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

Звіт до Теми №1

Функції та змінні

Під час виконання практичного завдання до Теми №1 було надано варіанти рішення до наступних задач:

1. **Перетворення рядка**

Необхідно рядок, що вводить користувач написати зворотнім чином.

Хід виконання завдання:

Виводимо користувачу запит, питаючи в нього бажаний текст, і додаємо, в зрізі тексту, від’ємний параметр.

Текст програми:



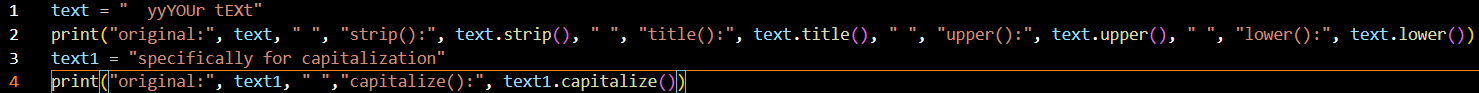
1. **Виконати тестування функцій, що працюють з рядками: strip(), capitalize(), title(), upper(), lower()**

Необхідно протестувати функції роботи з текстом як: strip(), capitalize(), title(), upper(), lower().

Хід виконання завдання:

Вводжу дві зміні з текстом для тестування функцій strip(), capitalize(), title(), upper(), lower().

Текст програми:



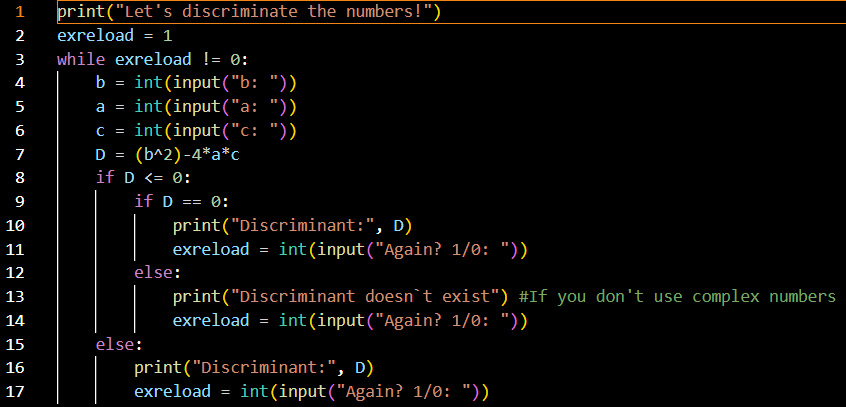
1. **Написати функцію пошуку дискримінанту квадратного рівняння**

Необхіднофункцію пошуку дискримінанту квадратного рівняння.

Хід виконання завдання:

Оголошую змінну «exreload» надаючи їй значення «1», для того щоб надати можливість користувачу обчислювати дискримінант скільки завгодно. В циклі «while», з прив’язкую до змінної «exreload», питаю в користувача значення двох коефіціентів і вільного члена квадратного рівняння та рахується значення дискримінанту. Потім значення дискримінанту проходить перевірку, від якої залежить відповідь програми.

Текст програми:



Звіт до Теми №2

Умовний перехід

Під час виконання практичного завдання до Теми №2 було надано варіанти рішення до наступних задач:

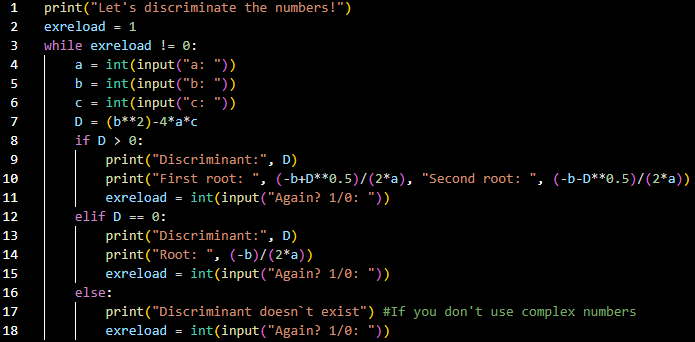
1. **Написати програми пошуку дискримінанту квадратного рівняння та його коренів**

Необхідно написати функцію пошуку коренів квадратного рівняння використовуючи функцію розрахунку дискримінанту з попередньої теми та умовні переходи.

