

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»
ІНСТИТУТ КОМП'ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗВІТ
для лабораторної роботи № 1
з дисципліни
«Спеціалізовані мови програмування»

Виконав:
студент гр. ІТ-32
Федик В.А

Прийняв:
доц. каф. ІСМ
Щербак С.С.

Львів-2023

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

Хід виконання:

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.

Код:

```
import math
import re

def add(num1, num2):
    return num1 + num2

def subtract(num1, num2):
```

```
def multiple(num1, num2):  
    return num1 * num2
```

```
def divide(num1, num2):  
    if num2 == 0:  
        raise ZeroDivisionError("Impossible to divide")  
    return num1 / num2
```

```
def raise_to_a_power(num1, num2):  
    return num1 ** num2
```

```
def compute_square_root(num):  
    if num < 0:  
        raise ArithmeticError("Number is negative, therefore it is impossible to calculate the square root")  
    return math.sqrt(num)
```

```
def divide_by_modulo(num1, num2):  
    return num1 % num2
```

```
def view_history():  
    if not history_of_calculations:  
        print("There is nothing in history")  
    else:  
        print("History of results:")  
        for i in history_of_calculations:  
            for j in i:  
                if isinstance(j, float):  
                    print("{:.{}f}".format(float(j), decimal_places) + " ", end="")  
                else:  
                    print(str(j) + " ", end="")  
            print()
```

```
def view_settings():  
    print("\tSettings:")  
    print("\tDecimal places are " + str(decimal_places))
```

```
def change_decimal_places(value):  
    if value <= 0:  
        raise ArithmeticError("Decimal digits can't be negative or 0")
```

```
global decimal_places
decimal_places = value
```

```
history_of_calculations = []
decimal_places = 2
```

```
while True:
```

```
    print("Options: ")
    print("1. Add numbers")
    print("2. Subtract numbers")
    print("3. Multiple numbers")
    print("4. Divide numbers")
    print("5. Raise to a power")
    print("6. Divide by modulo")
    print("7. Compute the square root")
    print("8. View history")
    print("9. Open settings")
    print("0. Exit")
```

```
input_value = input("Your option is ")
```

```
if re.match("^[1-6]$", input_value):
    first_number = float(input("Enter first number: "))
    second_number = float(input("Enter second number: "))
```

```
    try:
```

```
        match input_value:
            case "1":
                operand = "+"
                result = add(first_number, second_number)
            case "2":
                operand = "-"
                result = subtract(first_number, second_number)
            case "3":
                operand = "*"
                result = multiple(first_number, second_number)
            case "4":
                operand = "/"
                result = divide(first_number, second_number)
            case "5":
                operand = "**"
                result = raise_to_a_power(first_number, second_number)
            case "6":
                operand = "%"
                result = divide_by_modulo(first_number, second_number)
```

```
    history_of_calculations += [(first_number, operand, second_number, "=", result)]
```

```

        print("Result is " + (" {:. " + str(decimal_places) + "f} ").format(float(result)) + "\n"))
    except ZeroDivisionError as e:
        print(e + "\n")
elif input_value == "7":
    try:
        number = float(input("Enter number: "))
        result = compute_square_root(number)

        history_of_calculations += [("√", number, "=", result)]
        print("Result is " + (" {:. " + str(decimal_places) + "f} ").format(float(result)) + "\n"))
    except ArithmeticError as e:
        print(str(e) + "\n")
elif input_value == "8":
    view_history()
    print()
elif input_value == "9":
    while True:
        print("\tSettings options:")
        print("\t1. View settings")
        print("\t2. Change decimal places")
        print("\t3. Clean all records")
        print("\t0. Exit from the settings mode")

        inner_input_value = str(input("\tYour option is "))

        if inner_input_value == "1":
            view_settings()
            print()
        elif inner_input_value == "2":
            new_value = int(input("\tEnter a new value for decimal places: "))

            try:
                change_decimal_places(new_value)
                print()
            except ArithmeticError as e:
                print("\t" + str(e) + "\n")
        elif inner_input_value == "3":
            history_of_calculations.clear()
            print()
        elif inner_input_value == "0":
            print()
            break
        else:
            print("\tYou have just entered a wrong option\n")
elif input_value == "0":
    break
else:
    print("You have just entered a wrong option\n")

```

Посилання на GitHub-репозиторій: <https://github.com/fedykvitalik2004/spl>

Висновки: 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.