МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Кафедра систем управління літальними апаратами

Лабораторна робота № 3

з дисципліни «Об'єктно-орієнтоване програмування авіаційно-транспортних систем»

Тема: «"Структурування програм з використанням функцій""»

ХАІ.301.174.322.04 ЛР

Виконав студент гр. _____322_____

_____Зубов Євген Павлович_____
(підпис, дата) (П.І.Б.)

Перевірив
_____ к.т.н., доц. О. В. Гавриленко
_____ ас. В. О. Білозерський
(підпис, дата) (П.І.Б.)

2024

МЕТА РОБОТИ

Вивчити теоретичний матеріал із синтаксису визначення і виклику функцій та особливостей послідовностей у Python, а також документацію бібліотеки numpy; отримати навички реалізації бібліотеки функцій з параметрами, що структурують вирішення завдань «згори – до низу».

ПОСТАНОВКА ЗАДАЧІ

Завдання 1.

Описати функцію відповідно до варіанту. Для виклику функції (друга частина задачі) описати іншу функцію, що на вході має список вхідних даних і повертає список вихідних даних. Введення даних, виклик функції та виведення результатів реалізувати в третій функції без параметрів.

Proc21.Описати функцію Calc (A, B, Op) дійсного типу, що виконує над ненульовими дійсними числами A та B одну з арифметичних операцій і повертає її результат. Вид операції визначається цілим параметром Op: 1 - віднімання, 2 - множення, 3 - розподіл, інші значення - складання. За допомогою Calc виконати для даних A і B операції, які визначаються даними цілими N1, N2, N3.

Завдання 2.

Розробити дві вкладені функції для вирішення задачі обробки двовимірних масивів відповідно до варіанту: зовнішня – без параметрів, внутрішня має на вході ім’я файлу з даними, на виході – підраховані параметри матриці (перша частина задачі) та перетворену матрицю (друга частина задачі). Для обробки масивів використати функції бібліотеки numpy.

Matrix 14. У текстовому файлі задана матриця розміру M × N. У кожному її стовпці знайти кількість елементів, великих середнього арифметичного всіх елементів цього стовпчика. Знайти поелементний добуток заданої матриці з матрицею того ж розміру, заповненої випадковими числами.

ВИКОНАННЯ РОБОТИ

Завдання 1. Вирішення задачі Func21:

Вхідні дані (ім’я, опис, тип, обмеження):

A,B – дійсні числа, float

Op – оператор, int

Вихідні дані (ім’я, опис, тип):

Вивід результату у консолі

Алгоритм вирішення показано нижче чи показано на рис. 1.1 Лістинг коду вирішення задачі наведено в дод. А.1 (стор. 6).

Екран роботи програми показаний на рис. Б.1.

