МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра інформаційних систем та мереж

Лабораторна робота №1

з дисципліни

СПЕЦІАЛІЗОВАНІ МОВИ ПРОГРАМУВАННЯ

на тему

Введення в Python

Виконала:

ст. гр. РІ-31

Оксана ЛЕСЮК

Прийняв:

Сергій Сергійович ЩЕРБАК

Львів-2024

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

**Завдання:**

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.

**Хід роботи**

**functions.py:**

def calculator():

decimal\_places = int(input("Enter the number of decimal places: "))

while True:

try:

firstNumber = input("Enter the first number or press m to memory: ").lower()

if firstNumber == 'm':

if globalVariables.memory is None:

print("Memory is empty!")

continue

firstNumber = globalVariables.memory

else:

firstNumber = float(firstNumber)

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

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

print("Wrong operator! Try again!")

continue

if operator != "√": # √ need only one number

secondNumber = input("Enter the second number or press m to memory: ").lower()

if secondNumber == 'm':

if globalVariables.memory is None:

print("Memory is empty!")

continue

secondNumber = globalVariables.memory

else:

secondNumber = float(secondNumber)

else:

secondNumber = None

result = calculateExpression(firstNumber, secondNumber, operator)

if result is None:

continue

result = round(result, decimal\_places)

print(f"Result: {result}")

expression = f"{firstNumber} {operator} {secondNumber if secondNumber is not None else ''} = {result}"

globalVariables.history.append(expression) # expression recording

save\_to\_memory = input("Save result in memory? (yes/no): ").lower()

if save\_to\_memory == 'yes':

globalVariables.memory = result

print("Result has been saved in the memory.")

except ValueError:

print("Please, enter right numbers.")

continue

repeat = input("Do you want to make another operation? (yes/no): ").lower()

if repeat != 'yes':

viewHistory(globalVariables.history)

print("finished!")

break

def calculateExpression(firstNumber, secondNumber, operator):

if operator == '+':

return firstNumber + secondNumber

elif operator == '-':

return firstNumber - secondNumber

elif operator == '\*':

return firstNumber \* secondNumber

elif operator == '/':

if secondNumber == 0:

print("You cannot divide by zero!")

return None

return firstNumber / secondNumber

elif operator == '^':

return firstNumber \*\* secondNumber

elif operator == '√':

if firstNumber < 0:

print("Number cannot be lower than zero!")

return None

return math.sqrt(firstNumber)

elif operator == '%':

return firstNumber % secondNumber

def viewHistory(history):

view\_history = input("Do you want to watch history? (yes/no): ").lower()

if view\_history == 'yes':

print("History: ")

for record in globalVariables.history:

print(record)

**runner.py**:

from functions import calculator

if \_\_name\_\_ == "\_\_main\_\_":

calculator()

**globalVariables.py:**

memory = None # variable to save the previous calculation result

history = [] # empty ist

**Посилання на гіт-репозиторій:**

https://github.com/lesiukoksana/SMP-labs

**Висновок:**

Під час виконання цієї лабораторної роботи я реалізувала консольну програму-калькулятор з допомогою таких ключових елементів, як цикли, умовні оператори, функції, глобальні змінні. Ця лабораторна робота дозволила мені закріпити навички роботи з базовими конструкціями Python і створити за допомогою них програму, що реалізовує математичні операції.