

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 проектування СУ»

Тема: «Структурування програм з використанням
функцій»

ХАІ.301 . 3.320.3 ЛР

Виконав студент гр. 320

_____ Семеняга Ігор____
(підпис, дата) (П.І.Б.)

Перевірив

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

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.

Описати функцію `Minmax (X, Y)`, що записує в змінну `X` мінімальне зі значень `X` і `Y`, а в змінну `Y` - максимальне з цих значень (`X` і `Y` – речові параметри, які є одночасно вхідними та вихідними). Використовуючи чотири виклики цієї функції, знайти мінімальне і максимальне з даних чисел `A`, `B`, `C`, `D`.

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).

У текстовому файлі задана матриця розміру $M \times N$. Для кожного стовпця матриці знайти суму і добуток його елементів. Знайти суму заданої матриці з матрицею того ж розміру, заповненої випадковими числами.

ВИКОНАННЯ РОБОТИ

Завдання 1. Вирішення задачі 1, №11

Описати функцію Minmax (X, Y), що записує в змінну X мінімальне зі значень X і Y, а в змінну Y - максимальне з цих значень (X і Y - речові параметри, які є одночасно вхідними та вихідними). Використовуючи чотири виклики цієї функції, знайти мінімальне і максимальне з даних чисел A, B, C, D.

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

a = float(input("A: ")) float

b = float(input("B: ")) float

c = float(input("C: ")) float

d = float(input("D: ")) float

Вихідні дані (ім'я, опис, тип):

print("Мінімум: {min_val}, Максимум: {max_val}") float

Алгоритм вирішення

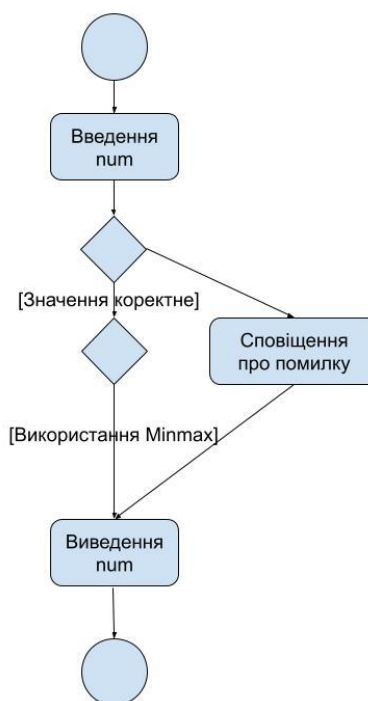


Рисунок 1 – Алгоритм вирішення завдання 1

Лістинг коду вирішення задачі наведено в дод. А (стор. 6). Екран роботи програми показаний на рис. Б.7.

Завдання 2. Вирішення задачі 2, №4

У текстовому файлі задана матриця розміру $M \times N$. Для кожного стовпця матриці знайти суму і добуток його елементів. Знайти суму заданої матриці з матрицею того ж розміру, заповненої випадковими числами

Вхідні дані:

```
file_name = input("Введіть ім'я файлу з матрицею .txt: ") txt file with int numbers
```

Вихідні дані:

```
print("Сума елементів кожного стовпця:", column_sums) int
print("Добуток елементів кожного стовпця:", column_products) int
print("Перетворена матриця:\n", transformed_matrix) int
```

Алгоритм вирішення

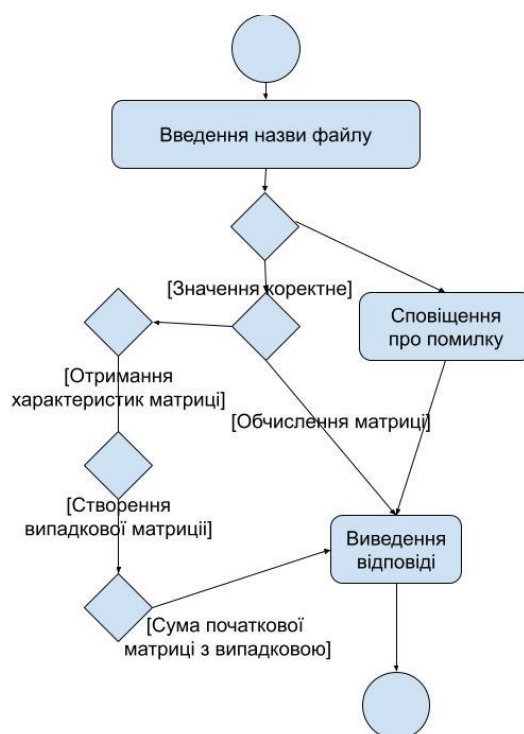


Рисунок 2 – Алгоритм вирішення завдання 2

ВИСНОВКИ

Було вивчено теоретичний матеріал із синтаксису визначення та виклику функцій у Python, а також особливості роботи з послідовностями. Ознайомлено з документацією бібліотеки NumPy для роботи з багатовимірними масивами та числовими обчисленнями. На практиці отримано навички створення бібліотеки функцій із параметрами, що забезпечують структурування вирішення завдань за підходом «згори – до низу».

