Технології програмування

Лабораторна робота №4

**Зворотний польський запис та обчислення математичного виразу**

**Мета:** Використовуючи теоретичне підґрунтя про зворотний польський запис розробити програму на вхід якої подається математичний вираз, що має довільний набір операндів, операторів та дужок, на виході програма обчислює результат математичного виразу.

**Завдання**

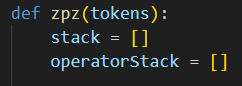
Використовуючи теоретичне відомості розробити програму яка на вхід отримує математичний вираз з довільною кількістю операндів, операторів та дужок. В першу чергу сформувати послідовність символів у ЗПН. На другому етапі виконання лабораторної роботи вирахувати результат послідовності, що була сформована, використовуючи алгоритм запису математичного виразу у ЗПН.

**Хід роботи**

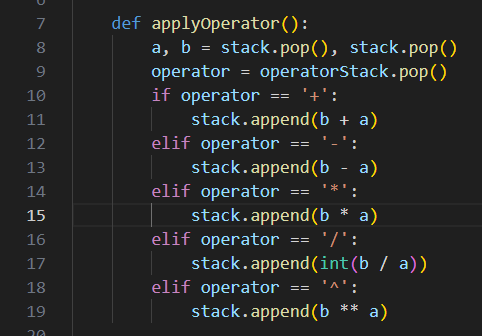
1. Створив файл lab04.py для виконання завдання.
2. Створив змінну з операторами.



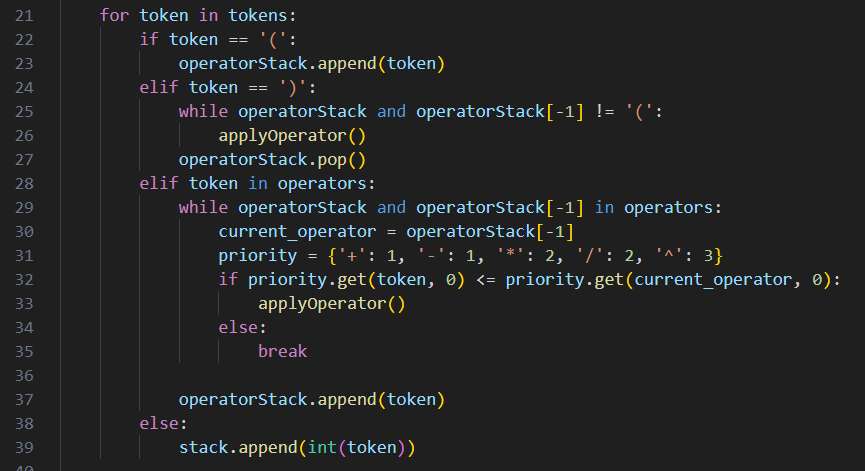
1. Написав функцію яка реалізовує Зворотний польський запис, назвав її zpz.
2. В цій функції додав 2 зміні для зберігання стеку та операторів стеку.



1. Написав функцію яка приймає операторів для розрахування математичного виразу (міг винести її окремо функції zpz, функціонал би не змінився).



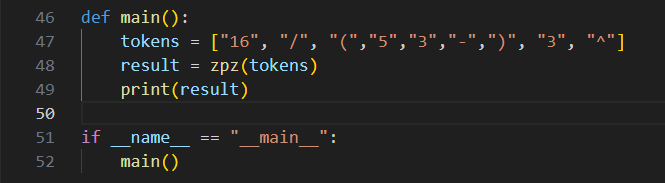
1. Створив цикл для перевірки токену (або ж ключа, або символа, який знаходиться в стеку).
2. Обробив цей цикл умовними операторами які виконуються при різних випадках з токенами, а саме коли токеном є (, або ), або оператором стеку, або коли це просте число. У випадку коли токеном є оператор стеку, проводиться перевірка задля визначення пріорітетності операції.



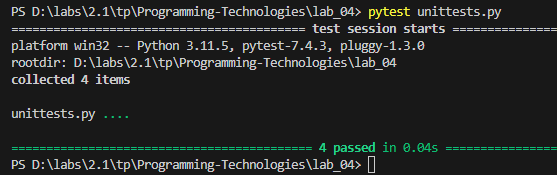
1. 41 строка виконується після завершення обробки усіх токенів у виразі. Вона використовуються для того, щоб виконати залишкові операції, які можуть залишитися в стеку операторів після того, як усі токени були оброблені.



1. Написав функцію мейн, в якій створив змінну tokens в якій знаходиться вираз, далі, створив змінну result в якій викликається основна функція zpz і приймає tokens як параметр, далі друкується результат виконання функції зі змінної result.



1. Створив файл unittest.py який виконує юніт-тести через pytest. Написав 4 теста, в яких просто підставляються різні вирази та перевіряється їх правильність.
2. Всі 4 теста пройшли перевірку успішно.



Код lab04.py:

operators = {'+', '-', '\*', '/', '^'}

def zpz(tokens):

    stack = []

    operatorStack = []

    def applyOperator():

        a, b = stack.pop(), stack.pop()

        operator = operatorStack.pop()

        if operator == '+':

            stack.append(b + a)

        elif operator == '-':

            stack.append(b - a)

        elif operator == '\*':

            stack.append(b \* a)

        elif operator == '/':

            stack.append(int(b / a))

        elif operator == '^':

            stack.append(b \*\* a)

    for token in tokens:

        if token == '(':

            operatorStack.append(token)

        elif token == ')':

            while operatorStack and operatorStack[-1] != '(':

                applyOperator()

            operatorStack.pop()

        elif token in operators:

            while operatorStack and operatorStack[-1] in operators:

                current\_operator = operatorStack[-1]

                priority = {'+': 1, '-': 1, '\*': 2, '/': 2, '^': 3}

                if priority.get(token, 0) <= priority.get(current\_operator, 0):

                    applyOperator()

                else:

                    break

            operatorStack.append(token)

        else:

            stack.append(int(token))

    while operatorStack:

        applyOperator()

    return stack.pop()

def main():

    tokens = ["16", "/", "(","5","3","-",")", "3", "^"]

    result = zpz(tokens)

    print(result)

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

    main()

Код unittests.py:

import pytest

from lab04 import \*

operators = {'+', '-', '\*', '/', '^'}

def test1():

    tokens = ["2","3","+"]

    result = zpz(tokens)

    assert result == 5

def test2():

    tokens = ["16", "/", "(","5","3","-",")", "3", "^"]

    result = zpz(tokens)

    assert result == 2

def test3():

    tokens = ["2", "5", "\*", "(", "4", "5", "\*", "3", "^", ")", "\*"]

    result = zpz(tokens)

    assert result == 5000

def test4():

    tokens = ["(", "2", "5", "\*", "(", "4", "5", "\*", "3", "^", ")", "\*", ")", "100", "/"]

    result = zpz(tokens)

    assert result == 50

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

    pytest.main()

**Висновок:**

Під час виконання цієї лабораторної роботи використовуючи теоретичне підґрунтя про зворотний польський запис розробив програму на вхід якої подається математичний вираз, що має довільний набір операндів, операторів та дужок, на виході програма обчислює результат математичного виразу.