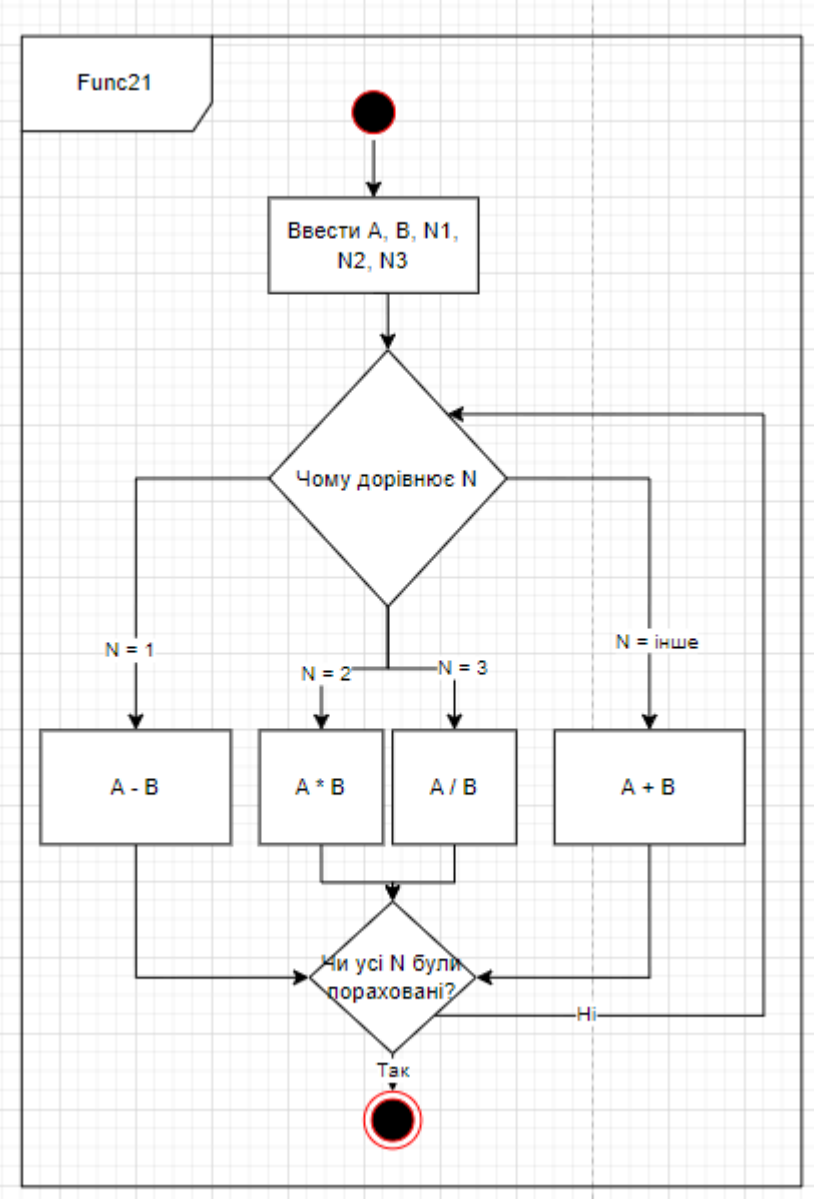

Рисунок 1.1 – Алгоритм вирішення задачі Func21

Завдання 2. Вирішення задачі Matrix14:
Вхідні дані (ім’я, опис, тип, обмеження):
M,N - розміри масиву, int
Вихідні дані (ім’я, опис, тип):
Вивід результату у консолі

Лістинг коду вирішення задачі наведено в дод. А.2 (стор. 6).
Екран роботи програми показаний на рис. Б.2.

ВИСНОВКИ

1. **Функції у Python**:
   Визначаються через `def`, викликаються за іменем. Аргументи можуть бути позиційними чи іменованими.
2. **Послідовності**:
   Це списки, кортежі, рядки тощо. Підтримують індексацію, зрізи, ітерації.
3. **NumPy**:
   Бібліотека для роботи з масивами та математики.
4. **Підхід «згори – до низу»**:
   Розбиває задачу на частини, кожна реалізується окремою функцією. Це зручно та зрозуміло.

ДОДАТОК А.1
Лістинг головного меню завдання

```
import numpy as np
import random

def proc21():
    A = float(input("Введіть A: "))
    B = float(input("Введіть B: "))
    N1 = int(input("Введіть N1: "))
    N2 = int(input("Введіть N2: "))
    N3 = int(input("Введіть N3: "))
    if A == 0 or B == 0:
        print("Error: A and B must be non-zero numbers.")
        return

    print("Завдання№1 = ",Calc(A,B,N1))
    print("Завдання№2 = ", Calc(A, B, N2))
    print("Завдання№3 = ", Calc(A, B, N3))




def Calc(A,B,Op):
    if Op == 1:
        return A-B
    elif Op == 2:
        return A*B
    elif Op == 3:
        return A/B
    else:
        return A+B


def matrix14():
    filename = ("input.txt")
    with open(filename) as f:
        mean(f)
        f.seek(0)
        rand(f)


def mean(f):
    rule = f.readline().split(" ")
    M = int(rule[0])
    N = int(rule[1])

    count = 0
    ariph = 0
    bigger_one = 0
    bigger_list = []
    big_list = []
    ariph_list = []

    arr = np.loadtxt(f, max_rows=M)
    print(arr)
    for i in range (0,M):
        for j in range (0,N):
            ariph += arr[i][j]


        ariph /= N
        ariph_list.append(ariph.item())
        ariph = 0
```

```
        for j in range (0,N):
            if arr[i][j] > ariph_list[i]:
                bigger_one = arr[i][j]
                bigger_list.append(bigger_one.item())
                bigger_one = 0
                count += 1

        print("array№",[i]," = ",bigger_list, "\n Кількість чисел більших за с.а.
ряду: ", count)
        count = 0
        bigger_list = []


    print("Mean of ariph:", ariph_list)



def rand(f):
    rule = f.readline().split(" ")
    M = int(rule[0])
    N = int(rule[1])

    arr = np.loadtxt(f, max_rows=M)

    default = np.zeros((M,N))

    for i in range(M):
        for j in range(N):
            default[i][j] = random.randint(0, 20)

    print("Дефолт рандом = \n", default)


    for i in range(M):
        for j in range(N):
            default[i][j] *= arr[i][j]


    print("Дефолт рандом * Массив = \n", default)
    print("Массив: \n", arr)
```

