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

Полапа МАКСИМ

**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 memory to save inf about users

class Memory:

    def \_\_init\_\_(self): #init func to memory storage

        self.memory = []

    def store(self, expression, result): #add func to memory storage

        self.memory.append((expression, result))

    def recall(self): #return inf from starage

        return self.memory

    def clear(self): #clear func to storage

        self.memory = []

def calculate(num1, num2, operator):

    operator\_functions = {

        '+': lambda: num1 + num2,

        '-': lambda: num1 - num2,

        '\*': lambda: num1 \* num2,

        '/': lambda: num1 / num2 if num2 != 0 else "Error: division on zero",

        '^': lambda: num1 \*\* num2,

        '√': lambda: math.sqrt(num1) if num1 >= 0 else "Error: negative number under the root",

        '%': lambda: num1 % num2 if num2 != 0 else "Error: division on zero"

    }

    operation = operator\_functions.get(operator) #get operator

    if operation is None:

        return "Error: missing operator, try again"

    try:

        return operation()

    except ZeroDivisionError:

        return "Error: division by zero"

    except ValueError:

        return "Error: invalid input"

memory = Memory()

while True:

    try:

        num1 = float(input("First number: ")) #first number

        num2 = float(input("Secons number: ")) #second number

    except ValueError:

        print("Error: it`s wrote not number, try again")

        continue

    while True:

        operator = input("Choose operator (+, -, \*, /, ^, √, %): ") #operator

        if operator in ['+', '-', '\*', '/', '^', '√', '%']: #find which operator was change

            result = calculate(num1, num2, operator) #variable result

            print(f"Result: {result}")

            memory.store(f"{num1} {operator} {num2}", result)

            break

        else:

            print("Error: bad operator. Please, choose on of this operator '+, -, \*, /, ^, √, %' ")

    memory\_values = memory.recall() #return inf from storage memory

    print(f"Memory operator:")

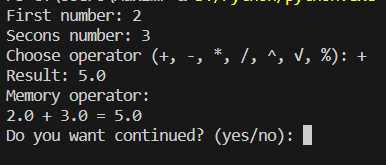
    for expression, result in memory\_values:

        print(f"{expression} = {result}")

    repeat = input("Do you want continued? (yes/no): ")

    if repeat.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.