МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Кафедра інформаційних систем та мереж

Лабораторна робота №1

з курсу

СПЕЦІАЛІЗОВАНІ МОВИ ПРОГРАМУВАННЯ

Виконала студентка

групи ІТ-21сп

**Нечайко П.В.**

Прийняв

**Щербак С.С.**

Львів - 2023

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

**План роботи**

Завдання 1. Введення користувача

Створити Python-програму, яка приймає введення користувача для двох чисел і оператора (наприклад, +, -, \*, /).

print("Python-калькулятор Нечайко Поліни!")

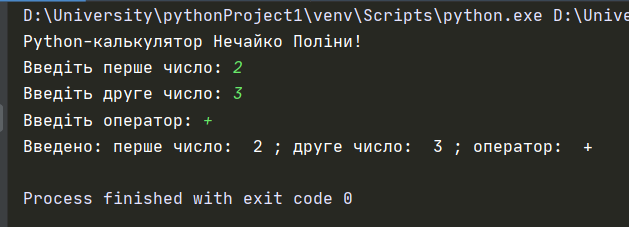
a = int(input("Введіть перше число: "))

b = int(input("Введіть друге число: "))

oper = input("Введіть оператор: ")

print("Введено: перше число: ", a, "; друге число: ", b, "; оператор: ", oper)

На рисунку 1 зображено код виконання програми.



*Рис.1 Код виконання програми*

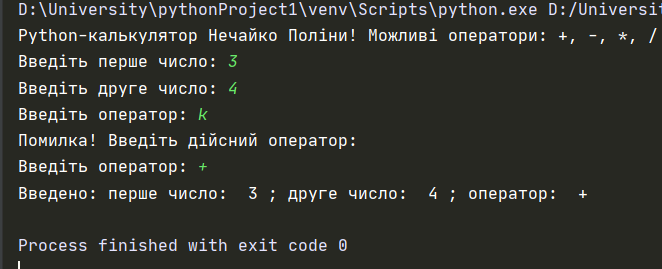
Завдання 2. Перевірка оператора.

Перевірити чи введений оператор є дійсним (тобто одним із +, -, \*, /). Якщо ні, відобразити повідомлення про помилку і попросити користувача ввести дійсний оператор.

print("Python-калькулятор Нечайко Поліни! Можливі оператори: +, -, \*, /")  
a = int(input("Введіть перше число: "))  
b = int(input("Введіть друге число: "))  
while True:  
 oper = input("Введіть оператор: ")  
 if oper in ('+', '-', '\*', '/'):  
 print("Введено: перше число: ", a, "; друге число: ", b, "; оператор: ", oper)

break  
 else:  
 print("Помилка! Введіть дійсний оператор: ")

На рисунку 2 зображено результат виконання задачі.



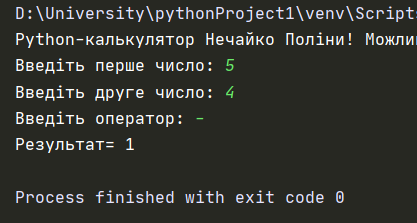
*Рис.2 Результат виконання задачі*

Завдання 3. Обчислення.

Виконати обчислення на основі введення користувача (наприклад, додавання, віднімання, множення, ділення) і відобразити результат.

print("Python-калькулятор Нечайко Поліни! Можливі оператори: +, -, \*, /")  
a = int(input("Введіть перше число: "))  
b = int(input("Введіть друге число: "))  
while True:  
 oper = input("Введіть оператор: ")  
 if oper not in ('+', '-', '\*', '/'):  
 print("Помилка! Введіть дійсний оператор: ")  
 continue  
 if oper == '+':  
 res = a + b  
 elif oper == '-':  
 res = a - b  
 elif oper == '\*':  
 res = a \* b  
 elif oper == '/':  
 res = a / b  
 print("Результат=", res)  
 break

На рисунку 3 зображено обчислення.



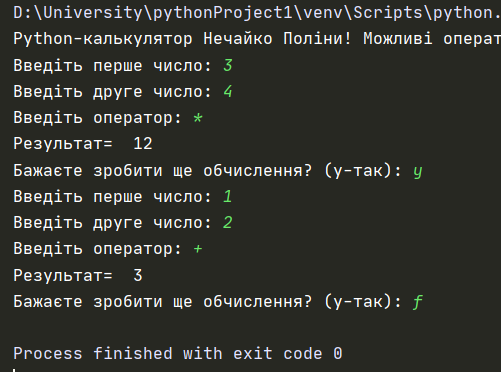
*Рис.3 Виконання програми*

Завдання 4. Повторення обчислень.

