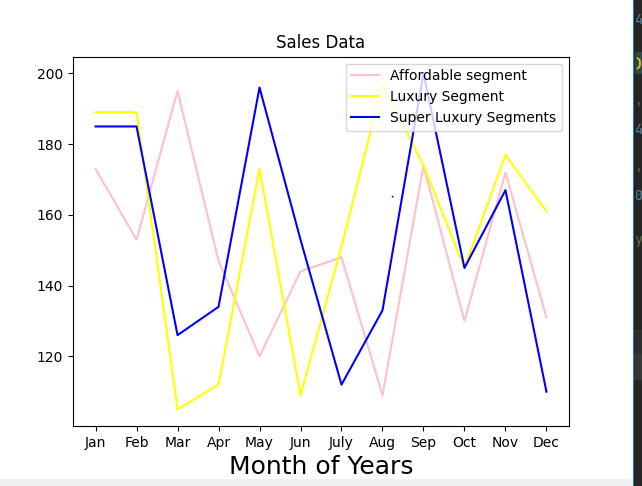
4. Display the above data using multiline plot (3 different lines in same graph)

 Display the description of the graph in upper right corner (use legend ())

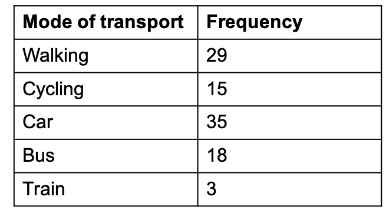
 Use different colors and line styles for 3 different lines



Output:



1. 100 Students were asked what their primary mode of transport for getting to school was. The results of this survey are recorded in the table below. Construct a bar graph representing this information.

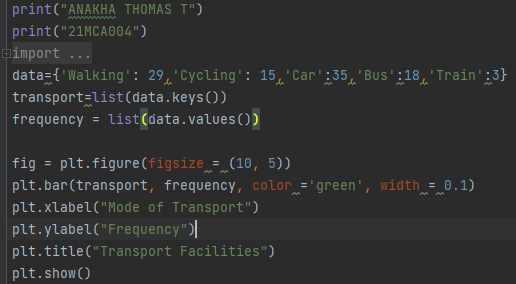


Create a bar graph with

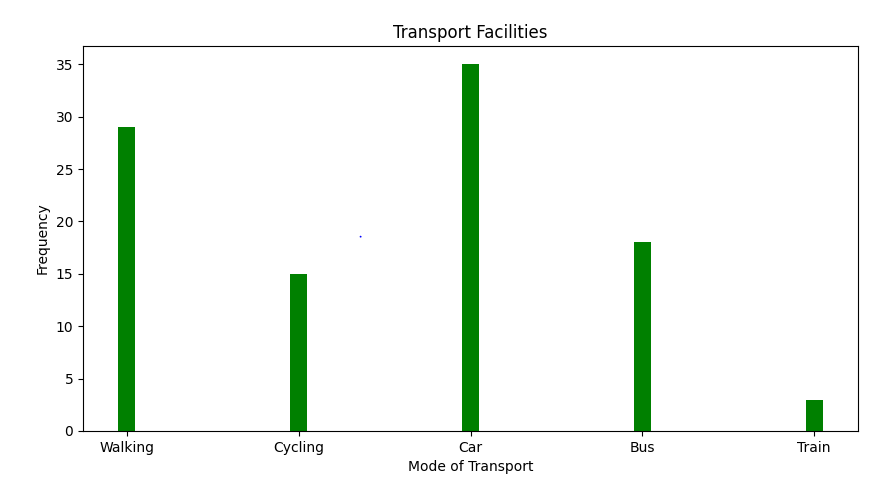
 X axis -mode of Transport and Y axis ‘frequency’

 Provide appropriate labels and title

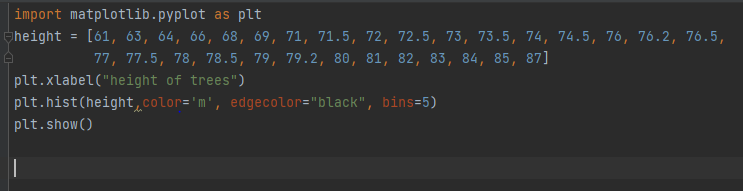
 Width .1, color green



Output:



1. 6. We are provided with the height of 30 cherry trees. The height of the trees (in inches): 61, 63, 64, 66, 68, 69, 71, 71.5, 72, 72.5, 73, 73.5, 74, 74.5, 76, 76.2,76.5, 77, 77.5, 78, 78.5, 79, 79.2, 80, 81, 82, 83, 84, 85, 87.Create a histogram with a bin size of 5



Output:

