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. ІТ-21сп

Борисов П.Р.

Прийняв:

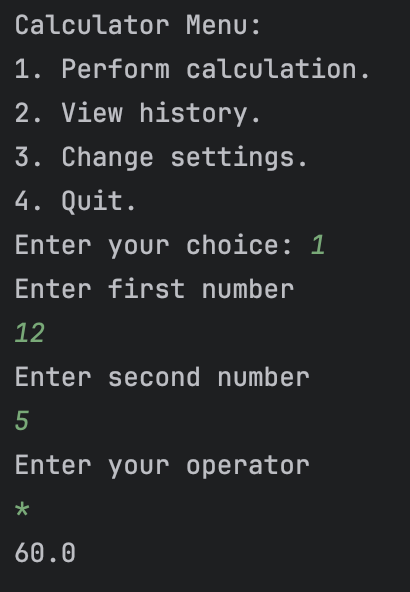
Щербак С.С.

Львів-2023

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

**План роботи**

import math  
  
memory = None  
  
history = []  
  
settings = {  
 "decimal\_places": 2,  
 "use\_memory": True  
}  
  
  
def add\_to\_history(expression, value):  
 entry = f'{expression} - {value}'  
 history.append(entry)  
  
  
def store\_in\_memory(value):  
 global memory  
 memory = value  
  
  
def retrieve\_from\_memory():  
 global memory  
 return memory  
  
  
def view\_history():  
 if not history:  
 print('Empty history')  
 return  
 for entry in history:  
 print(entry)  
  
  
def change\_settings():  
 global settings  
 print("1. Change decimal places.")  
 print("2. Toggle memory function.")  
  
 choice = input("Enter your choice: ")  
 if choice == '1':  
 try:  
 places = int(input("Enter number of decimal places (0-10): "))  
 if 0 <= places <= 10:  
 settings["decimal\_places"] = places  
 print(f"Set decimal places to {places}.")  
 else:  
 print("Invalid number of decimal places.")  
 except ValueError:  
 print("Invalid input.")  
 elif choice == '2':  
 settings["use\_memory"] = not settings["use\_memory"]  
 status = "enabled" if settings["use\_memory"] else "disabled"  
 print(f"Memory function is now {status}.")  
  
  
def calculate\_number(first\_number, second\_number, operator):  
 try:  
 if operator == '-':  
 return first\_number - second\_number  
 if operator == '+':  
 return first\_number + second\_number  
 if operator == '\*':  
 return first\_number \* second\_number  
 if operator == '%':  
 return first\_number % second\_number  
 if operator == '^':  
 return first\_number \*\* second\_number  
 if operator == '√':  
 return math.sqrt(first\_number)  
 if operator == '/':  
 if second\_number == 0: # Prevent division by zero  
 print("Error: Division by zero.")  
 return None  
 return first\_number / second\_number  
 except Exception as e:  
 print(f"An error occurred: {e}")  
 return None  
  
  
def get\_correct\_operator():  
 operator\_list = ['+', '-', '\*', '/', '%', '^', '√)']  
 while True:  
 operator = input('Enter your operator\n')  
 if operator in operator\_list:  
 return operator  
 else:  
 print('Incorrect operator')  
  
  
def get\_float\_input(prompt):  
 while True:  
 try:  
 return float(input(prompt))  
 except:  
 print("Incorrent number! Enter valid")  
  
  
def run\_calculator():  
 global memory  
  
 while True:  
 print("\nCalculator Menu:")  
 print("1. Perform calculation.")  
 print("2. View history.")  
 print("3. Change settings.")  
 print("4. Quit.")  
 user\_choice = input("Enter your choice: ")  
  
 if user\_choice == '1':  
 if memory is not None:  
 print(f'Number in memory {memory}')  
 get\_from\_memory = input('Get value from memory? Y/N \n').upper()  
 if get\_from\_memory == 'Y':  
 first\_number = memory  
 else:  
 first\_number = get\_float\_input('Enter first number\n')  
 else:  
 first\_number = get\_float\_input('Enter first number\n')  
  
 second\_number = get\_float\_input('Enter second number\n')  
 operator = get\_correct\_operator()  
  
 result = calculate\_number(first\_number, second\_number, operator)  
  
 expression = f"{first\_number} {operator} {second\_number}"  
 add\_to\_history(expression, result)  
 if settings["decimal\_places"] == 0:  
 result = int(round(result))  
 else:  
 result = round(result, settings["decimal\_places"])  
 print(result)  
  
 if settings["use\_memory"]:  
 store\_in\_memory(result)  
  
  
 elif user\_choice == '2':  
 view\_history()  
 elif user\_choice == '3':  
 change\_settings()  
 elif user\_choice == '4':  
 break  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 run\_calculator()

****

**Рис.1 Виконання програми**

**Висновок:** під час виконання лабораторної роботи я створив простий консольний калькулятор на Python, який може виконувати арифметичні операції, обробляти помилки та надавати користувачу зручний інтерфейс. Цей проєкт допоміг мені вивчити основний синтаксис Python і концепції, такі як введення користувача, умовні оператори, цикли та обробка помилок.