Запитати користувача, чи він хоче виконати ще одне обчислення. Якщо так, дозволити йому ввести нові числа і оператор. Якщо ні, вийти з програми.

print("Python-калькулятор Нечайко Поліни! Можливі оператори: +, -, \*, /")  
while True:  
 try:  
 a = int(input("Введіть перше число: "))  
 b = int(input("Введіть друге число: "))  
 oper = input("Введіть оператор: ")  
 if oper not in ('+', '-', '\*', '/'):  
 print("Помилка! Введіть дійсний оператор: ")  
 continue  
 if oper == '+':  
 res = a + b  
 elif oper == '-':  
 res = a - b  
 elif oper == '\*':  
 res = a \* b  
 elif oper == '/':  
 res = a / b  
 print("Результат= ", res)  
 ques = input("Бажаєте зробити ще обчислення? (y-так): ")  
 if ques.lower() != 'y':  
 break  
 except ValueError:  
 print("Помилка! Спробуйте ще раз.")

На рисунку 4 вивід роботи програми.



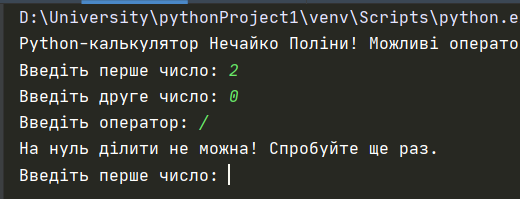
*Рис.4 Вивід роботи програми*

Завдання 5. Обробка помилок.

Реалізувати обробку помилок для обробки ділення на нуль або інших потенційних помилок. Відобразити відповідне повідомлення про помилку, якщо виникає помилка.

print("Python-калькулятор Нечайко Поліни! Можливі оператори: +, -, \*, /")  
while True:  
 try:  
 a = int(input("Введіть перше число: "))  
 b = int(input("Введіть друге число: "))  
 oper = input("Введіть оператор: ")  
 if oper not in ('+', '-', '\*', '/'):  
 print("Помилка! Введіть дійсний оператор: ")  
 continue  
  
 if oper == '+':  
 res = a + b  
 elif oper == '-':  
 res = a - b  
 elif oper == '\*':  
 res = a \* b  
 elif oper == '/':  
 if b == 0:  
 print("На нуль ділити не можна! Спробуйте ще раз.")  
 continue  
 res = a / b  
  
 print("Результат= ", res)  
 ques = input("Бажаєте зробити нове обчислення? (y-так): ")  
 if ques.lower() != 'y':  
 break  
 except ValueError:  
 print("Введено не число, спробуйте ще раз!")

На рисунку 5 зображено результат програми.



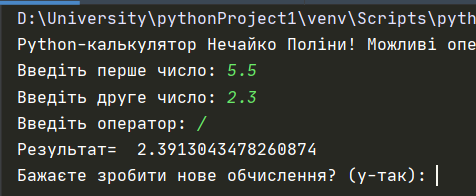
*Рис.5 Результат програм*

Завдання 6. Десяткові числа.

Змінити калькулятор так, щоб він обробляв десяткові числа (плаваючу кому) для більш точних обчислень.

print("Python-калькулятор Нечайко Поліни! Можливі оператори: +, -, \*, /")  
while True:  
 try:  
 a = float(input("Введіть перше число: "))  
 b = float(input("Введіть друге число: "))  
 oper = input("Введіть оператор: ")  
  
 if oper not in ('+', '-', '\*', '/'):  
 print("Помилка! Введіть дійсний оператор: ")  
 continue  
  
 if oper == '+':  
 res = a + b  
 elif oper == '-':  
 res = a - b  
 elif oper == '\*':  
 res = a \* b  
 elif oper == '/':  
 if b == 0:  
 print("На нуль ділити не можна! Спробуйте ще раз.")  
 continue  
 res = a / b  
 print("Результат= ", res)  
 ques = input("Бажаєте зробити нове обчислення? (y-так): ")  
 if ques.lower() != 'y':  
 break  
 except ValueError:  
 print("Введено не число, спробуйте ще раз!")

