

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра інформаційних систем та мереж



Звіт

Про виконання лабораторної роботи №1
з дисципліни «Спеціалізовані мови програмування»
на тему «Введення в Python»

Виконала:

Студентка гр. РІ-21

Зузяк Л. Р.

Прийняв:

Щербак С.С.

Львів 2024

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

План 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.

Виконання:

```
from calculator.operations import add, subtract, multiply, divide, exponentiate,
sqrt, modulo
from calculator.memory import Memory
from calculator.history import History
from calculator.settings import Settings

def get_user_input():
    while True:
        try:
            num1 = float(input("Enter the first number: "))
            operator = input("Enter an operator (+, -, *, /, ^, √, %): ")
            num2 = None
            if operator not in ('√',): # √ only needs one number
                num2 = float(input("Enter the second number: "))
```

```

        return num1, operator, num2
    except ValueError:
        print("Invalid input. Please enter numbers correctly.")

def calculate(num1, operator, num2):
    if operator == '+':
        return add(num1, num2)
    elif operator == '-':
        return subtract(num1, num2)
    elif operator == '*':
        return multiply(num1, num2)
    elif operator == '/':
        return divide(num1, num2)
    elif operator == '^':
        return exponentiate(num1, num2)
    elif operator == '√':
        return sqrt(num1)
    elif operator == '%':
        return modulo(num1, num2)
    else:
        print("Invalid operator")
        return None

def main():
    history = History()
    memory = Memory()
    settings = Settings()

    while True:
        num1, operator, num2 = get_user_input()
        result = calculate(num1, operator, num2)

        if result is not None:
            result = settings.round_result(result)
            print(f"Result: {result}")
            history.add_entry(num1, operator, num2, result)

        save_memory = input("Do you want to save this result to memory? (y/n): ")
        if save_memory.lower() == 'y' and result is not None:
            memory.store(result)

        view_history = input("View calculation history? (y/n): ")
        if view_history.lower() == 'y':
            history.show()

        new_calc = input("Would you like to perform another calculation? (y/n): ")
        if new_calc.lower() != 'y':

```

```
        print("Goodbye!")
        break

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

Результати виконаної роботи в GitHub:

https://github.com/Lilia427/Lab_1_python-calculator/tree/main

Висновки. В ході виконання лабораторної роботи я на практиці засвоїла як створювати 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.