МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

ІНСТИТУТ КОМП’ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА  ІСМ



ЗВІТ

про виконання лабораторної роботи №1

з дисципліни

«Спеціалізовані мови програмування»

студента групи ІТ-32

Ткачишина Юрія

Прийняв 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

**Індивідуальне завдання**

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.

**Хід виконання:**

Код:  
import math

import re

def calculate(num1, num2, operator, decimal\_places):

result = None

if operator == '+':

result = num1 + num2

elif operator == '-':

result = num1 - num2

elif operator == '\*':

result = num1 \* num2

elif operator == '/':

if num2 == 0:

print("Ділення на 0 неможливе.")

return None

else:

result = num1 / num2

elif operator == '^':

result = num1 \*\* num2

elif operator == 'sqrt':

if num1 < 0:

print("Неможливо взяти корінь з від'ємного числа.")

return None

else:

result = math.sqrt(num1)

elif operator == '%':

result = num1 % num2

else:

print("Невідомий знак оператор")

return None

return round(result, decimal\_places) if result is not None else result

#CREATED Get\_input function for code optimization and easy reading

def get\_input():

try:

num1 = float(input("Введіть перше число: "))

except ValueError:

print("Помилка: недійсне число.")

return None, None, None

valid\_operators = ['+', '-', '\*', '/', '^', 'sqrt', '%']

operator = input(f"Введіть знак оператор ({', '.join(valid\_operators)}): ")

while operator not in valid\_operators:

print("Помилка: недійсний знак оператор.")

operator = input(f"Введіть дійсний знак оператор ({', '.join(valid\_operators)}): ")

num2 = 0

if operator != 'sqrt':

try:

num2 = float(input("Введіть друге число: "))

except ValueError:

print("Помилка: недійсне число.")

return None, None, None

return num1, operator, num2

memory = None

history = []

decimal\_places = 2 # Кількість десяткових розрядів за замовчуванням

while True:

# Налаштування користувача

settings = input("Зайти у налаштування? (y/n): ")

if settings.lower() == 'y':

decimal\_places = int(input("Введіть кількість десяткових розрядів для відображеня: "))

num1, operator, num2 = get\_input()

if num1 is not None:

result = calculate(num1, num2, operator, decimal\_places)

if result is not None:

history.append(f"{num1} {operator} {num2 if operator != 'sqrt' else ''} = {result}")

print(f"Результат: {result}")

mem\_option = input("Зберегти результат у пам'яті? (y/n): ")

if mem\_option.lower() == 'y':

memory = result

if memory is not None:

use\_mem = input(f"Використати збережене значення {memory} як перше число? (y/n): ")

if use\_mem.lower() == 'y':

num1 = memory

view\_history = input("Переглянути історію операцій обчислень? (y/n): ")

if view\_history.lower() == 'y':

for i, entry in enumerate(history):

print(f"{i + 1}. {entry}")

repeat = input("Виконати ще одне обчислення? (y/n): ")

if repeat.lower() != 'y':

break

**Висновок:** 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.