На рисунку 6 вивід виконаного завдання.



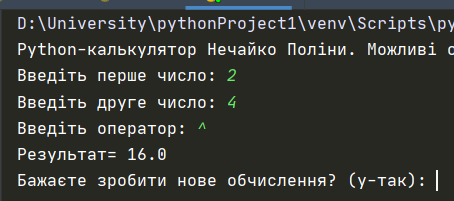
*Рис.6 Вивід виконаного завдання*

Завдання 7. Додаткові операції.

Додати підтримку додаткових операцій, таких як піднесення до степеня (^), квадратний корінь (√) і залишок від ділення (%).

import math  
print("Python-калькулятор Нечайко Поліни. Можливі оператори: +, -, \*, /, ^, √, %. Примітка: квадратний корінь береться від першого введеного числа.")  
while True:  
 try:  
 a = float(input("Введіть перше число: "))  
 b = float(input("Введіть друге число: "))  
 oper = input("Введіть оператор: ")  
  
 if oper not in ('+', '-', '\*', '/', '%', '^','√'):  
 print("Помилка! Введіть дійсний оператор: ")  
 continue  
  
 if oper == '+':  
 res = a + b  
 elif oper == '-':  
 res = a - b  
 elif oper == '\*':  
 res = a \* b  
 elif oper == '%':  
 res = a % b  
 elif oper == '^':  
 res = a \*\* b  
 elif oper == '/':  
 if b == 0:  
 print("Операція неможлива! Спробуйте ще раз.")  
 continue  
 res = a / b  
 elif oper == '√':  
 if a < 0:  
 print("Операція неможлива! Спробуйте ще раз. ")  
 continue  
 res = math.sqrt(a)  
  
 print("Результат=", res)  
  
 ques = input("Бажаєте зробити нове обчислення? (y-так): ")  
 if ques.lower() != 'y':  
 break  
 except ValueError:  
 print("Введено не число, спробуйте ще раз!")

На рисунку 7 відображення результату програми.



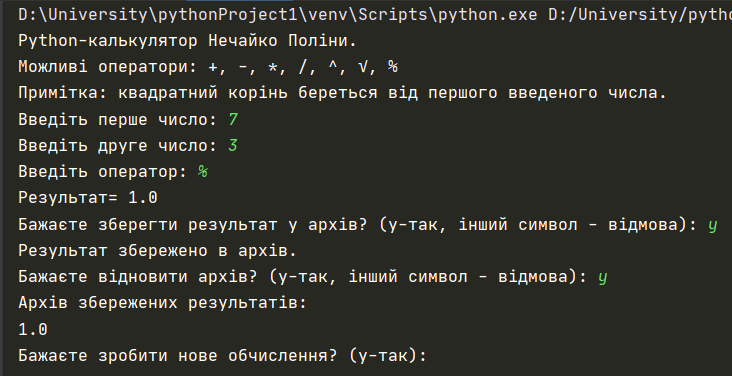
*Рис.7 Результат виконання програми*

Завдання 8. Функція пам'яті.

Реалізувати функцію пам'яті, яка дозволяє користувачам зберігати і відновлювати результати. Додати можливості для зберігання та отримання значень з пам'яті.

import math  
memoryres = []  
print("Python-калькулятор Нечайко Поліни. \nМожливі оператори: +, -, \*, /, ^, √, % \nПримітка: квадратний корінь береться від першого введеного числа.")  
while True:  
 try:  
 a = float(input("Введіть перше число: "))  
 b = float(input("Введіть друге число: "))  
 oper = input("Введіть оператор: ")  
  
 if oper not in ('+', '-', '\*', '/', '%', '^', '√', 'M','R'):  
 print("Помилка! Введіть дійсний оператор: ")  
 continue  
  
 if oper == '+':  
 res = a + b  
 elif oper == '-':  
 res = a - b  
 elif oper == '\*':  
 res = a \* b  
 elif oper == '%':  
 res = a % b  
 elif oper == '^':  
 res = a \*\* b  
 elif oper == '/':  
 if b == 0:  
 print("Операція неможлива! Спробуйте ще раз.")  
 continue  
 res = a / b  
 elif oper == '√':  
 if a < 0:  
 print("Операція неможлива! Спробуйте ще раз. ")  
 continue  
 res = math.sqrt(a)  
  
 print("Результат=", res)  
 ques3 = input("Бажаєте зберегти результат у архів? (y-так, інший символ - відмова): ")  
 if ques3.lower() == 'y':  
 memoryres.append(res)  
 print("Результат збережено в архів.")  
  
