Міністерство освіти і науки України

Національний університет «Львівська політехніка»



**ЗВІТ**

про виконання лабораторної роботи № 1

з курсу:

«Спеціалізовані мови програмування»

**Виконала:**

студент гр. ІТ-31

Сейтумеров ЕНВЕР

**Прийняв:**

Сергій ЩЕРБАК

Львів 2023

**Тема:** Введення в Python

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

**Хід 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.

import math

# Class to track user calculations

class CalculationTracker:

    def \_\_init\_\_(self):

        self.history = []

    def add\_record(self, expr, res):

        self.history.append((expr, res))

    def get\_history(self):

        return self.history

    def reset\_history(self):

        self.history = []

def perform\_calculation(a, b, op):

    operations = {

        '+': lambda: a + b,

        '-': lambda: a - b,

        '\*': lambda: a \* b,

        '/': lambda: a / b if b != 0 else "Error: Division by zero",

        '^': lambda: a \*\* b,

        '√': lambda: math.sqrt(a) if a >= 0 else "Error: Square root of negative number",

        '%': lambda: a % b if b != 0 else "Error: Modulo by zero"

    }

    func = operations.get(op)

    if not func:

        return "Error: Unknown operator. Please try again."

    try:

        return func()

    except ZeroDivisionError:

        return "Error: Division by zero"

    except ValueError:

        return "Error: Invalid input"

calc\_tracker = CalculationTracker()

while True:

    try:

        first\_num = float(input("Enter the first number: "))

        second\_num = float(input("Enter the second number: "))

    except ValueError:

        print("Invalid input. Please enter a number.")

        continue

    while True:

        operation = input("Operator (+, -, \*, /, ^, √, %): ")

        if operation in ['+', '-', '\*', '/', '^', '√', '%']:

            outcome = perform\_calculation(first\_num, second\_num, operation)

            print(f"Result: {outcome}")

            calc\_tracker.add\_record(f"{first\_num} {operation} {second\_num}", outcome)

            break

        else:

            print("Invalid operator. Choose from +, -, \*, /, ^, √, %.")

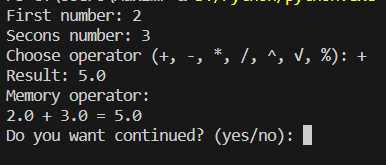
    print("Calculation History:")

    for expr, res in calc\_tracker.get\_history():

        print(f"{expr} = {res}")

    if input("Continue? (yes/no): ").lower() != 'yes':

        break



*Рис.1 – Результат виконання програми*

**Висновок:** на даній лабораторній роботі я створив простий консольний калькулятор на Python, який може виконувати арифметичні операції, обробляти помилки та надавати користувачу зручний інтерфейс. Цей проект допоміг вивчити основний синтаксис Python і концепції, такі як введення користувача, умовні оператори, цикли та обробка помилок.