мІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

нАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «лЬВІВСЬКА ПОЛІТЕХНІКА»

ІНСТИТУТ КОМП’ЮТЕРНИХ НАУК ТА ІНОРМАЦІЙНИХ ТЕХНОЛОГІЙ

|  |
| --- |
|  |



ЗВІТ

Про виконання лабораторної роботи № 1

З дисципліни

«Спеціалізовані мови програмування»

Виконала:

студентка гр. ІТ-31

Проців Р.В.

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 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.

**Програмний код**:

calculation\_history = []

decimal\_places = 2

memory\_option = 'calculation'

def settings\_menu():

while True:

print("Меню налаштувань:")

print("1. Змінити кількість десяткових розрядів")

print("2. Налаштування пам'яті")

print("3. Вихід з меню налаштувань"

setting\_option = input("Виберіть опцію: ")

if setting\_option == '1':

global decimal\_places

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

elif setting\_option == '2':

print("Функція пам'яті:")

print("1. Виводити історію результатів")

print("2. Виводити історію обчислень")

option = input("Виберіть опцію (1 або 2): ")

if option == '1':

global memory\_option

memory\_option = 'results'

elif option == '2':

memory\_option = 'calculation'

elif setting\_option == '3':

break

def calc\_memory(history):

if len(history) != 0:

print("Історія обчислень:")

for entry in calculation\_history:

num1, operator, num2, result = entry

if memory\_option == 'calculation':

if(operator == '√'):

print(f"{operator} {num1} = {result:.{decimal\_places}f}")

else:

print(f"{num1} {operator} {num2} = {result:.{decimal\_places}f}")

else:

print(f"{result:.{decimal\_places}}")

else:

print("Пам'ять порожня.")

while True:

print("Головне меню:")

print("1. Викликати меню налаштувань")

print("2. Обчислення")

print("3. Переглянути історію")

print("4. Вихід")

option = input("Виберіть опцію (1, 2, 3 або 4): ")

if option == '1':

settings\_menu()

elif option == '2':

while True:

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

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

while True:

while True:

operator = input("Введіть оператор (+, -, \*, /, ^ для піднесення до степеня, √ для квадратного кореня першого числа, % для залишку від ділення): ")

if operator not in ['+', '-', '\*', '/', '^', '√', '%']:

print("Невірний оператор. Будь ласка, введіть один із +, -, \*, /, ^, √, %.")

else:

break

if operator == '+':

result = num1 + num2

elif operator == '-':

result = num1 - num2

elif operator == '\*':

result = num1 \* num2

elif operator == '/':

if num2 == 0:

print("Помилка: Ділення на нуль.")

break

else:

result = num1 / num2

elif operator == '^':

result = num1 \*\* num2

elif operator == '√':

result = num1 \*\* 0.5

elif operator == '%':

result = num1 % num2

calculation\_history.append((num1, operator, num2, result))

if(operator == '√'):

print(f"Результат: {operator} {num1} = {result:.{decimal\_places}f}")

else:

print(f"Результат: {num1} {operator} {num2} = {result:.{decimal\_places}f}")

another\_calculation = input("Виконати ще одне обчислення? (Так/Ні): ")

if another\_calculation.lower() != 'так':

break

else:

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

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

break

elif option == '3':

calc\_memory(calculation\_history)

elif option == '4':

break

**GitHub Repository:** <https://github.com/fxxwol/python/blob/main/lab_1.py>

**Висновок:** в результаті виконання даної лабораторної роботи було створено простий консольний калькулятор на Python, який може виконувати арифметичні операції, обробляти помилки та надавати користувачу зручний інтерфейс. Було вивчено основний синтаксис Python і концепції, такі як введення користувача, умовні оператори, цикли та обробка помилок.