 ques2 = input("Бажаєте відновити архів? (y-так, інший символ - відмова): ")  
 if ques2.lower() == 'y':  
 if memoryres:  
 print("Архів збережених результатів:")  
 for result in memoryres:  
 print(result)  
 else:  
 print("Архів порожній!")  
  
 ques1 = input("Бажаєте зробити нове обчислення? (y-так): ")  
 if ques1.lower() != 'y':  
 break  
 except ValueError:  
 print("Введено не число, спробуйте ще раз!")

На рисунку 8 зображено виконання програми.



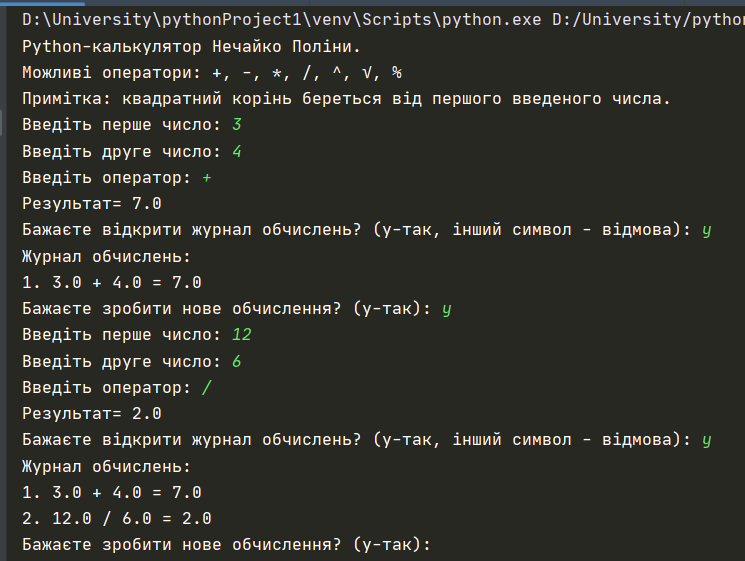
*Рис.8 Код виконання програми*

Завдання 9. Історія обчислень.

Створити журнал, який зберігає історію попередніх обчислень, включаючи вираз і результат. Дозволити користувачам переглядати історію своїх обчислень.

import math  
memoryres = []  
print("Python-калькулятор Нечайко Поліни. \nМожливі оператори: +, -, \*, /, ^, √, % \nПримітка: квадратний корінь береться від першого введеного числа.")  
while True:  
 try:  
 a = float(input("Введіть перше число: "))  
 b = float(input("Введіть друге число: "))  
 oper = input("Введіть оператор: ")  
  
 if oper not in ('+', '-', '\*', '/', '%', '^', '√', 'M','R'):  
 print("Помилка! Введіть дійсний оператор: ")  
 continue  
  
 if oper == '+':  
 res = a + b  
 elif oper == '-':  
 res = a - b  
 elif oper == '\*':  
 res = a \* b  
 elif oper == '%':  
 res = a % b  
 elif oper == '^':  
 res = a \*\* b  
 elif oper == '/':  
 if b == 0:  
 print("Операція неможлива! Спробуйте ще раз.")  
 continue  
 res = a / b  
 elif oper == '√':  
 if a < 0:  
 print("Операція неможлива! Спробуйте ще раз. ")  
 continue  
 res = math.sqrt(a)  
  
 expression = f"{a} {oper} {b}"  
 memoryres.append((expression, res))  
  
 print("Результат=", res)  
  
 ques2 = input("Бажаєте відкрити журнал обчислень? (y-так, інший символ - відмова): ")  
 if ques2.lower() == 'y':  
 if memoryres:  
 print("Журнал обчислень:")  
 for i, (expression, result) in enumerate(memoryres, start=1):  
 print(f"{i}. {expression} = {result}")  
 else:  
 print("Журнал порожній!")  
  
 ques1 = input("Бажаєте зробити нове обчислення? (y-так): ")  
 if ques1.lower() != 'y':  
 break  
 except ValueError:  
 print("Введено не число, спробуйте ще раз!")

На рисунку 9 зображено роботу калькулятора.