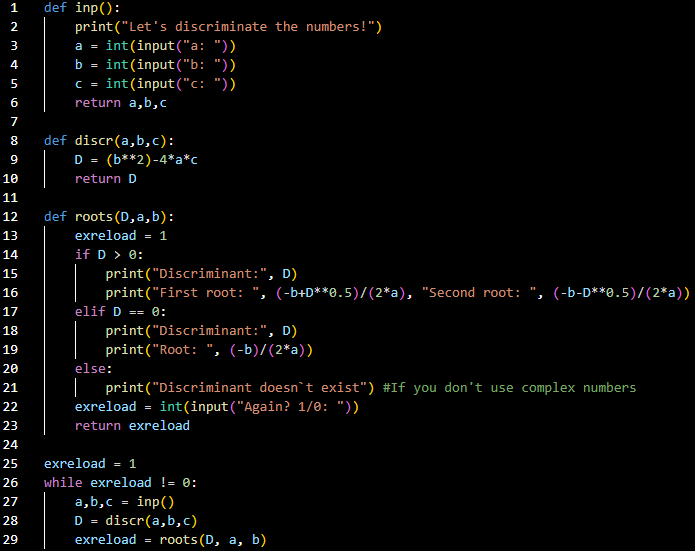
Хід виконання завдання:

Так як програма з попередньої теми не повністю підходить під умови, тому я написав два коди: один з яких звичайна модифікація попереднього коду, другий же розбитий на функції та має функціонал першого коду.

Текст програми №1:



Текст програми №2:



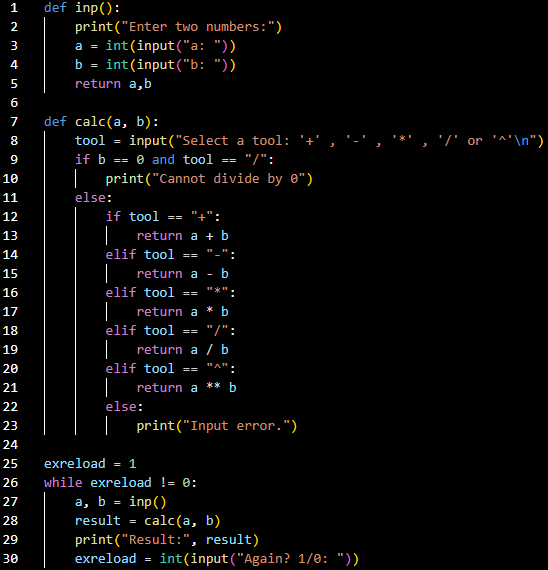
1. **Написати програму калькулятор використовуючи if else конструкцію**

Написати програму калькулятор використовуючи if else конструкцію. Кожна операція має бути виконана в окремій функції.

Хід виконання завдання:

Для оптимізації роботи, було взято дві функції(вводу та поновлення) з попередньої програми.

Текст програми:



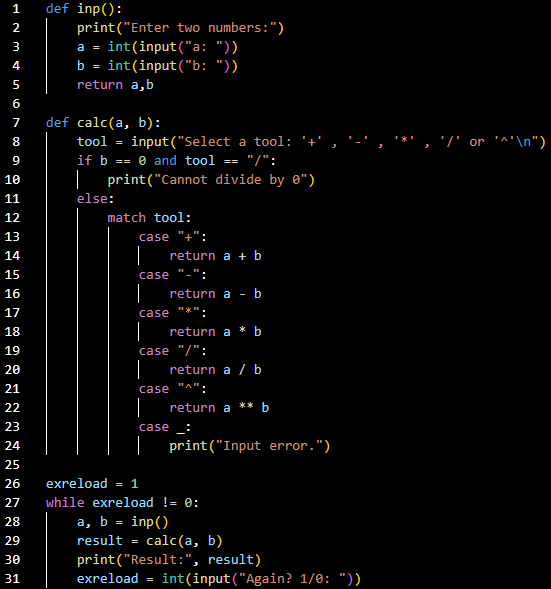
1. **Написати програму калькулятор використовуючи match конструкцію**

Написати програму калькулятор використовуючи match конструкцію. Кожна операція має бути виконана в окремій функції.

Хід виконання завдання:

Наступний код відрізняється лише тим, що конструкції if else замінені на match case.

Текст програми:



Звіт до Теми №3

Цикли

Під час виконання практичного завдання до Теми №3 було надано варіанти рішення до наступних задач:

