НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені Ігоря Сікорського»

Факультет прикладної математики

Кафедра прикладної математики

Звіт

**ЕТАП №7**

«Розробка та перевірка програми»

із дисципліни «Програмування»

на тему

“Програма обернення матриці прямими методами (формула Крамера ) ”

|  |  |
| --- | --- |
| Виконала: | Керівник: |
| студентка групи КМ-02 | Олефір О.С |
| Гриб В.О. |  |

Київ — 2020

import sys

import numpy as np

import itertools

def prog():

def number\_permutations(n):

if n == 0:

return 1

else:

return n\*number\_permutations(n - 1)

def random\_matrix(dim):

"""

The function generates dim x dim array of integers

between 0 and 10.

"""

matrix = np.random.randint(10, size = (dim, dim))

return matrix

def permutations(n):

"""

The function generates all permutations

"""

num = ""

for i in range(n):

num += str(i+1)

t = list(itertools.permutations(num, n))

return t

def multiply(n, Matrix):

"""

The function count result of multiplication

"""

res = list(n)

a = 0

while res != [str(i + 1) for i in range(len(res))]:

for i in range(len(res) - 1):

if int(res[i]) > int(res[i + 1]):

temp = res[i + 1]

res[i + 1] = res[i]

res[i] = temp

a += 1

result = 1

for j in range(len(res)):

result \*= Matrix[j][int(n[j]) - 1]

result \*= ((-1) \*\* a)

return result

def sum(n, Matrix):

"""

The function count the sum

"""

result = 0

for i in n:

result += multiply(i, Matrix)

return result

def minor(matrix, n, x, y):

matrix1 = []

a = 0

for k in range(n):

if k == x:

continue

else:

matrix1.append([])

for m in range(n):

if m == y:

continue

else:

matrix1[a].append(matrix[k][m])

a += 1

return sum(permutations(n-1), matrix1)

matrix = []

j = int(input('Enter matrix size '))

q = input('Enter 1 if you want to create a random matrix\n Enter 0 if you want to create your own matrix ')

if q == '1':

matrix = random\_matrix(j)

else:

for k in range(j):

matrix.append([])

for n in range(j):

matrix[k].append(int(input(f'Enter {k+1}, {n+1} matrix element:')))

for k in matrix:

print(k)

det = sum(permutations(j), matrix)

res\_matrix = []

for k in range(j):

res\_matrix.append([])

for n in range(j):

if (k % 2 == 1 and n % 2 == 0) or (k % 2 == 0 and n % 2 == 1):

res\_matrix[k].append((-1)\*minor(matrix, j, k, n))

else:

res\_matrix[k].append(minor(matrix, j, k, n))

res\_matrix[k][n] /= det

res\_matrix[k][n] = round(res\_matrix[k][n], 3)

for el in res\_matrix:

print(el)

while True:

res = input('Запустити програму заново? ')

if res == 'Так':

prog()

else:

sys.exit(0)

prog()

