**Звіт про виконання лабораторних завдань до лекцій з курсу Технології програмування на мові Python**

**Лабораторна робота №4**

**Тема:** воротний польський запис та обчислення математичного виразу

**Мета роботи**: 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.

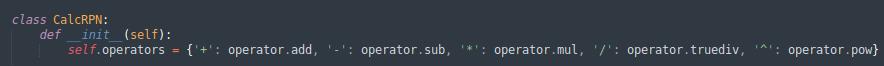
**Суть завдання:** Використовуючи теоретичне відомості розробити програму яка на вхід отримує математичний вираз з довільною кількістю операндів, операторів та дужок. В першу чергу сформувати послідовність символів у ЗПН. На другому етапі виконання лабораторної роботи вирахувати результат послідовності,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

**Теоретичні відомості:**

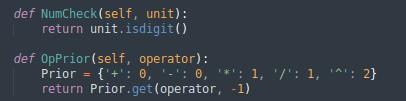
Зворотній польський запис представляє собою форму запису математичних виразів, де бінарні операції розміщуються після операндів. Наприклад, замість a + b використовується a b +. Цей запис також відомий як постфіксний і був запропонований польським логіком Яном Лукасевичем. ЗПЗ має важливі властивості, що роблять його ідеальним для трансляції коду програм. Обчислення виразів у ЗПЗ можна проводити шляхом однократного перегляду виразу, що дозволяє ефективно виконувати обчислення. Для перетворення інфіксного запису в ЗПЗ використовується алгоритм, який можна назвати "сортувальною станцією". Цей алгоритм, який базується на використанні стеку, дозволяє послідовно обробляти символи вхідного виразу і формувати вихідний вираз в ЗПЗ. Пріоритетність операцій визначається так: вираз в дужках має найвищий пріоритет, за ним йде піднесення до степеня, потім множення або ділення, і, нарешті, додавання або віднімання. Для ілюстрації, в нас є такий приклад: 3 + 4 \* 2 / (1 - 5) ^ 2. Його перетворення в ЗПЗ виглядає наступним чином: 3 4 2 \* 1 5 - 2 ^ / +.

**Хід роботи:**

**1.** Спочатку я створив клас CalcRPN, який містить методи для ініціалізації, перевірки числових одиниць, визначення пріоритету операторів, генерації ЗП та обчислення результату. Я визначив словник operators, який містить арифметичні оператори та відповідні їм функції з модуля operator.



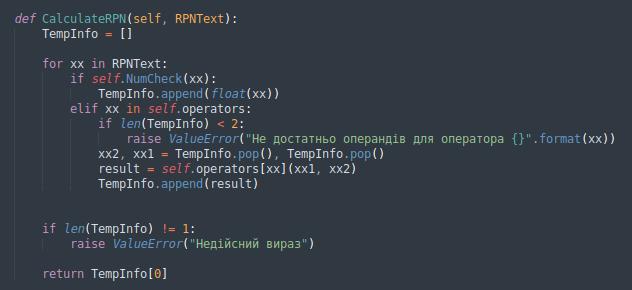
2. Далі я створив функцію NumCheck перевіряє, чи подана одиниця є числом, використовуючи метод isdigit(). Також я написав функцыю OpPrior, яка визначає пріоритет оператора, повертаючи значення зі словника Prior.



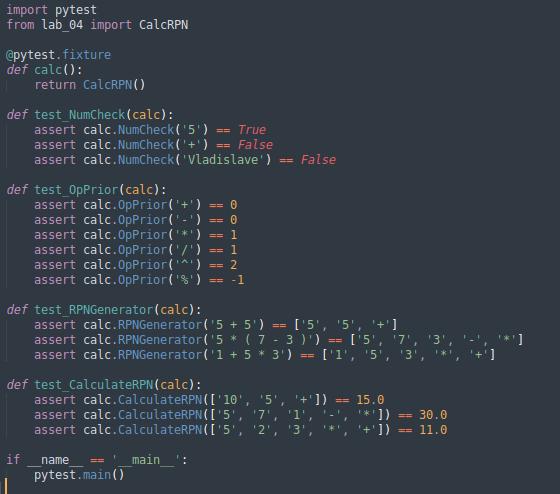
3. Далі я створив функцію RPNGenerator, яка отримує вхідний арифметичний вираз в інфіксній формі від користувача та перетворює його в ЗП. Я використовую стек TempInfo для ведення проміжних результатів та стеку RPNRes для зберігання ЗП. В процесі обробки вхідного виразу відбувається ітерація по кожному його елементу, де числа додаються до результуючого виразу у зворотньому польському записі, а оператори та дужки обробляються за допомогою стеку. На кінці я виводжу залишки зі стеку TempInfo в результат ЗП.

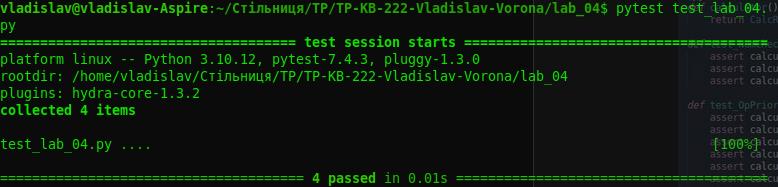


4. Далі я написав функціюCalculateRPN, яка приймає ЗП та обчислює результат. Я використовую стек TempInfo для зберігання проміжних результатів під час обчислень. Проходження через зворотній польський запис відбувається поелементно: числа додаються до стеку, а оператори обчислюють результат взятих зі стеку значень. Я визначив поведінку для кожного оператора в словнику operators. Якщо кількість чисел у стеку менше 2, то я викликаю виняток ValueError. По завершенні циклу, якщо у стеку залишився тільки один елемент, то я повертаю його як результат. Якщо у стеку залишається інше число елементів, я викликаю виняток ValueError і повідомляю користувача про недійсний вираз.



5. Я використав бібліотеку pytest для написання юніт-тестів для класу CalcRPN. За допомогою фікстури calc створюється екземпляр класу для використання в тестах. Кожна тестова функція перевіряє різні аспекти функціональності класу, такі як визначення числових операндів, пріоритет операторів, генерація зворотнього польського запису і обчислення результату. Завдяки assert порівнюються очікувані результати з фактичними.





**Висновок:** Під час виконання цієї лабораторної роботи я створив програму, яка реалізує обчислення математичних виразів, використовуючи зворотній польський запис. Для досягнення цієї мети я розробив клас CalcRPN, який включає методи для ініціалізації, перевірки числових одиниць, визначення пріоритету операторів, генерації ЗПН та обчислення результату. У процесі виконання лабораторної роботи, я вивчив та використав теоретичні відомості про зворотній польський запис, а також освоїв принципи роботи інфіксних математичних виразів та їх перетворення в ЗПН. Я розробив функції для перевірки числових одиниць, визначення пріоритету операторів та генерації ЗПН. Особливу увагу я приділив реалізації алгоритму генерації ЗПН та обчислення результату. Я використовував стеки для ефективного вирішення цих завдань. У своїй програмі я врахував різні сценарії, такі як обробка чисел, операторів та дужок. Додатково, я написав юніт-тести за допомогою бібліотеки pytest для перевірки функціональності класу CalcRPN. Ці тести допомагають забезпечити правильність роботи програми та виявити можливі помилки. В результаті виконання лабораторної роботи я отримав практичні навички у розробці програм, які використовують алгоритми обчислення математичних виразів у ЗПН, що безумовно розширило мої знання та навички в області програмування.