МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Кафедра інформаційних систем та мереж

Лабораторна робота №1

з курсу

СПЕЦІАЛІЗОВАНІ МОВИ ПРОГРАМУВАННЯ

Виконав студент

групи ІТ-21сп

**Резніченко Р.Д**

Прийняв

**Щербак С.С.**

Львів - 2023

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

**Хід роботи**

Завдання 1. Введення користувача

Створити Python-програму, яка приймає введення користувача для двох чисел і оператора (наприклад, +, -, \*, /).

Завдання 2. Перевірка оператора.

Перевірити чи введений оператор є дійсним (тобто одним із +, -, \*, /). Якщо ні, відобразити повідомлення про помилку і попросити користувача ввести дійсний оператор.

Завдання 3. Обчислення.

Виконати обчислення на основі введення користувача (наприклад, додавання, віднімання, множення, ділення) і відобразити результат.

Завдання 4. Повторення обчислень.

Запитати користувача, чи він хоче виконати ще одне обчислення. Якщо так, дозволити йому ввести нові числа і оператор. Якщо ні, вийти з програми.

Завдання 5. Обробка помилок.

Реалізувати обробку помилок для обробки ділення на нуль або інших потенційних помилок. Відобразити відповідне повідомлення про помилку, якщо виникає помилка.

Завдання 6. Десяткові числа.

Змінити калькулятор так, щоб він обробляв десяткові числа (плаваючу кому) для більш точних обчислень.

Завдання 7. Додаткові операції.

Додати підтримку додаткових операцій, таких як піднесення до степеня (^), квадратний корінь (√) і залишок від ділення (%).

Завдання 8. Функція пам'яті.

Реалізувати функцію пам'яті, яка дозволяє користувачам зберігати і відновлювати результати. Додати можливості для зберігання та отримання значень з пам'яті.

Завдання 9. Історія обчислень.

Створити журнал, який зберігає історію попередніх обчислень, включаючи вираз і результат. Дозволити користувачам переглядати історію своїх обчислень.

Завдання 10. Налаштування користувача.

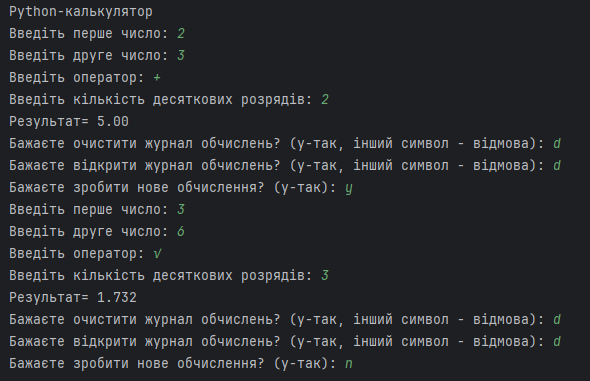
Надати користувачам можливість налаштувати поведінку калькулятора, таку як зміну кількості десяткових розрядів, які відображаються, або налаштування функцій пам'яті.

Код програми:

import math  
memoryres = []  
while True:  
 try:  
 a = float(input("Введіть перше число: "))  
 b = float(input("Введіть друге число: "))  
 oper = input("Введіть оператор: ")  
  
 if oper not in ('+', '-', '\*', '/', '%', '^', '√', 'M','R'):  
 print("Помилка! Введіть дійсний оператор: ")  
 continue  
  
 if oper == '+':  
 res = a + b  
 elif oper == '-':  
 res = a - b  
 elif oper == '\*':  
 res = a \* b  
 elif oper == '%':  
 res = a % b  
 elif oper == '^':  
 res = a \*\* b  
 elif oper == '/':  
 if b == 0:  
 print("Операція неможлива! Спробуйте ще раз.")  
 continue  
 res = a / b  
 elif oper == '√':  
 if a < 0:  
 print("Операція неможлива! Спробуйте ще раз. ")  
 continue  
 res = math.sqrt(a)  
  
 expression = f"{a} {oper} {b}"  
 memoryres.append((expression, res))

decimal\_places = int(input("Введіть кількість десяткових розрядів: "))  
 formatted\_res = "{:.{}f}".format(res, decimal\_places)  
 print(f"Результат= {formatted\_res}")  
 ques3 = input("Бажаєте очистити журнал обчислень? (y-так, інший символ - відмова): ")  
 if ques3.lower() == 'y':  
 memoryres.clear()  
 print("Журнал очищено.")  
 ques2 = input("Бажаєте відкрити журнал обчислень? (y-так, інший символ - відмова): ")  
 if ques2.lower() == 'y':  
 if memoryres:  
 print("Журнал обчислень:")  
 for i, (expression, res) in enumerate(memoryres, start=1):  
 print(f"{i}. {expression} = {formatted\_res}")  
 else:  
 print("Журнал порожній!")  
 ques1 = input("Бажаєте зробити нове обчислення? (y-так): ")  
 if ques1.lower() != 'y':  
 break  
 except ValueError:  
 print("Введено не число, спробуйте ще раз!")

На рисунку 1 зображено результат роботи програми.



*Рис.1 Результат роботи*

**Висновок:** під час виконання лабораторної роботи я створив простий консольний калькулятор на Python, який може виконувати арифметичні операції, обробляти помилки та надавати користувачу зручний інтерфейс.