Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Звіт про виконання лабораторної роботи №1

з дисципліни «Спеціалізовані мови програмування»

на тему «Введення в Python»

Виконала:  
студентка групи РІ-31  
Заплетнюк Ю.І.

Прийняв:  
Щербак С.С.

**Мета роботи:** 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.

**Код програмної реалізації**

**functions.py**

import math

memory = 0

history = []

def get\_number(prompt):

while True:

try:

return float(input(prompt))

except ValueError as e:

print("Please enter a number.")

raise e

def get\_operator():

while True:

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

if operator in ['+', '-', '\*', '/', '^', '√', '%']:

return operator

print("Invalid operator! Please try again.")

def calculate(num1, num2, operator):

try:

if operator == '+':

return num1 + num2

elif operator == '-':

return num1 - num2

elif operator == '\*':

return num1 \* num2

elif operator == '/':

if num2 == 0:

raise ZeroDivisionError("Division by zero is not possible.")

return num1 / num2

elif operator == '^':

return num1 \*\* num2

elif operator == '√':

if num1 < 0:

raise ValueError("Square root of a negative number is not defined.")

return math.sqrt(num1)

elif operator == '%':

return num1 % num2

except (ZeroDivisionError, ValueError) as e:

print(e)

raise e

def store\_in\_memory(result):

global memory

memory = result

print(f"Result {result} has been stored in memory.")

def add\_to\_history(expression, result):

history.append(f"{expression} = {result}")

def view\_history():

if not history:

print("History is empty.")

else:

print("Calculation history:")

for record in history:

print(record)

def main():

while True:

use\_memory = input("Use value from memory? (yes/no): ").strip().lower() == "yes"

if use\_memory:

num1 = memory

else:

num1 = get\_number("Enter the first number: ")

operator = get\_operator()

if operator == '√':

result = calculate(num1, None, operator)

expression = f"√{num1}"

else:

num2 = get\_number("Enter the second number: ")

result = calculate(num1, num2, operator)

expression = f"{num1} {operator} {num2}"

if result is not None:

print(f"Result: {result}")

add\_to\_history(expression, result)

if input("Save the result in memory? (yes/no): ").strip().lower() == "yes":

store\_in\_memory(result)

if input("View calculation history? (yes/no): ").strip().lower() == "yes":

view\_history()

if input("Do another calculation? (yes/no): ").strip().lower() != "yes":

break

**runner.py**

import sys

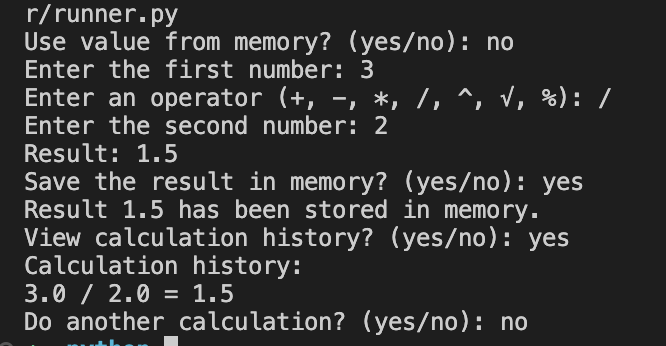
import os

sys.path.append(os.path.join(os.path.dirname(os.path.abspath(\_\_file\_\_)), '..'))

from functions.functions import main

main()

На рис. 1 показано результат виконання завдання.

  
Рис. 1. Результат виконання завдання

**Висновки:** виконавши завдання лабораторної роботи, було створено простий консольний калькулятор. Цей проєкт допоміг вивчити основний синтаксис Python і концепції, такі як введення користувача, умовні оператори, цикли та обробка помилок.