ДОДАТОК А

Лістинг коду програми до задач 1-№11, 2-№4

```

from functions import minmax, find_minmax, process_matrix
import numpy as np

def main_menu():
    while True:
        print("\nМеню:")
        print("1. Завдання 1: Знайти мінімум і максимум з чотирьох чисел")
        print("2. Завдання 2: Периметр і площа трикутників")
        print("3. Вихід")

        choice = input("Оберіть опцію (1-3): ")

        if choice == "1":
            # Завдання 1
            def process_minmax(data):
                """
                Функція для пошуку мінімального і максимального значень серед
                чотирьох чисел.

                Args:
                    data (list): Список чисел [A, B, C, D].

                Returns:
                    tuple: Мінімальне та максимальне значення.
                """
                A, B, C, D = data
                A, B = minmax(A, B)
                C, D = minmax(C, D)
                A, C = minmax(A, C)
                B, D = minmax(B, D)
                return A, D # Мінімальне і максимальне значення

            def main_minmax():
                """
                Основна функція для введення даних, виклику minmax і виведення
                результату.
                """
                data = list(map(float, input("Введіть чотири числа через пробіл
                (A B C D): ").split()))
                if len(data) != 4:
                    print("Помилка: потрібно ввести рівно чотири числа!")
                    return

                minimum, maximum = process_minmax(data)
                print(f"Мінімальне значення: {minimum}")
                print(f"Максимальне значення: {maximum}")

            main_minmax()

```

```

elif choice == "2":

    def process_matrix():
        """
        Зовнішня функція без параметрів, яка викликає внутрішню функцію
        для обробки матриці.
        """
        def inner_function(file_name):

            # Внутрішня функція для обробки двовимірної матриці.

            # Зчитування матриці з текстового файлу
            matrix = np.loadtxt(file_name, delimiter=' ')

            # Обчислення суми та добутку елементів кожного стовпця
            column_sums = matrix.sum(axis=0)
            column_products = matrix.prod(axis=0)

            # Генерація матриці такого ж розміру з випадковими числами
            random_matrix = np.random.randint(1, 10, size=matrix.shape)

            # Сума початкової матриці з випадковою
            transformed_matrix = matrix + random_matrix

            return (column_sums, column_products), transformed_matrix

        # Введення імені файлу та виклик внутрішньої функції
        file_name = input("Введіть ім'я файлу з матрицею: ")
        try:
            (column_sums, column_products), transformed_matrix =
inner_function(file_name)

            # Вивід результатів
            print("Сума елементів кожного стовпця:", column_sums)
            print("Добуток елементів кожного стовпця:", column_products)
            print("Сума початкової матриці з випадковою:\n",
transformed_matrix)
        except FileNotFoundError:
            print("Файл не знайдено. Будь ласка, перевірте ім'я файлу.")
        except ValueError:
            print("Файл має неправильний формат або не містить числових
даних.")

    # Виклик зовнішньої функції
    process_matrix()

elif choice == "3":
    print("Вихід з програми.")
    break

```

```
        else:
            print("Неправильний вибір. Спробуйте ще раз.")

if __name__ == "__main__":
    main_menu()
```

ДОДАТОК Б

Скрін-шоти вікна виконання програми

```
Меню:  
1. Завдання 1: Знайти мінімум і максимум з чотирьох чисел  
2. Завдання 2: Периметр і площа трикутників  
3. Вихід  
Оберіть опцію (1-3): 1  
Введіть чотири числа через пробіл (A B C D): -5 3 4 6  
Мінімальне значення: -5.0  
Максимальне значення: 6.0
```

Рисунок Б.1 – Екран виконання програми для вирішення завдання

1

```
Меню:  
1. Завдання 1: Знайти мінімум і максимум з чотирьох чисел  
2. Завдання 2: Периметр і площа трикутників  
3. Вихід  
Оберіть опцію (1-3): 2  
Введіть ім'я файлу з матрицею .txt: matrix.txt  
Сума елементів кожного стовпця: [12. 15. 18.]  
Добуток елементів кожного стовпця: [ 28. 80. 162.]  
Сума початкової матриці з випадковою:  
[[ 4. 5. 7.]  
[12. 8. 13.]  
[12. 15. 14.]]
```

Рисунок Б.2 – Екран виконання програми для вирішення завдання

2