Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій

Кафедра ІСМ



**Звіт**

до лабораторної роботи № 1

з дисципліни

​*Спеціалізовані мови програмування*

на тему:

“**Введення в Python** ”

Виконав:  
 студент РІ-31

Яворський О. В.

Прийняв:  
к.т.н., доцент  
 Щербак С.С.

Львів – 2024

**Мета роботи:** створення консольної програми-калькулятора за допомогою основних синтаксичних конструкцій Python, з іншим завданням на заміну тестуванню та валідації:

**Завдання лабораторної роботи**

Завдання 1: Введення користувача Створіть Python-програму, яка приймає введення користувача для двох чисел і оператора (наприклад, +, -, \*, /).

Завдання 2: Перевірка оператора Перевірте чи введений оператор є дійсним (тобто одним із +, -, \*, /). Якщо ні, відобразіть повідомлення про помилку і попросіть користувача ввести дійсний оператор. Завдання 3: Обчислення Виконайте обчислення на основі введення користувача (наприклад, додавання, віднімання, множення, ділення) і відобразіть результат.

Завдання 4: Повторення обчислень Запитайте користувача, чи він хоче виконати ще одне обчислення. Якщо так, дозвольте йому ввести нові числа і оператор. Якщо ні, вийдіть з програми.

Завдання 5: Обробка помилок Реалізуйте обробку помилок для обробки ділення на нуль або інших потенційних помилок. Відобразіть відповідне повідомлення про помилку, якщо виникає помилка. Завдання 6: Десяткові числа Змініть калькулятор так, щоб він обробляв десяткові числа (плаваючу кому) для більш точних обчислень.

Завдання 7: Додаткові операції Додайте підтримку додаткових операцій, таких як піднесення до степеня (^), квадратний корінь (√) і залишок від ділення (%).

Завдання 8: Функція пам'яті Реалізуйте функцію пам'яті, яка дозволяє користувачам зберігати і відновлювати результати. Додайте можливості для зберігання та отримання значень з пам'яті.

Завдання 9: Історія обчислень Створіть журнал, який зберігає історію попередніх обчислень, включаючи вираз і результат. Дозвольте користувачам переглядати історію своїх обчислень.

Завдання 10: Налаштування користувача Надайте користувачам можливість налаштувати поведінку калькулятора, таку як зміну кількості десяткових розрядів, які відображаються, або налаштування функцій пам'яті.

from calculator\_function import get\_number, get\_operator, calculate, save\_to\_memory, recall\_memory, show\_history, add\_to\_history

from constants import global\_var\_calculator

def calculator():

    while True:

        num1 = get\_number("Введіть перше число (або 'm' для значення з пам'яті): ")

        operator = get\_operator()

        if operator != '√':

            num2 = get\_number("Введіть друге число (або 'm' для значення з пам'яті): ")

        else:

            num2 = None

        result = calculate(num1, num2, operator)

        if result is not None:

            print(f"Результат: {result}")

            add\_to\_history(num1, num2, operator, result)

        save = input("Зберегти результат в пам'ять? (y/n): ")

        if save.lower() == 'y':

            save\_to\_memory(result)

        again = input("Виконати ще одне обчислення? (y/n): ")

        if again.lower() != 'y':

            break

        show\_hist = input("Показати історію обчислень? (y/n): ")

        if show\_hist.lower() == 'y':

            show\_history()

if \_\_name\_\_ == "\_\_main\_\_":

    print("Ласкаво просимо до калькулятора!")

    calculator()

    print("Програма завершена")

**Висновок**

Виконавши ці завдання, я створив простий консольний калькулятор на Python, який може виконувати арифметичні операції, обробляти помилки та надавати користувачу зручний інтерфейс. Цей проект допоміг мені вивчити основний синтаксис Python і концепції, такі як введення користувача, умовні оператори, цикли та обробка помилок.