ДОДАТОК Б

Скрін-шоти вікна виконання програми

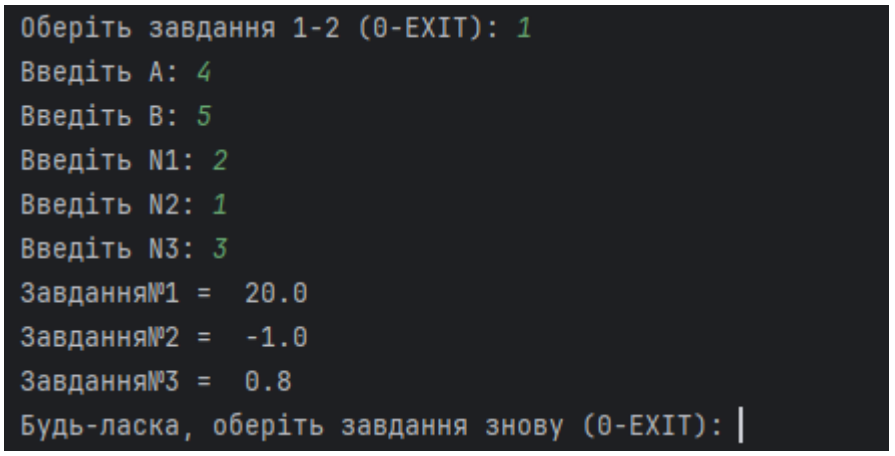

Рисунок Б.1 – Екран виконання програми для вирішення завдання Func21

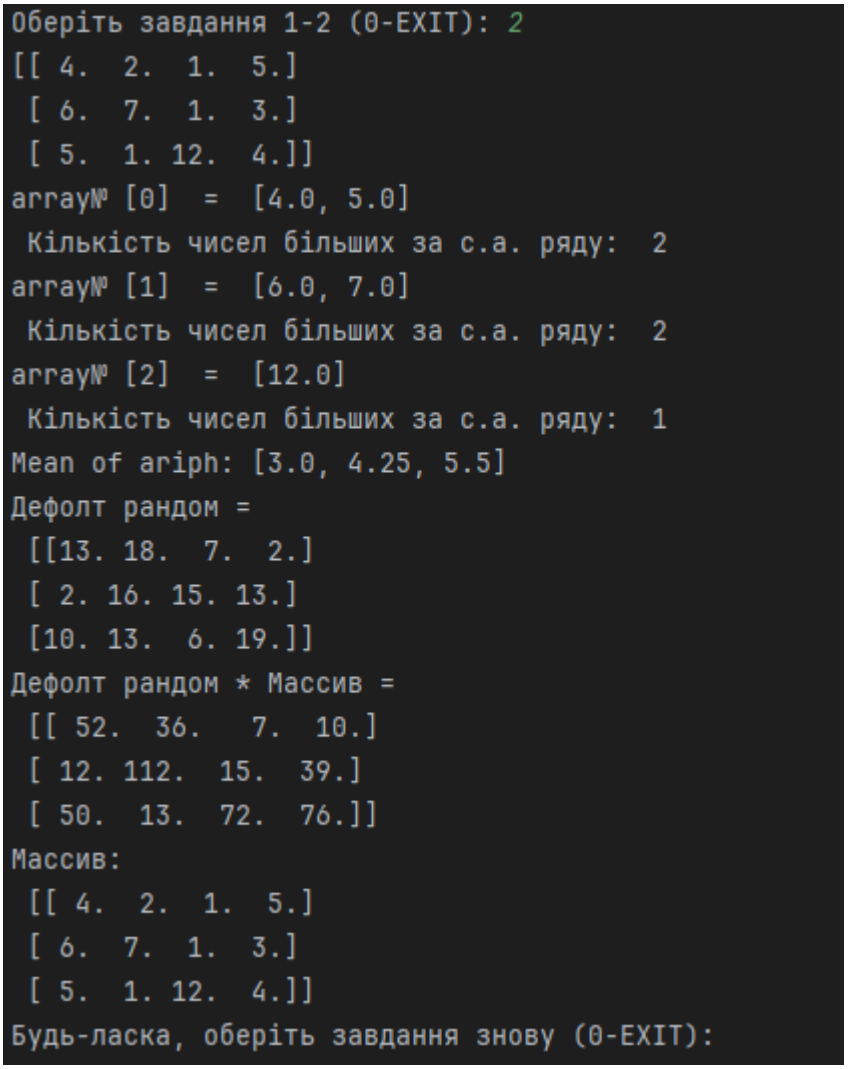

Рисунок Б.2 – Екран виконання програми для вирішення завдання Matrix14