*Рис.9 Вивід роботи програми*

Завдання 10. Налаштування користувача.

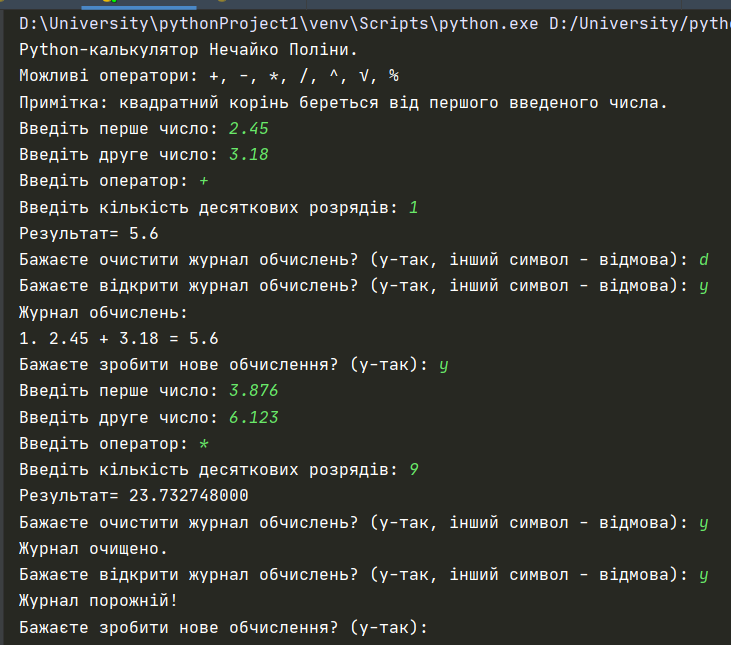
Надати користувачам можливість налаштувати поведінку калькулятора, таку як зміну кількості десяткових розрядів, які відображаються, або налаштування функцій пам'яті.

Було додано можливість користувачу обирати кількість десяткових розрядів та очищувати історію обчислень.

import math  
memoryres = []  
print("Python-калькулятор Нечайко Поліни. \nМожливі оператори: +, -, \*, /, ^, √, % \nПримітка: квадратний корінь береться від першого введеного числа.")  
while True:  
 try:  
 a = float(input("Введіть перше число: "))  
 b = float(input("Введіть друге число: "))  
 oper = input("Введіть оператор: ")  
  
 if oper not in ('+', '-', '\*', '/', '%', '^', '√', 'M','R'):  
 print("Помилка! Введіть дійсний оператор: ")  
 continue  
  
 if oper == '+':  
 res = a + b  
 elif oper == '-':  
 res = a - b  
 elif oper == '\*':  
 res = a \* b  
 elif oper == '%':  
 res = a % b  
 elif oper == '^':  
 res = a \*\* b  
 elif oper == '/':  
 if b == 0:  
 print("Операція неможлива! Спробуйте ще раз.")  
 continue  
 res = a / b  
 elif oper == '√':  
 if a < 0:  
 print("Операція неможлива! Спробуйте ще раз. ")  
 continue  
 res = math.sqrt(a)  
  
 expression = f"{a} {oper} {b}"  
 memoryres.append((expression, res))

decimal\_places = int(input("Введіть кількість десяткових розрядів: "))  
 formatted\_res = "{:.{}f}".format(res, decimal\_places)  
 print(f"Результат= {formatted\_res}")  
 ques3 = input("Бажаєте очистити журнал обчислень? (y-так, інший символ - відмова): ")  
 if ques3.lower() == 'y':  
 memoryres.clear()  
 print("Журнал очищено.")  
 ques2 = input("Бажаєте відкрити журнал обчислень? (y-так, інший символ - відмова): ")  
 if ques2.lower() == 'y':  
 if memoryres:  
 print("Журнал обчислень:")  
 for i, (expression, res) in enumerate(memoryres, start=1):  
 print(f"{i}. {expression} = {formatted\_res}")  
 else:  
 print("Журнал порожній!")  
 ques1 = input("Бажаєте зробити нове обчислення? (y-так): ")  
 if ques1.lower() != 'y':  
 break  
 except ValueError:  
 print("Введено не число, спробуйте ще раз!")

На рисунку 10 вивід роботи програми.



*Рис.10 Виконання завдання*

**Висновок:** під час виконання лабораторної роботи я створила 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.