import math

import random

import numpy as np

#Матриця

#1.1

print (' ')

a=np.matrix([[1, 2], [4, -1]])

print (' #1.1 ')

print (' A = ')

print(a)

print (' ')

b=np.matrix([[2,-3],[-4,1]])

print (' B = ')

print(b)

print (' ')

c = a\*b - b\*a

print (' C = A\*B - B\*A')

print(c)

print (' ')

#2.1

print (' #2.1 ')

z=np.matrix([[-1,2],[0,1]])

print (' Z = ')

print(z)

print (' ')

k = z\*\*2

print (' K = Z\*\*2')

print(k)

print (' ')

#3.1

print (' #3.1 ')

w=np.matrix([[3,5],[6,-1]])

print (' W = ')

print(w)

print (' ')

f=np.matrix([[2,1],[-3,2]])

print (' F = ')

print(f)

print (' ')

t = w\*f

print (' T = W\*F')

print(t)

print (' ')

#4.4

print (' #4.4 ')

m=np.matrix([[1,2,3],[-1,2,1],[1,3,2]])

print (' M = ')

print(m)

print (' ')

v = np.linalg.det(m)

print (' V = |M|')

print(v)

print (' ')

#5.1

print (' #5.1 ')

q=np.matrix([[1,2,3,4],[-2,1,-4,3],[3,-4,-1,2],[4,3,-2,-1]])

print (' Q = ')

print(q)

print (' ')

y = np.linalg.det(q)

print (' Y = |Q|')

print(y)

print (' ')

#6.3

print (' #6.3 ')

ck=np.matrix([[1,2,2],[2,1,-2],[2,-2,1]])

print (' CK = ')

print(ck)

print (' ')

tm=np.linalg.inv(ck)

print ('Обернена матриця CK=')

print(tm)

print (' ')

#7.1

print (' #7.1 ')

lt=np.matrix([[1,2,3,4],[3,-1,2,5],[1,2,3,4],[1,3,4,5]])

print (' LT = ')

print(lt)

print (' ')

rank = np.linalg.matrix\_rank(lt)

print (' RANK = ')

print(rank)

print (' ')

#8.3

#kramera

print (' #8.3 ')

A = np.array ([[3,-5,3], [1,2,1], [2,7,-1]])

B = np.array ([[1,-5,3], [4,2,1], [8,7,-1]])

C = np.array ([[3,1,3], [1,4,1], [2,8,-1]])

D = np.array ([[3,-5,1], [1,2,4], [2,7,8]])

print (' A = ')

print(A)

print (' B = ')

print(B)

print (' C = ')

print(C)

print (' D = ')

print(D)

det\_A=np.linalg.det(A)

print('det\_A=',det\_A)

det\_B=np.linalg.det(B)

print('det\_B=',det\_B)

det\_C=np.linalg.det(C)

print('det\_C=',det\_C)

det\_D=np.linalg.det(D)

print('det\_D=',det\_D)

x = det\_B/det\_A

y = det\_C/det\_A

z = det\_D/det\_A

print('x=',x)

print('y=',y)

print('z=',z)

#9.5

#матричний метод

print(' ')

print('#9.5')

mat1 = np.matrix([[4,1,4], [1,1,2], [2,1,2]])

mat2 = np.matrix([[-2], [-1], [0]])

mrt = np.linalg.inv(mat1)

result = mrt \* mat2

print(' ')

print(' mat1 = ')

print(mat1)

print(' ')

print(' mat2 = ')

print(mat2)

print(' ')

print(' result = ')

print(result)

print("revision = ")

print(np.linalg.solve(mat1, mat2))

#Розділ2

#Завдання 3

print (' Розділ 2 завдання 3 ')

print (' ')

#a[n][m]

n = 4

m = 5

#b = np.matrix[n,m]

b=np.matrix([[1,4,5,2,3],[4,0,-3,1,3],[4,-7,2,-1,8],[7,-2,0,5,-9]])

print (' b = ')

print(b)

print (' ')

i0 = 0

i1 = 0

i2 = 0

i3 = 0

j0 = 0

j1 = 0

j2 = 0

j3 = 0

j4 = 0

for i in range(n):

for j in range(m):

#for n:

if i == 0:

i0 += b[i,j]

si0 = i0/m

if i == 1:

i1 += b[i,j]

si1 = i1/m

if i == 2:

i2 += b[i,j]

si2 = i2/m

if i == 3:

i3 += b[i,j]

si3 = i3/m

#for m:

if j == 0:

j0 += b[i,j]

sj0 = j0/n

if j == 1:

j1 += b[i,j]

sj1 = j1/n

if j == 2:

j2 += b[i,j]

sj2 = j2/n

if j == 3:

j3 += b[i,j]

sj3 = j3/n

if j == 4:

j4 += b [i,j]

sj4 = j4/n

print (' ')

print (' середні значення по рядках')

print (si0)

print (si1)

print (si2)

print (si3)

print (' ')

print (' середні значення по стовпчиках ')

print (sj0)

print (sj1)

print (sj2)

print (sj3)

print (sj4)