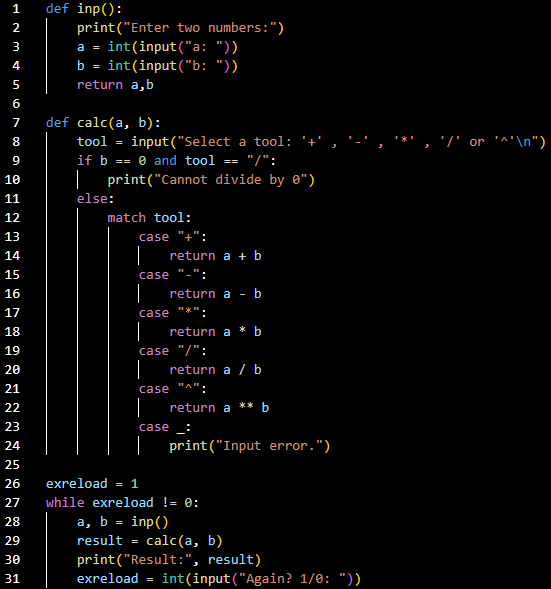
1. **Розширити програму калькулятор, що була створена до Теми 2, постійними запитами на введення нових даних та операцій**

Розширити програму калькулятор, що була створена до Теми 2, постійними запитами на введення нових даних та операцій. Реалізувати механізм завершення програми після отримання відповідної команди.

Хід виконання завдання:

Під це завдання підпадає програма, що була написана мною до Теми №2, а саме до третього завдання.

Текст програми:

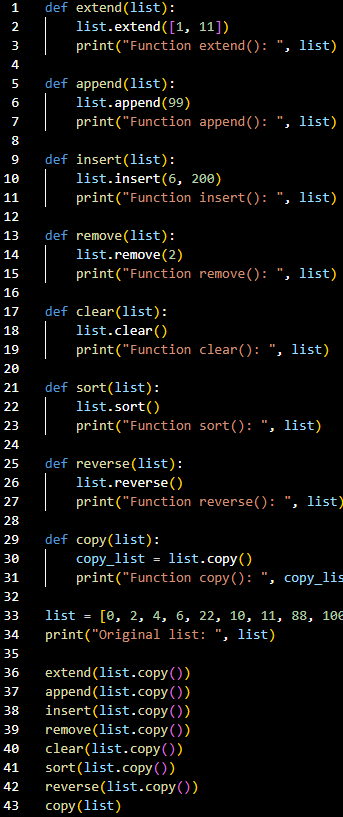


1. **Написати програму тестування функцій списків таких як: extend(), append(), insert(id, val), remove(val), clear(), sort(), reverse(), copy()**

Хід виконання завдання:

Для тестування функцій була написана програма з декількома функціями та невідсортованим списком.

Текст програми:

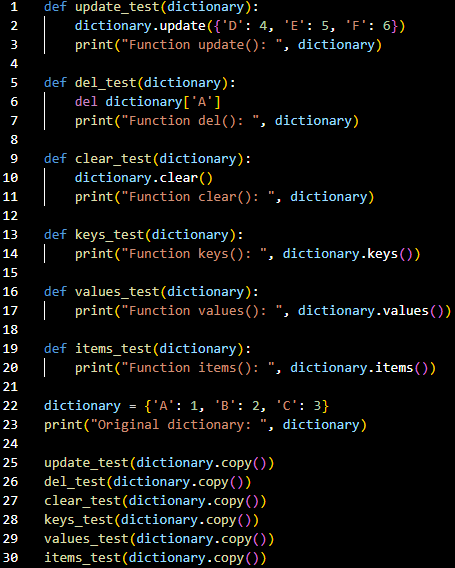


1. **Написати програму тестування функцій словників таких як: update(), del(), clear(), keys(), values(), items()**

Хід виконання завдання:

Для тестування функцій була написана програма з декількома функціями та невідсортованим словником.

Текст програми:

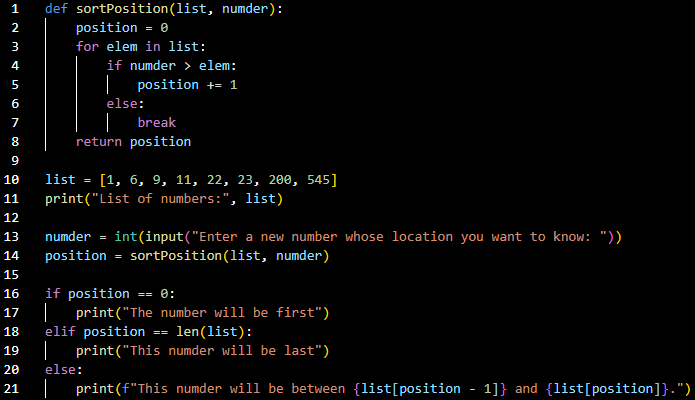


1. **Маючи відсортований список, написати функцію пошуку позиції для вставки нового елементу в список**

Хід виконання завдання:

Була написана програма з функцією пошуку позиції по відсортованому списку, що складається з чисел.

Текст програми:



Посилання на github:

https://github.com/GAGGAGX/TP-KB-231-Novyk-Maksym

Знімок екрану з посилання на github:

